Построение тетраэдральных сеток для областей с заданными поверхностными триангуляциями

А. А. Данилов[@]

В работе рассмотрена задача построения тетраэдральной сетки для произвольной области с заданной триангуляцией на границе. Предложен метод решения этой задачи, основанный на алгоритме продвигаемого фронта и надежном алгоритме построения тетраэдральных сеток для многогранников.

1. Введение

Построение сеток является одним из шагов при решении задач компьютерного моделирования. В приложениях часто используют неструктурированные тетраэдральные сетки, так как с их помощью проще дискретизировать сложные геометрические области. Одним из способов задания геометрии области может служить использование дискретизации ее границы. В качестве дискретизации границы может выступать поверхностная триангуляция. Во многих приложениях важной задачей является построение тетраэдральной сетки с заданной фиксированной поверхностной триангуляцией. Например, это может быть связано с особенностями задания граничных условий в рассматриваемой модели, или с необходимостью иметь одинаковую дискретизацию на интерфейсе между двумя областями.

В данной статье рассматривается задача построения конформной тетраэдральной сетки для произвольной области с заданной триан-

[@]Институт вычислительной математики РАН

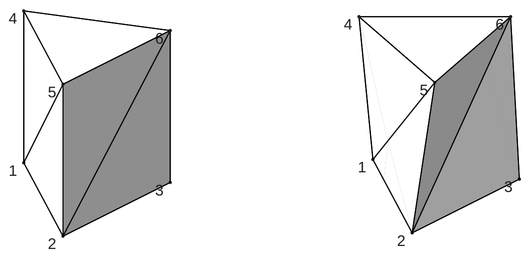


Рис. 1. Призма Schönhardt'а. Для построения тетраэдральной сетки с заданной поверхностной триангуляцией необходимо добавить хотя бы одну вершину внутри призмы. (на рис. скрыты ребра 3-4)

гуляцией на границе. Граничная триангуляция является конформной, т.е. любые два треугольника либо не пересекаются, либо имеют ровно одну общую вершину, либо имеют целое общее ребро. Построенная тетраэдральная сетка также должна быть конформной — любые два тетраэдра либо не пересекаются, либо имеют ровно одну общую вершину, либо целое общее ребро, либо целую общую грань. В аналогичной задаче для плоскости произвольный простой многоугольник может быть разбит на треугольники с помощью диагоналей без добавления дополнительных внутренних точек [1]. В трехмерном пространстве даже для выпуклых многогранников аналогичное утверждение неверно [2] (см. Рис. 1), однако наличие хотя бы одной внутренней точки решает проблему для выпуклых многогранников [3]. В общем случае доказано существование тетраэдральной сетки при условии возможности добавления новых вершин внутри области [4].

В последнее время было предложено много методов для построения неструктурированных сеток, основанных на тетраэдральном разбиении Делоне, и на методе продвигаемого фронта. Алгоритмы, основанные на тетраэдральном разбиении Делоне, гарантируют сохранение заданной границы области только в частных случаях. Если на границе выпуклой области любые два соседних треугольника лежат в разных плоскостях, то тетраэдральное разбиение Делоне будет иметь заданный след на границе области. В общем случае для восстановления триангуляции на границе могут быть использованы локальные перестроения сетки [5]. Предложены способы измельчения сетки для восстановления границы [6], а также методы построения тетраэдральных разбиений Делоне с ограничениями [7]. Метод продвигаемого фронта [8], напротив, отлично работает около произвольной границы области, но может столкнуться с проблемами внутри области. Перспективным направлением видится комбинация методов Делоне и продвигаемого фронта [9].

Во многих практических приложениях важна регулярность построенной сетки, то есть малое отклонение построенных тетраэдров от равносторонних. Если поверхностная триангуляция области уже является сильно нерегулярной, то построение регулярной тетраэдральной сетки с заданной триангуляцией на границе является неразрешимой задачей. В подобных случаях более практичным подходом может быть предварительное перестроение и улучшение поверхностной триангуляции [10].

В данной статье предлагается метод, состоящий из двух частей. Первая часть основана на алгоритме продвигаемого фронта [10] и используется для построения большей части тетраэдральной сетки. Вторая часть метода основана на конструктивном доказательстве существования тетраэдральной сетки для произвольного многогранника [4, 11]. Вместе эти два алгоритма представляют надежный способ построения тетраэдральных сеток для областей с заданными поверхностными триангуляциями. В разделе 2 будет кратко описан алгоритм продвигаемого фронта. В разделе 3 будет описан второй, надежный алгоритм. В разделе 4 обсуждаются способы получения регулярных сеток.

