

УДК 004.925.8

© А. С. Караваяев, С. П. Копысов

МОДИФИКАЦИЯ ГЕНЕРАТОРА ШЕСТИГРАННЫХ СЕТОК, ОСНОВАННОГО НА ВОКСЕЛЬНОМ ПРЕДСТАВЛЕНИИ ГЕОМЕТРИИ

Рассматривается модификация ранее разработанного генератора шестигранных сеток из воксельных данных для построения моделей, заданных в форме CAD геометрии. Генератор относится к семейству методов, основанных на модификации регулярной сетки, и является универсальным с точки зрения возможности использования в качестве исходных данных как объемного (воксельного), так и STL-поверхностного представления геометрии модели. В настоящее время алгоритм работает с CAD моделями, описанными в хорошо известном формате STL. Вместе с тем, метод позволяет обрабатывать поверхности более высокого порядка, описанные в произвольном формате, если определены соответствующие процедуры для операций проекции и пересечения. Для определения начальной позиции узлов сетки используется полученный из STL-геометрии файл объемных данных в виде «знакопределенных полей расстояний». Разработана специальная процедура проецирования с целью адаптации построенной ортогональной сетки к границам модели. Данный подход обеспечивает аппроксимацию острых ребер и углов и выполняется перед любыми другими операциями построения сетки. Реализован дополнительный функционал для улучшения качества сетки, включающий вставку дополнительных граничных слоев, разбиение ячеек плохого качества и оптимизированное сглаживание узлов. Алгоритм протестирован на значительном числе моделей, часть из которых приведена в качестве примеров.

Ключевые слова: генератор шестигранных сеток, объемные данные, CAD модель, STL геометрия.

DOI: [10.35634/vm200308](https://doi.org/10.35634/vm200308)

Введение

Первоначальным этапом построения сетки является описание исходной геометрии модели. Поверхностное представление, описывающее границу модели в виде набора ограничивающих поверхностей, является основным способом, применяемым в вычислительных методах и широко используется в существующих CAD-системах.

Вместе с тем, для некоторых приложений, таких как биомоделирование, более естественным является вариант использования объемных (воксельных) данных, полученных со сканеров магнитно-резонансной или компьютерной томографии [1]. Точное восстановление острых углов и ребер не является критичным при построении сеток биологических тканей и органов. Используемые в этих целях алгоритмы обладают высокой производительностью, робастностью и основываются на попарном объединении вокселей декартовой сетки в шестигранные с последующей коррекцией позиции граничных узлов.

Кроме того, окончательно не решена задача автоматической генерации шестигранных сеток заданного качества, обеспечивающих точное описание границы произвольной CAD-модели [2, 3].

Специально для этого разработано значительное число алгоритмов, обладающих своими преимуществами и недостатками. Так, в методах декомпозиции разделяют модель на составные части, для каждой из которых легко сгенерировать шестигранную сетку. Такие подходы создают ячейки хорошего качества, но являются полуавтоматическими, в связи с чем

затратными по времени при построении сложных геометрий. Следует отметить, что разработано большое количество методик, направленных на автоматизацию процесса декомпозиции («medial surface transformation», «midpoint subdivision», «singularity-restricted frame fields») [3]. Методы подвижного фронта («sweeping», «paving», «plastering») генерируют шестигранную сетку от границы внутрь области [4]. Данные подходы не всегда надежны, в особенности для случаев слияния фронтов в процессе работы. Проблемы с робастностью также отмечаются в методике «spatial twist continuum», использующей двойственное представление шестигранной сетки для определения глобальной связи ее компонент [5].

В последние годы много исследований посвящено развитию непрямых методов, объединяющих тетраэдры в шестигранники с использованием различных способов [2,6]. В большинстве случаев данные алгоритмы генерируют смешанную сетку с преобладанием шестигранных ячеек.

Методы регулярной сетки и на основе октодеревьев обладают надежностью и высокой скоростью построения, однако зависят от ориентации исходной ортогональной сетки и создают шестигранники плохого качества вдоль границы [3, 7]. Также указанные алгоритмы имеют тенденцию к генерации большого числа ячеек и появлению «висячих» узлов при согласовании участков перестроения.

В данной работе рассматривается модификация ранее разработанного воксельного сеточного генератора для построения шестигранных сеток из поверхностного и объемного описания геометрии [8].

Предложенный подход принадлежит к семейству методов на основе регулярной сетки и в качестве исходной информации использует поверхностное представление в комбинации с форматом воксельных данных в виде «знакоопределенных полей расстояний», полученных из STL геометрии.

§ 1. Представление данных геометрии модели

Воксельные данные \mathcal{V} есть выборка скалярных значений функции на декартовой сетке координат $\mathcal{V} = \mathcal{F}(i, j, k)$, где i, j, k индексы в x, y, z координатах декартовой сетки. Отдельная точка $\nu^{(i,j,k)}$ декартовой сетки имеет название воксель. Изоповерхность (функция уровня, граница тела), соответствующая значению α , есть множество точек, в которых функция \mathcal{F} равняется указанной величине $I_{\mathcal{F}}(\alpha) = \{(x, y, z) | \mathcal{F}(x, y, z) = \alpha\}$.

Объемные данные могут быть представлены посредством задания изоповерхностей и их изображения в виде полигональных сеток, либо напрямую через рендеринг объема как блока данных.

В настоящее время разрабатываемый алгоритм использует стандартизированный формат томографических данных DICOM для построения сеток биологических тканей и органов. В случае CAD-моделей применяется поверхностная треугольная сетка (STL) в комбинации с файлом формата «знакоопределенные поля расстояний» (SDF).

Каждый воксель SDF файла ν сохраняет расстояние до поверхности тела Ω с учетом знака ($\mathcal{F}(\nu) > 0$, если воксель внутри тела Ω , и $\mathcal{F}(\nu) < 0$ в противном случае).

Пусть тело Ω ограничено поверхностной треугольной сеткой \mathcal{T} , тогда для произвольного вокселя ν функцию \mathcal{F} можно записать в виде

$$\mathcal{F}(\nu) = S_{\mathcal{T}}(\nu) \cdot \min_{c \in \mathcal{T}} d(c, \nu), \quad S_{\mathcal{T}}(\nu) = \begin{cases} +1, & \text{если } \nu \subseteq \Omega, \\ -1, & \text{если } \nu \not\subseteq \Omega, \end{cases}$$

где $d(c, \nu)$ обозначает минимальное расстояние между треугольником c сетки \mathcal{T} и вокселем ν .

Массив вокселей SDF файла создается из заданной STL геометрии посредством конвертации методом «brute force». Для каждого вокселя $\nu \in \mathcal{V}$ вычисляется расстояние до всех треугольников сетки \mathcal{T} после чего берется минимальное. Алгоритм естественным образом распараллеливается. Кроме того, алгоритм достаточно прост в программной реализации и вычисление поля расстояний выполняется без погрешностей.

Полученный файл объемных данных содержит всю информацию об ортогональной сетке, описывающей модель. Таким образом, результат работы сеточного генератора зависит от точности значений вокселей в \mathcal{V} .

§ 2. Генерация шестигранной сетки из STL-геометрии

Существуют различные методы представления и форматы для хранения информации о 3D моделях. Один из универсальных форматов CAD-геометрии — STL-формат, в котором граничные поверхности объемной модели описываются треугольниками (плоские грани) и их нормальными векторами. Кроме того, заслуживает внимание возможность представления STL-файлами сборочных геометрических моделей. Рассматриваемый алгоритм построения сетки из вокселей может быть адаптирован к работе с поверхностями, заданными в произвольной форме, если будут определены соответствующие процедуры проекции и пересечения узлов сетки и границы модели. Как отмечается в работе [7], построенная шестигранная сетка должна удовлетворять следующим требованиям: каждая вершина модели представляется узлом сетки; каждое ребро модели покрывается последовательностью ребер сетки.

Для представления всех вершин и ребер геометрической модели строится множество Λ «характерных» ребер и узлов модели. Множество Λ можно рассматривать как «каркас» модели, сохраняющий все ее острые «характерные» формы. Для сохранения точности описания границы результирующая шестигранная сетка должна покрывать Λ непрерывной цепочкой ее узлов и ребер.

При построении Λ не будем рассматривать ребра и треугольники поверхностной STL сетки T , которые образует двугранный угол близкий к 180° . Тогда в Λ из всех ребер $\mathcal{E} \subset T$ попадает только множество \mathcal{E}_Z , такое что $|\alpha - 180^\circ| > \alpha_{\min}$. Узлы с числом прилегающих ребер $e_i \in \mathcal{E}_Z$ не равным 0 и 2 образуют множество «характерных» вершин. Ребра \mathcal{E}_Z группируются в «характерные» ребра \mathcal{E}_Λ , где каждый элемент \mathcal{E}_Λ является замкнутой цепочкой ребер \mathcal{E}_Z или цепочкой между двумя «характерными» вершинами. Так строится «каркас» модели, при этом значением угла α_{\min} определяется точность описания «каркасом» исходной формы модели.

На рис. 1 а, б изображена STL-геометрия части втулки, а также соответствующее множество Λ , содержащее 10 вершин и 15 ребер.

Далее осуществляется построение структурированной ортогональной сетки шага h , покрывающую модель (рис. 1, в). Значение h определяет интервал между вокселями, участвующими в построении структурированной ортогональной сетки. Величина шага определяет число ячеек результирующей сетки и конечную точность описания модели. Минимальное значение h — единица, что означает рассмотрение всех вокселей \mathcal{V} в процессе построения сетки.

Построение ячейки структурированной ортогональной сетки осуществляется из любых восьми соседних вокселей, в совокупности имеющих четыре и более положительных значений.