2. Алгоритм продвигаемого фронта

В качестве основного алгоритма для построения тетраэдральной сетки был выбран алгоритм продвигаемого фронта [10]. Этот алгоритм подразумевает наличие поверхностной триангуляции, называемой начальным фронтом. У треугольников фронта имеется ориентация, она определяет направление для дальнейшего продвижения фронта. Алгоритм является многошаговым. На каждом его шаге строится один тетраэдр на границе фронта, при этом фронт сдвигается внутрь области. После каждого шага текущий фронт отделяет построенную тетраэдральную сетку от оставшейся области. Алгоритм продвигаемого фронта не гарантирует построение тетраэдральной сетки для всей заданной области. Для разбиения оставшейся части используется второй, более надежный и медленный алгоритм, рассмотренный в следующем разделе.

Алгоритм продвигаемого фронта строит по одному тетраэдру на каждом шаге. Из фронта выбирается треугольник с самой маленькой площадью. Из его центра масс проводится нормаль внутрь области. На нормали выбирается четвертая точка будущего тетраэдра в соответствии с желаемым размером элементов сетки в данной точке. Если построенный тетраэдр пересекает текущий фронт, то начинается перебор возможного положения четвертой вершины тетраэдра среди соседних вершин фронта. Если после перебора всех возможных кандидатов тетраэдр так и не будет построен, то весь шаг повторяется с самого начала, где из фронта выбирается следующий треугольник. Когда все треугольники фронта будут рассмотрены, и ни на одном из них не получится построить тетраэдр, алгоритм останавливается. Оставшийся фронт будет ограничивать неразбитую область. Тетраэдральная сетка внутри этой области будет построена с помощью надежного алгоритма.

При построении новых вершин поддерживается следующее дополнительное условие: добавление новой вершины не должно уменьшать наименьшее из попарных расстояний между вершинами фронта. Если это условие будет выполняться при продвижении фронта, то расстояние между любыми двумя вершинами сетки будет ограничено снизу наименьшим расстоянием между вершинами начального фронта. В конечной области, ограниченной начальным фронтом, может быть расположено лишь конечное число вершин сетки. Следовательно, количество тетраэдров также будет ограничено, что гарантирует конечность времени работы алгоритма. В частности поддерживаемое условие на расстояние между вершинами накладывает ограничение на возможный размер тетраэдров в сетке.

Выполнение этого условия является более приоритетным, чем поддержание желаемого размера элементов в сетке. Предложенный алгоритм может быть использован для построения равномерных или разгрубляющихся сеток. Для получения сгущающихся сеток может быть необходимо использование других алгоритмов для перестроения полученной сетки [14, 15].

В конце работы алгоритма проводится разглаживание сетки. Для каждой внутренней вершины сетки вычисляется центр масс ее соседей, и внутренние вершины сдвигаются в новые посчитанные положения. Сдвиг выполняется, если не нарушается конформность сетки. Операция повторяется еще раз, но при этом вершины сдвигаются лишь на половину вектора, затем на треть, четверть и т.д. Уменьшение множителей позволяет избежать нарушения конформности сетки при сдвиге вершин. С другой стороны, расходимость ряда $\sum \frac{1}{n}$ не ограничивает расстояние, на которое может сдвинуться вершина. Делается от 5 до 10 операций разглаживания сетки.

3. Надежный алгоритм построения тетраэдральных сеток

В этом разделе рассматривается вариант алгоритма [4, 11], предназначеного для гарантированного построения конформной тетраэдральной сетки внутри многогранника с заданным следом на границе. Алгоритм состоит из трех частей. Сначала на множестве начальных вершин строится тетраэдральное разбиение Делоне (раздел 3.1). Далее проводится поиск недостающих граней в построенной сетке, на границе области строятся точки пересечения (раздел 3.2). В конце построенные новые вершины сдвигаются внутрь области, восстанавливая требуемую граничную триангуляцию (раздел 3.3).

3.1. Тетраэдральное разбиение Делоне. Основная цель первой части алгоритма состоит в построении вспомогательной тетраэдральной сетки на наборе вершин из заданной поверхностной триангуляции. Тетраэдральная сетка Делоне для набора вершин в пространстве — это такое разбиение, что никакая вершина не попадает внутрь ни одной из сфер, описанных вокруг тетраэдров [12].