Пусть $\dim_z \times \dim_y \times \dim_x$ размерность воксельного SDF массива \mathcal{V} и $\text{sgn}_{\mathcal{F}}^{(x,y,z)}$ сигнум-функция значения вокселя с индексами (x, y, z) , $\text{sgn}_{\mathcal{F}}^{(x,y,z)} = \text{sgn}(\mathcal{F}(\nu^{(x,y,z)}))$. В таком случае процесс построения ортогональной сетки записывается в следующем виде:

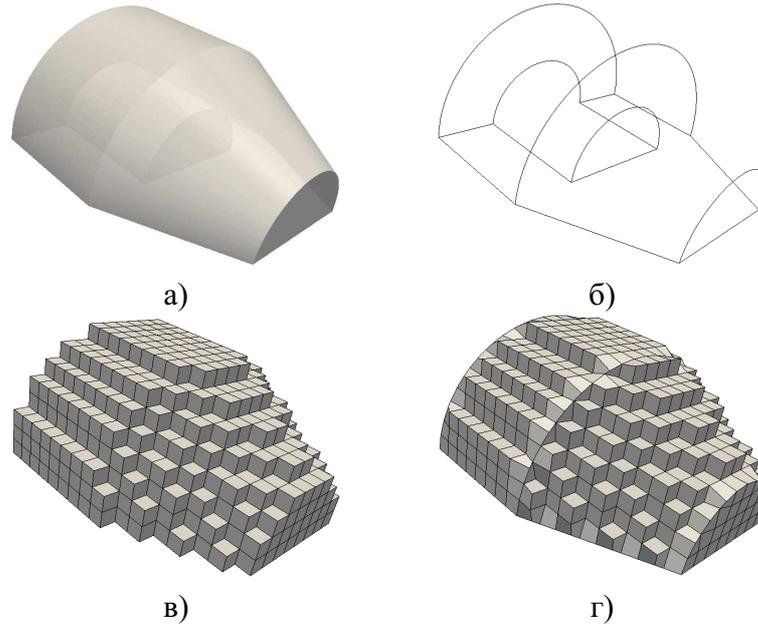


Рис. 1: Построение шестигранной сетки: а) CAD-геометрия; б) «каркас» модели; в) структурированная ортогональная сетка; г) проекция сетки на «каркас» модели

```

for (z = 1; z < dimz / h; z++)
  for (y = 1; y < dimy / h; y++)
    for (x = 1; x < dimx / h; x++)
      if (sgn $\mathcal{F}$ (x,y,z) + sgn $\mathcal{F}$ (x+h,y,z) + sgn $\mathcal{F}$ (x+h,y+h,z) + sgn $\mathcal{F}$ (x,y+h,z) + sgn $\mathcal{F}$ (x,y,z+h)
        + sgn $\mathcal{F}$ (x+h,y,z+h) + sgn $\mathcal{F}$ (x+h,y+h,z+h) + sgn $\mathcal{F}$ (x,y+h,z+h) ≥ 4) then
        Создать шестигранник c из вокселей { $\nu^{(x+i,y+j,z+k)}$ }i,j,k=0,h
      endif
    endfor
  endfor
endfor

```

Необходимо отметить, что число положительных вокселей может быть меньше четырех. Также существует возможность добавления других условий, таких как допустимое расстояние вокселей до поверхности. Например, ячейка не строится в случае, если $\exists \nu' \in \{\nu^{(x+i,y+j,z+k)}\}, i, j, k = 0, h$ такой что $\mathcal{F}(\nu') < \varepsilon_0 < 0$, где ε_0 — значение, определенное пользователем.

Следующим шагом является проецирование граничных узлов ортогональной сетки на множество Λ для обеспечения аппроксимации «характерных» вершин и ребер модели. Данная процедура отличается от методики предложенной Шнейдером [7]. Процесс адаптации выполняется до любых топологических операций и процедур сглаживания и включает три основных шага.

1. На первом из них на каждую «характерную» вершину η проецируется соответствующий узел x ортогональной сетки. Как правило, выбирается ближайший к η узел. Вместе с тем в некоторых случаях целесообразней применить иные условия. В частности, если два и более узлов x_i являются ближайшими к η , следует выбрать узел являющейся соседом для всех x_i .
2. Следующий шаг состоит в формировании «характерных» ребер модели. Вначале осуществляется проекция граничных узлов с расстоянием до «характерных» ребер мень-

ше, чем заданное значение $\varepsilon_1 \approx 0.1 \cdot h$.

3. Оставшиеся непокрытые участки Λ обрабатываются процедурой последовательной проекции узлов. Она начинается с любого узла который уже спроецирован на Λ . Следующий узел x' выбирается из граничных соседей узла x в соответствии со следующими критериями:

- граничный узел не принимается во внимание, если после проекции он будет содержать два и более вывернутых четырехугольника;
- узлы с тремя прилегающими граничными четырехугольниками имеют приоритет перед остальными;
- среди узлов, удовлетворяющих первым двум условиям, выбирается тот который расположен на минимальном расстоянии до «характерного» ребра.

На рис. 1, *г* показан результат работы алгоритма. Следует отметить, что каждый выбранный узел $x' \in \mathcal{T}$ проецируется на ближайшее ребро $e \in E_\Lambda$, и эта позиция обычно не совпадает с вершинами треугольной сетки \mathcal{T} .

После построения множества Λ , оставшиеся граничные узлы проецируются на поверхность модели. В алгоритме реализованы два метода: хорошо известный метод трассировки лучей и специальное итеративное смещение по значениям вокселей, описанная в [1]. Первая процедура является более точной, вместе с тем вторая имеет лучшую производительность, в особенности для случая STL-геометрии с большим количеством треугольников при описании моделей с криволинейными поверхностями.

Каждое итерационное уточнение состоит из двух шагов:

- проверки близости узла x к изоповерхности $I_{\mathcal{F}}(\alpha)$; если значение $\mathcal{F}(x)$ близко к α , например, $|\mathcal{F}(x) - \alpha| \approx 10^{-4} \cdot h$, тогда полагается, что x принадлежит поверхности и дальнейших итераций не требуется;
- определение новых координат узла с приращением на небольшое расстояние $x^+ = x + \delta n$, где n — нормаль узла x , а δ — величина шага на итерации; вычисление нормали n осуществлялось через осреднение нормалей, прилегающих к узлу x граничных четырехугольников [7].

В случае формата данных SDF граница модели определяется нулевой изоповерхностью со значением $\alpha = 0$.

Согласно проведенным тестам в большинстве случаев реализация этих процедур позволяет полностью описать модель так, чтобы каждому «характерному» узлу был присвоен соответствующий узел сетки и каждое «характерное» ребро полностью покрывалось цепочкой ребер расчетной сетки.

Как видно из рис. 2, *а*, проекция оставшихся узлов на поверхность позволяет обеспечить точное описание исходной STL модели.

Шаг с выделением острых ребер и узлов для случая томографических данных (КТ-снимков) трансформируется в процедуру проекции узлов сетки в положения, вычисленные с помощью алгоритма «двойственного контурирования» («Dual Contouring») [1, 8]. Данный шаг применялся для построения сетки многокомпонентной модели, изображенной на рис. 3, *г*.

Алгоритм «двойственного контурирования» используется для построения изображений по томографическим данным с помощью поверхностной четырехугольной сетки. Он позволяет до определенной степени восстановить острые грани контура модели. Вместе с тем, для случая STL-геометрии точность описания границы, полученной алгоритмом «сопряженного контурирования» (из файла формата SDF) является недостаточной.

§ 3. Улучшение качества сетки

Одной из часто применяемых процедур улучшения качества сетки является сглаживание по Лапласу. Вместе с тем этой операции недостаточно, чтобы избавиться от вывернутых ячеек сетки вдоль границы. Это связано с тем, что шестигранник может иметь две или три грани, проецируемые на одну плоскость, либо два ребра, принадлежащие одному «характерному» ребру.

Для решения этих проблем используются три стандартные процедуры, применяемые в методах на основе регулярной сетки.

Первоначально осуществляется вставка дополнительного объемного слоя граничных ячеек так, чтобы новые элементы имели только одну грань, проецируемую на реальную геометрию. Пример операции показан на рис. 2, в, з), форма светло-серых четырехугольников (рис. 2, в) значительно улучшилась с точки зрения значения вывернутых углов (рис. 2, з).

Следует отметить, что иногда для улучшения качества целесообразно пропустить вставку объемного слоя вдоль поверхностей, которые ортогональны декартовым осям координат. Например, на рис. 2 нижняя, передняя и задняя поверхности модели остались без изменений, и только для верхней криволинейной поверхности добавлен дополнительный слой ячеек.

Аналогично, второй слой элементов вставляется вокруг набора «характерных» ребер (выделенная линия на рис. 2, а, в случае если прилегающий к ним внутренний угол значительно меньше 180°). Также, операция не выполняется, если число «характерных» ребер, сходящихся в «характерной» вершине, не равно двум или трем.

На рис. 2, б представлена полученная после применения шага оптимизации сетка. Необходимо отметить, что все вывернутые углы вокруг «характерного» ребра были удалены.

Оставшиеся вырожденные элементы удаляются с помощью процедуры разбиения (шестигранники белого цвета на рис. 2 в предложенной в [7]. Все ячейки с перевернутой гранью делятся на четыре новых согласно шаблону (рис. 2, з, что обеспечивает удаление вырожденных ячеек, прилегающих к «характерному» ребру. Для поддержания конформности сетки соседние элементы разбиваются аналогичным образом (рис. 2, д).

В данном случае по шаблону разбивается как сама ячейка с четырехугольником q , содержащим «вывернутый» угол вдоль «характеристического» ребра, так и все ячейки, прилегающие к узлу $x \in q$, не находящемуся на «характеристическом» ребре (рис. 2, д). Позиции вновь генерируемых узлов вычисляются в серединах ребер ячеек, прилегающих к x .

В дальнейшем, для улучшения качества сетки применяется процедура сглаживания. В алгоритме реализованы различные типы сглаживания по Лапласу, взвешенного по длине ребер, площадям квадратов и объему шестигранников.