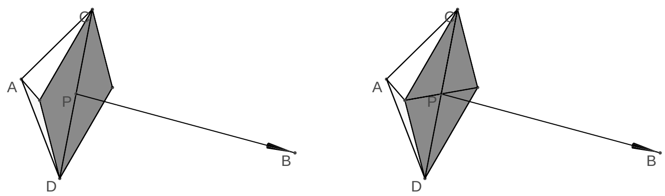
Для построения триангуляции Делоне может быть применен простейший последовательный алгоритм [13]. Вначале искусственно вводятся дополнительные 8 вершин прямоугольного параллелепипеда, полностью покрывающего исходный набор точек. Этот параллелепипед с 8 узлами разбивается на тетраэдры, удовлетворяющие условию Делоне. Далее на каждом шаге выбирается очередная вершина из исходного набора. Если вершина попала внутрь одного из тетраэдров, то этот тетраэдр делится этой вершиной на четыре части. Если вершина попала на общую грань для двух тетраэдров, то тетраэдры делятся на тройки. Если вершина попала на общее ребро для нескольких тетраэдров, то каждый тетраэдр делится этой вершиной на пару тетраэдров. Далее проводятся локальные проверки на соответствие условию Делоне. Если условие не выполняется, то проводится перестроение тетраэдров [13].

В результате будет получено тетраэдральное разбиение с вершинами из исходного набора вершин. Из алгоритма построения следует, что полученная сетка будет конформной, и будет удовлетворять условию Делоне. Но алгоритм не гарантирует совпадение требуемой поверхностной триангуляции с полученной сеткой. Часть ребер и треугольников из исходной поверхностной триангуляции может отсутствовать в построенной тетраэдральной сетке Делоне. Эти недостающие элементы будут добавлены в следующей части алгоритма.

3.2. Разбиение граничной триангуляции. Во второй части алгоритма производится поиск недостающих ребер и треугольников заданной поверхностной триангуляции. В первую очередь последовательно рассматриваются все недостающие ребра. Для требуемого ребра AB и имеющихся тетраэдров с вершиной A существуют следующие возможности:

1. Ребро AB является общим ребром, выходящим из A, для нескольких тетраэдров вокруг A. В этом случае AB уже целиком присутствует в построенной сетке.

2. Ребро AB проходит по одной из граней тетраэдра с вершиной в A. Иначе говоря, AB пересекается с некоторым ребром CD в построенной сетке. В этом случае добавляется новая вершина



а) б)

Рис. 2. Ребро проходит по грани тетраэдра: а) ребро AB пересекается с ребром CD в точке P; б) тетраэдры разбиваются на пары тетраэдров.

P — точка пересечения AB и CD. Каждый тетраэдр с ребром CD делится на пару тетраэдров с ребрами CP и PD. В частности делятся и тетраэдры, с вершиной в A, таким образом отрезок AP попадает в сетку (см. Рис. 2). Далее рассматривается оставшаяся часть PB и вершина P.

3. Ребро AB проходит через внутренность одного тетраэдра с вершиной A, и пересекает противоположную грань в некоторой точке. В этой точке создается новая вершина P, два тетраэдра, граничащие по этой грани, делятся каждый на три части. После этой операции отрезок AP попадает в сетку, и далее рассматривается оставшаяся часть PB и вершина P (см. Рис. 3).

Описанные операции могут измельчать ребра в тетраэдральной сетке, но не удаляют их из нее. Таким образом, при добавлении в сетку очередного ребра из заданной поверхностной триангуляции гарантируется сохранность уже добавленных ребер.

После добавления всех необходимых ребер полученная сетка по-прежнему является конформной. Теперь рассматриваются треугольники из исходной поверхностной триангуляции один за другим. Для каждого треугольника его ребра или их разбиения уже находятся в сетке. Однако возможна ситуация, в которой грани имеющейся сетки не покрывают требуемый треугольник полностью, в