В более сложных случаях применяется метод оптимизированного сглаживания, предложенного в [9]. В алгоритме применяется модификация метода для случая шестигранных ячеек. Рассматриваемый подход является итеративным и может улучшить значение любой меры качества ячейки q , изменяющейся в интервале $[-1; 1]$.

Алгоритм построен для работы с тремя критериями качества ячеек шестигранной формы среди которых: нормированный якобиан, обратное аспектное соотношение и скошенность. Также используется нормированная мера Θ для угла искривленности четырехугольной грани $\bar{q}_\Theta = (\pi - \Theta)/\pi$.

Поскольку метод оптимизированного сглаживания является затратным по времени, его применение ограничивалось областями, где сглаживание по Лапласу оказывалось малоэффективным.

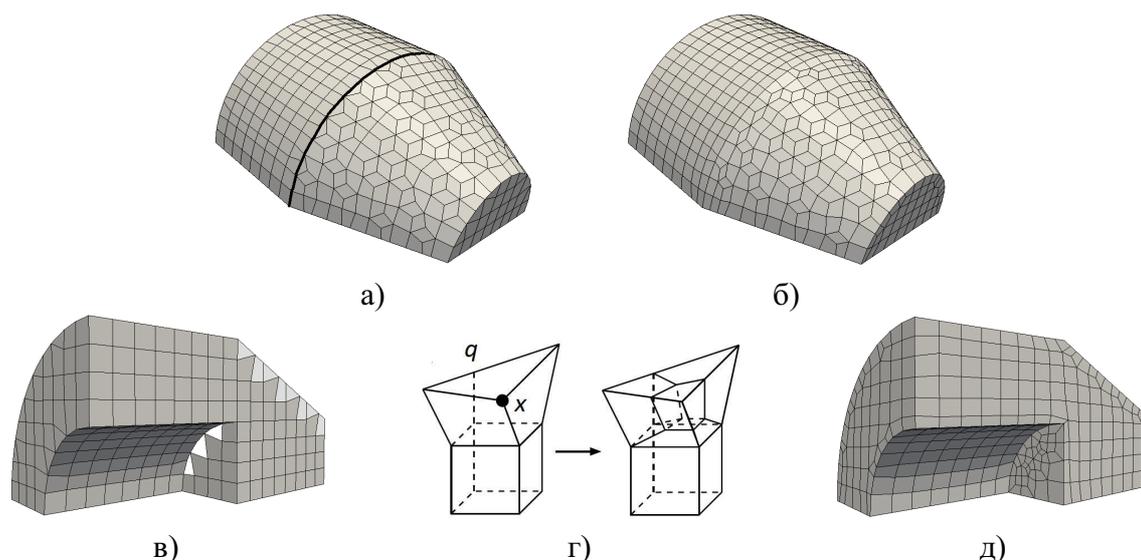


Рис. 2: Построение шестигранной сетки: а) ортогональная сетка проецируемая на границу модели; б) вставка дополнительного слоя ячеек; в) сетка после разбиения вывернутых ячеек и вставки граничного слоя; г) шаблон разбиения на четыре ячейки; д) результирующая сетка

§ 4. Тестовые примеры

Рассмотрим результаты работы алгоритма на трех типах САД-геометрии и объемном массиве томографических данных.

Все представленные шестигранные сетки не содержат вывернутых клеток. Минимально нормированный якобиан имеет значение $Q_J \geq 0.2$, что приемлемо для конечного-элементных алгоритмов. Данная мера была выбрана в качестве контрольной в ходе работы процедуры оптимизированного сглаживания. Значения мер качества сеток приведены в таблице 1. Вычислительные эксперименты проводились на сервере, на котором установлены два шестиядерных процессора Intel Xeon E-2620 тактовой частотой 2,1 ГГц с оперативной памятью объемом 64 Гбайт.

Рис. 3, а демонстрирует разрез втулки с цилиндрическим включением. Полученная шестигранная сетка содержит $N = 15930$ узлов и $C = 13842$ ячеек. Относительно большое значение максимального аспектного отношения $Q_a = 33.8$ появляется в ячейках граничного слоя вокруг включения.

Максимальная скошенность ячеек сетки модели звезды составляет $Q_s = 0.84$. Несмотря на то, что все границы модели плоские, максимальный угол искривленности внутренних четырехугольников сетки равен $\Theta_w = 61.1^\circ$. Это связано с тем, что при оптимизированном сглаживании единственным критерием оптимизации ячеек являлось значение нормированного якобиана.

При наличии только двух «характерных» ребер и отсутствия вершин сетка модели пружины имеет лучшие значения всех анализируемых метрик качества ячеек: $Q_J = 0.4$, $Q_a = 4.2$, $Q_s = 0.8$.

На рис. 3, б показан пример шестигранной сетки для модели головы человека, сгенерированной из файла данных типа DICOM. Значения шкалы Хаунсфилда (значения извлекаемых изоповерхностей) для каждого материала были следующими: мягкая ткань $\alpha = -720$, мозг $\alpha = 1$, череп $\alpha = 172$. В данном случае, в дополнение к нормированному якобиану, также оптимизировался угол искривленности четырехугольной грани. Показатели качества полученной сетки также можно увидеть в таблице 1.