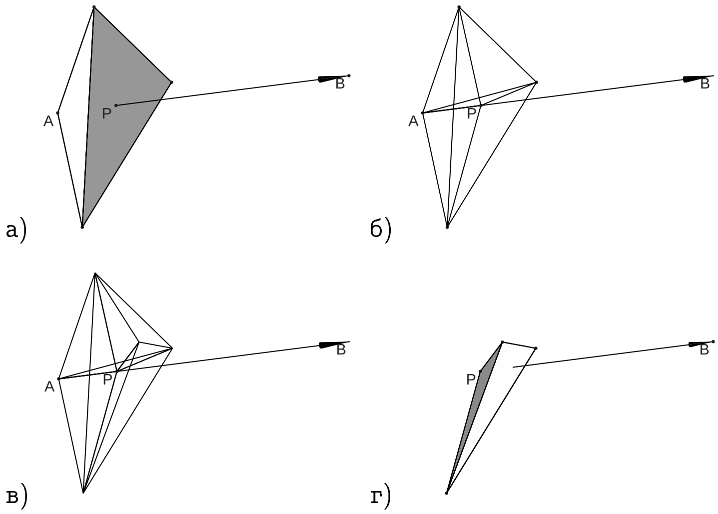


Рис. 3. Ребро проходит через тетраэдр: а) ребро AB пересекает грань в точке P; б) тетраэдр разбивается на три тетраэдра; в) соседний тетраэдр также разбивается; г) переход к отрезку PB.

этом случае найдется одно или несколько ребер, пересекающих этот треугольник. Каждое из таких ребер должно быть разделено на два отрезка с помощью новых вершин на пересечении ребра и плоскости треугольника. Соответствующие тетраэдры при этом также разбиваются. Эта операция может еще больше измельчить ребра и треугольники в поверхностной триангуляции, но не удаляет их из тетраэдральной сетки. То есть сохранность уже рассмотренных ребер и треугольников гарантируется.

В результате будет получена более мелкая конформная сетка, причем все треугольники заданной поверхностной триангуляции будут в ней присутствовать в виде разбиения. На этом этапе можно удалить из тетраэдральной сетки те тетраэдры, которые не лежат внутри заданного многогранника. Следующим шагом будет перенос новых точек пересечения внутрь области.

3.3. Восстановление граничной триангуляции. В последней части алгоритма производится перенос новых вершин внутрь области. Сначала переносятся вершины, лежащие внутри граней, затем вершины, лежащие на ребрах.

Пусть вершина P лежит внутри грани поверхностной триангуляции. Тетраэдры с вершиной в P образуют некоторую оболочку вокруг P, граница этой оболочки состоит из двух частей. Одна часть состоит из треугольников с вершиной P, лежащих на поверхности области, и образующих некоторый многоугольник, вторая часть состоит из остальных треугольников. Так как в рассмотренной оболочке конечное количество тетраэдров, и их ориентированные объемы строго положительны, то существует некоторая открытая окрестность точки P, внутри которой возможно перемещение вершины P с сохранением положительности ориентированных объемов у тетраэдров. После сдвига вершины внутрь области в пределах этой окрестности останется неразбитая часть области в виде пирамиды с вершиной P и основанием в виде многоугольника на поверхности области. Многоугольник можно разбить на треугольники с помощью диагоналей без добавления новых вершин [1], следовательно, пирамиду можно разбить на тетраэдры без добавления новых вершин. В результате будет получена конформная тетраэдральная сетка для той же области, с тем же количеством вершин, но одна из вершин будет перемещена внутрь области (см. Рис 4). Этот процесс повторяется для всех вершин, лежащих внутри треугольников поверхностной триангуляции.

Пусть теперь вершина P попала на ребро поверхностной триангуляции. Этот случай аналогичен уже рассмотренному. Граница оболочки из тетраэдров будет состоять из трех частей: двух поверхностных и одной внутренней. Соответственно, после сдвига внутрь области останется две пирамиды, которые могут быть разбиты на тетраэдры без добавления дополнительных вершин. Сдвигая по очереди все вершины внутрь области, в конце будет получена конформная тетраэдральная сетка с заданным следом на границе области.

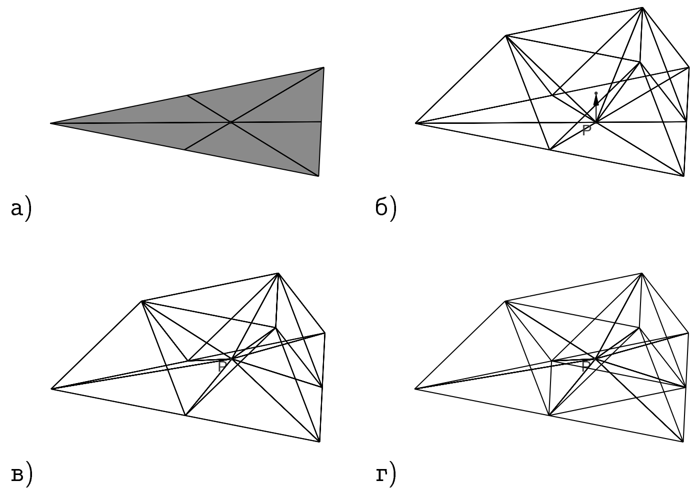


Рис. 4. Удаление вершины с грани: а) треугольники на границе области; б) оболочка из тетраэдров; в) перенос вершины внутрь области; г) разбиение пирамиды.