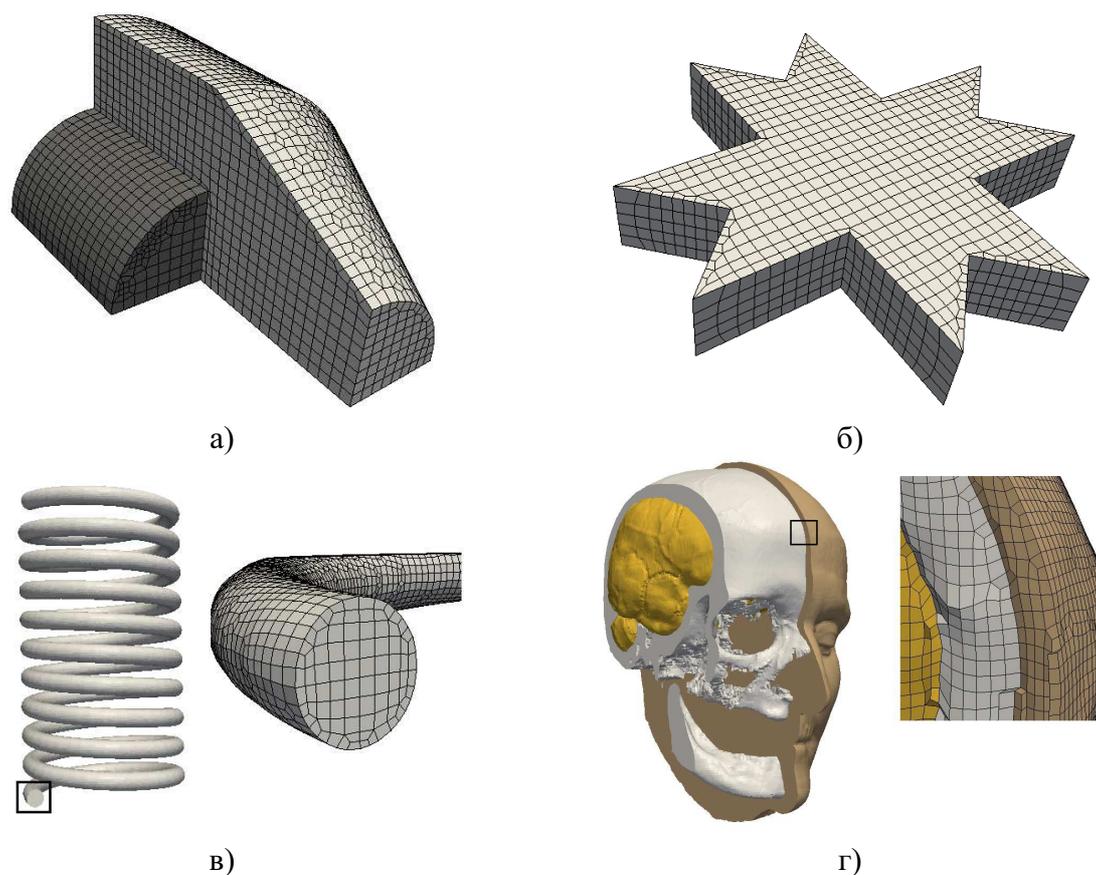


Рис. 3: Примеры построения неструктурированных шестигранных сеток: а) втулка с включением; б) звезда; в) пружина; г) голова человека с тремя материалами

Таблица 1: Меры качества сеток

| Параметры | Втулка | Звезда | Пружина | Голова человека |
|--------------|----------------------------|----------------------------|---------------------------|--------------------------------|
| Dim | $70 \times 129 \times 129$ | $129 \times 129 \times 28$ | $86 \times 151 \times 86$ | $1029 \times 1029 \times 1029$ |
| h | 4 | 4 | 1 | 2 |
| N | 15930 | 3894 | 239757 | 1363369 |
| C | 13842 | 2860 | 195904 | 1308341 |
| Θ_w | 73.6° | 61.1° | 54.3° | 37° |
| Q_J | 0.22 | 0.29 | 0.4 | 0.2 |
| Q_a | 33.8 | 20.7 | 4.2 | 15.9 |
| Q_s | 0.84 | 0.84 | 0.8 | 0.85 |
| Время (сек.) | 8 | 3 | 178 | 589 |
| Память (МБ) | 393 | 0.05 | 1526 | 6932 |

Время выполнения программ приведено без учета этапа предобработки, включающего работу по подбору шага h генерации сетки, заданию областей включений, удалению «зашумленных» областей томографических данных.

§ 5. Заключение

В данной работе представлена дальнейшая модификация воксельного генератора шестигранной сетки, применительно к моделям, описанным САД-геометрией в STL-формате. Разработаны следующие основные изменения в функционале генератора сеток.

- Получение из исходной STL-геометрии множества Λ «характерных» ребер и вершин, положение которых должны быть сохранены в результирующей шестигранной сетке \mathcal{T} . Множество Λ рассматривается как «каркас» модели, сохраняющий все ее острые вершины и грани.
- Построение и отображение части граничных узлов \mathcal{T} на множество Λ для обеспечения аппроксимации граничного контура. Процедура основана на последовательной проекции узлов каждого элемента Λ . Выбор узлов осуществляется в соответствии с предложенными критериями приоритета, включающими число соседних шестигранников, показатели качества ячеек и граней, а также расстояние до элементов Λ .
- Удаление вырожденных ячеек \mathcal{T} возникающих вдоль «характерных» ребер. Операция включает в себя вставку дополнительного поверхностного слоя шестигранников и разбиения вывернутых ячеек на четыре новых по шаблону, описанному в [7].

Алгоритм протестирован на достаточном количестве моделей различной сложности, в настоящий момент продолжают исследования по развитию метода для построения адаптивных шестигранных сеток.

Для случая воксельных данных построение сборочных (многокомпонентных) моделей осуществляется последовательной генерацией сеток для каждого значения изоповерхности (функции уровня) в порядке убывания. В случае поверхностного представления применение такого подхода представляется затруднительным, так как полученный из STL-геометрии файл «знакопределенных полей расстояний» (SDF) содержит всего одно значение функции уровня равное нулю. Применение одного файла формата SDF позволяет разрешить частный случай, когда модель содержит области с включениями находящимися полностью внутри, либо на одном уровне вдоль ортогональной плоскости (рис. 3, а). Для реализации общего случая построения многокомпонентных моделей из STL-геометрии (в т. ч. когда сходятся три и более материала) необходимо использовать набор SDF-файлов, каждый из которых сгенерирован для конкретного материала и выполнить последовательную сборку начальной ортогональной сетки по каждому SDF-файлу. На данный момент проводится работа по реализации описанной технологии в алгоритме.

СПИСОК ЛИТЕРАТУРЫ

1. Zhang Y., Bajaj C. Adaptive and quality quadrilateral/hexahedral meshing from volumetric data // Computer Methods in Applied Mechanics and Engineering. 2006. Vol. 195. Issues 9–12. P. 942–960. <https://doi.org/10.1016/j.cma.2005.02.016>
2. Pellerling J., Johnen A., Remacle J.-F. Identifying combinations of tetrahedra into hexahedra: a vertex based strategy // Procedia Engineering. 2017. Vol. 203. P. 2–13. <https://doi.org/10.1016/j.proeng.2017.09.779>

3. Awad M. A., Rushdi A. A., Abbas M. A., Mitchell S. A., Mahmoud A. H., Bajaj C. L., Ebeida M. S. All-hex meshing of multiple-region domains without cleanup // *Procedia Engineering*. 2016. Vol. 163. P. 251–261. <https://doi.org/10.1016/j.proeng.2016.11.055>
4. Ruiz-Gironés E., Roca X., Sarrate J. The receding front method applied to hexahedral mesh generation of exterior domains // *Engineering With Computers*. 2012. Vol. 28. Issue 4. P. 391–408. <https://doi.org/10.1007/s00366-011-0233-y>
5. Ledoux F., Weill J.-C. An extension of the reliable whisker weaving algorithm // *Proceedings of the 16th International Meshing Roundtable*. Seattle, Washington. 2007. P. 215–232. https://doi.org/10.1007/978-3-540-75103-8_13
6. Gao X., Jacob W., Tarini M., Panozzo D. Robust hex-dominant mesh generation using field-guided polyhedral agglomeration // *ACM Transactions on Graphics*. 2017. Vol. 36. No. 4. Article no. 114. <https://doi.org/10.1145/3072959.3073676>
7. Schneiders R., Bünten R. Automatic generation of hexahedral finite element meshes // *Computer Aided Geometric Design*. 1995. Vol. 12. No. 7. P. 693–707. [https://doi.org/10.1016/0167-8396\(95\)00013-v](https://doi.org/10.1016/0167-8396(95)00013-v)
8. Караваев А. С., Копысов С. П. Метод построения неструктурированных шестигранных сеток из объемных данных // *Компьютерные исследования и моделирование*. 2013. Т. 5. № 1. С. 11–24. <https://doi.org/10.20537/2076-7633-2013-5-1-11-24>
9. Cannan S., Tristano J., Staten M. An approach to combined Laplacian and optimization-based smoothing for triangular, quadrilateral, and quad-dominant meshes // *7th International Meshing Roundtable*. Dearborn, Michigan. 1998. P. 479–494.

Поступила в редакцию 08.05.2020

Караваев Александр Сергеевич, к. ф.-м. н., научный сотрудник, кафедра вычислительной механики, Удмуртский государственный университет, 426034, Россия, г. Ижевск, ул. Университетская, 1.
E-mail: karavaev-alexander@yandex.ru

Копысов Сергей Петрович, д. ф.-м. н., профессор, кафедра вычислительной механики, Удмуртский государственный университет, 426034, Россия, г. Ижевск, ул. Университетская, 1.
E-mail: s.kopysov@gmail.com

Цитирование: А. С. Караваев, С. П. Копысов. Модификация генератора шестигранных сеток, основанного на воксельном представлении геометрии // *Вестник Удмуртского университета. Математика. Механика. Компьютерные науки*. 2020. Т. 30. Вып. 3. С. 468–479.