4. Регулярность построенных сеток

В предложенном методе в первую очередь используется алгоритм продвигаемого фронта, который строит бо́льшую часть сетки. Этот алгоритм позволяет строить разгрубляющиеся и регулярные сетки, однако не гарантирует их полное построение. Второй, надежный алгоритм строит нерегулярные сетки. То есть метод не может гарантировать построение регулярных сеток. Для получения регулярных тетраэдральных сеток можно использовать два основных подхода: предварительное перестроение и улучшение поверхностной сетки [10]; и последующее перестроение полученной сетки [14, 15].

Если для некоторой задачи сохранение поверхностной триангуляции не является необходимым условием, то перестроение сетки позволит получить регулярную поверхностную сетку. Регулярность поверхностной сетки позволяет алгоритму продвигаемого фронта строить более регулярные тетраэдральные сетки. На практике применение поверхностной регуляризации позволяет избежать использования надежного алгоритма [10].

Предложенный в статье метод решает задачу построения конформной тетраэдральной сетки. Вместе с методами предобработки поверхностных сеток и методами постобработки тетраэдральных сеток они образуют полный комплекс программ для построения сеток. Примерами таких комплексов могут служить библиотеки для работы с анизотропными сетками [16, 17].

Список литературы

[1] Препарата Ф., Шеймос М. Вычислительная геометрия: Введение. — М.: Мир. 1989. С. 285–295.

[2] Schönhardt E. Über die Zerlegung von Dreieckspolyedern in Tetraeder. // *Mathematische Annalen.* 1928. V. 98. P. 309–312.

[3] Joe B. Three-dimensional boundary-constrained triangulations. // *Proc. of 13th IMACS World Congress.* 1992. P. 215–222.

[4] George P.L., Borouchaki H. Maillage simplicial d'un polyèdre arbitraire. // *C. R. Acad. Sci. Paris.* Ser. I 338. 2004. P. 735–740.

[5] Borouchaki H., Hecht F., Saltel E., George P.L. Reasonably efficient Delaunay based mesh generator in 3 dimensions. // *Proc. of 4th Int. Meshing Roundtable.* 1995. P. 3–14.

[6] Du Q., Wang D. Recent progress in robust and quality Delaunay mesh generation. // *J. of Comp. and App. Math.* 2006. V. 195. P. 8–23.

[7] Shewchuk J.R. Constrained Delaunay tetrahedralizations and provably good boundary recovery. // *Proc. of 11th Int. Meshing Roundtable.* 2002. P. 193–204.

[8] George P.L. Automatic mesh generation and finite element computation. // *Handbook of Num.Anal.* 1996. V. 4. P. 127–148.

[9] Yang Y.J., Yong J.H., Sun J.G. An algorithm for tetrahedral mesh generation based on conforming constrained Delaunay tetrahedralization. // *Computers & Graphics.* 2005. V. 29(4). P. 606–615.

[10] Василевский Ю., Вершинин А., Данилов А., Пленкин А. Технология построения тетраэдральных сеток для областей, заданных в САПР. // *Матричные методы и технологии решения больших задач.* — М.: ИВМ РАН. 2005. С. 21–32.

[11] George P.L., Borouchaki H., Saltel E. 'Ultimate' robustness in meshing an arbitrary polyhedron. // *Int. J. Numer. Meth. Eng.* 2003. V. 58. P. 1061–1089.

[12] Делоне Б.Н. О пустоте сферы. // *Изв. АН СССР.* 1934. Т. 4, С. 793–800.

[13] Скворцов А.В. Триангуляция Делоне и её применение. — Томск: Изд-во Том. ун-та. 2002. С. 22–43.

[14] Zavattieri P.D., Dari E.A., Buscaglia G.C. Optimization strategies in unstructured mesh generation. // *Int. J. Numer. Meth. Eng.* 1996. V. 39. P. 2055–2071.

[15] Agouzal A., Lipnikov K, Vassilevski Yu. Adaptive generation of quasi-optimal tetrahedral meshes. // *East-West J. Numer. Math.* 1999. V. 7, No. 4. P. 223-244, 1999.

[16] 2D Generator of Anisotropic Meshes.
`http://sourceforge.net/projects/ani2d/`

[17] 3D Generator of Anisotropic Meshes.
`http://sourceforge.net/projects/ani3d/`