A. S. Karavaev, S. P. Kopysov

A modification of the hexahedral mesh generator based on voxel geometry representation

Keywords: hexahedral mesh generator, volume data, CAD model, STL geometry.

MSC2010: 65M50

DOI: [10.35634/vm200308](https://doi.org/10.35634/vm200308)

We consider a modification of the previously developed voxel-based mesh algorithm to generate models given in STL-geometry format. Proposed hexahedral mesh generator belongs to the family of grid methods, and is general-purpose in terms of a capability to use as source data both volume (voxel) and STL-surface representation of model geometry. For now, the algorithm works with CAD models described in the well-known STL format. However, it also allows to handle higher-order surface patches defined in an arbitrary format if appropriate procedures for projection and intersection operations will be specified. To define the initial position of mesh nodes, a «signed distance field» volume data file, obtained from the STL-geometry, is used. A special projection technique was developed to adapt constructed orthogonal mesh on the model's boundary. It provides an approximation of sharp edges and corners and is performed before running any other operations with the mesh. Finally, to improve the quality of the mesh, additional procedures were implemented, including boundary layers insertion, bad quality cells splitting, and optimization-based smoothing technique. The algorithm has been tested on a sufficient number of models, some of which are given as examples.

REFERENCES

1. Zhang Y., Bajaj C. Adaptive and quality quadrilateral/hexahedral meshing from volumetric data, *Computer Methods in Applied Mechanics and Engineering*, 2006, vol. 195, issues 9–12, pp. 942–960. <https://doi.org/10.1016/j.cma.2005.02.016>
2. Pelling J., Johnen A., Remacle J.-F. Identifying combinations of tetrahedra into hexahedra: a vertex based strategy, *Procedia Engineering*, 2017, vol. 203, pp. 2–13. <https://doi.org/10.1016/j.proeng.2017.09.779>
3. Awad M. A., Rushdi A. A., Abbas M. A., Mitchell S. A., Mahmoud A. H., Bajaj C. L., Ebeida M. S. All-hex meshing of multiple-region domains without cleanup, *Procedia Engineering*, 2016, vol. 163, pp. 251–261. <https://doi.org/10.1016/j.proeng.2016.11.055>
4. Ruiz-Gironés E., Roca X., Sarrate J. The receding front method applied to hexahedral mesh generation of exterior domains, *Engineering With Computers*, 2012, vol. 28, issue 4, pp. 391–408. <https://doi.org/10.1007/s00366-011-0233-y>
5. Ledoux F., Weill J.-C. An extension of the reliable whisker weaving algorithm, *Proceedings of the 16th International Meshing Roundtable*, Seattle, Washington, 2007, pp. 215–232. https://doi.org/10.1007/978-3-540-75103-8_13
6. Gao X., Jacob W., Tarini M., Panozzo D. Robust hex-dominant mesh generation using field-guided polyhedral agglomeration, *ACM Transactions on Graphics*, 2017, vol. 36, no. 4, article no. 114. <https://doi.org/10.1145/3072959.3073676>
7. Schneiders R., Bünten R. Automatic generation of hexahedral finite element meshes, *Computer Aided Geometric Design*, 1995, vol. 12, no. 7, pp. 693–707. [https://doi.org/10.1016/0167-8396\(95\)00013-V](https://doi.org/10.1016/0167-8396(95)00013-V)
8. Karavaev A. S., Kopysov S. P. The method of unstructured hexahedral mesh generation from volumetric data, *Computer Research and Modeling*, 2013, vol. 5, no. 1, pp. 11–24 (in Russian). <https://doi.org/10.20537/2076-7633-2013-5-1-11-24>
9. Cannan S., Tristano J., Staten M. An approach to combined Laplacian and optimization-based smoothing for triangular, quadrilateral, and quad-dominant meshes, *7th International Meshing Roundtable*, Dearborn, Michigan, 1998, pp. 479–494.

Received 08.05.2020

Karavaev Aleksandr Sergeevich, Candidate of Physics and Mathematics, Researcher, Department of Computational Mechanics, Udmurt State University, ul. Universitetskaya, 1, Izhevsk, 426034, Russia.

E-mail: karavaev-alexander@yandex.ru

Kopysov Sergei Petrovich, Doctor of Physics and Mathematics, Professor, Department of Computational Mechanics, Udmurt State University, ul. Universitetskaya, 1, Izhevsk, 426034, Russia.

E-mail: s.kopysov@gmail.com

Citation: A. S. Karavaev, S. P. Kopysov. A modification of the hexahedral mesh generator based on voxel geometry representation, *Vestnik Udmurtskogo Universiteta. Matematika. Mekhanika. Komp'yuternye Nauki*, 2020, vol. 30, issue 3, pp. 468–479.