

УДК 519.6

DOI: 10.53403/97859515049372020_25.1_88_107

Автоматический генератор неструктурированных многогранных сеток в препроцессоре пакета программ ЛОГОС

Приводится описание автоматического генератора неструктурированных сеток, используемых при решении задач аэрогидромеханики в рамках пакета программ ЛОГОС. Рассматриваются этапы генерации объемной сетки, поэтапное построение сетки продемонстрировано на примере.

**Д. Н. Смолкина, О. Н. Борисенко,
М. В. Черенкова, А. Г. Гиниятуллина,
М. В. Кузьменко, Н. В. Чухманов,
Е. В. Потехина, Н. В. Попова,
М. Р. Турусов**

Введение

В связи с возрастающей ролью численного моделирования физических процессов все больше внимания уделяется упрощению процесса подготовки задач к расчету, наиболее сложным и ресурсоемким этапом которой является генерация сеток.

В настоящее время все более востребованы сетки, состоящие в основном из шестигранных ячеек. Существует два основных подхода к их генерации: построение на основе поверхностной четырехугольной сетки [1, 2] и построение, при котором такая сетка не требуется [3–6].

В методах [5, 6] построение начинается с заполнения области сеткой из шестигранников и удаления из нее ячеек, которые находятся вне области и которые располагаются вблизи границы. Затем граница полученной сетки из шестигранников отображается на поверхность. Таким образом, получается сетка на границе области. Данные методы относительно просты и отказоустойчивы. Однако, применяя их на практике, достаточно трудно описать топологически сложные области исходной модели (с высокой кривизной, острыми кромками).

В данной статье представлен метод автоматической генерации неструктурированных сеток с многогранными элементами на основе дробной сетки [7]. В первую очередь генератор ориентирован на построение сеток для решения задач аэрогидромеханики в рамках программного пакета ЛОГОС [8, 9].

Среди зарубежных аналогов, использующих похожий подход к построению сеток, можно выделить программные комплексы Star-CCM+ [10], GAMBIT [11], OpenFOAM [12].

Структура и этапы построения сетки

Ячейки объемной сетки, используемой для численного моделирования процессов аэрогидромеханики [13], подразделяются преимущественно на две категории: те, которые находятся вблизи моделируемого объекта, и те, которые заполняют остальной объем.

Неоднородная структура сетки является особенностью данного генератора. Вдали от границ модели ячейки сетки (*дробные*) имеют форму куба, но могут состоять и более чем из шести граней. Вблизи границы модели формируются ячейки в виде многогранных призм. Ячейки на стыке дробных ячеек и призм являются многогранниками произвольной формы. Далее эти ячейки будут называться *отсеченными*, а процесс их построения – *отсечением*. На рис. 1 приведен плоский срез сетки с обозначениями ячеек.

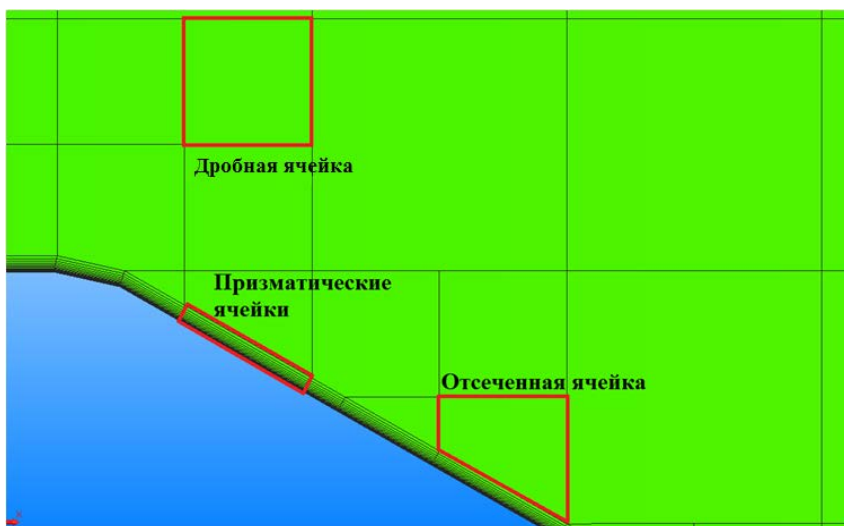


Рис. 1. Плоский срез сетки

Исходными данными для генератора является поверхностная сетка, состоящая из треугольных ячеек (для краткости будем называть их просто треугольниками). Она аппроксимирует поверхность геометрической модели с учетом кривизны поверхностей и характерных кривых. Характерные кривые – это ломаные, проходящие по особенностям геометрической модели (негладкие места стыковки поверхностей, острые ребра геометрии).

Перед началом работы генератора поверхностная сетка проверяется на наличие в ней открытых контуров и пересечений. В случае их обнаружения построение объемной сетки не выполняется.

Процесс генерации объемной неструктурированной сетки состоит из следующих этапов:

- определение/получение характерных кривых (особенностей) модели;
- построение поверхностной сетки из треугольников, смещенной относительно исходной;
- построение дробной сетки;
- отсечение дробных ячеек треугольниками смещенной поверхностной сетки;
- улучшение качества отсеченных ячеек;

- построение призматических ячеек;
- анализ качества ячеек.

Рассмотрим подробно каждый из этих этапов.

Определение характерных особенностей модели

Для описания острых кромок и/или деталей поверхности, которые необходимо сохранить при построении объемной сетки, в генераторе реализован алгоритм определения характерных кривых модели. Некоторые подходы к выявлению характерных особенностей поверхностной сетки описаны в [14–16].

В настоящее время в генераторе при построении сетки учитываются три типа характерных кривых: характерные кривые модели, сформированные в графическом интерфейсе (до запуска генератора), границы выделенных областей, а также автоматически определенные характерные особенности геометрии. Далее будет рассмотрен алгоритм автоматического определения характерных особенностей модели, основанный на методе, предложенном в [14].

Основным параметром для выявления участков характерных кривых является угол между нормальными к плоскостям треугольников поверхностной сетки, имеющих общее ребро. Если этот угол превышает заданное значение (по умолчанию оно составляет 44°), то считается, что общее ребро треугольников является участком характерной кривой (рис. 2,а).

Еще одним параметром для определения характерных кривых является *дефицит угла* в узлах сетки на поверхности модели. Дефицит угла d узла v вычисляется как $d(v) = 2\pi - \sum_i \alpha_i$,

где $\alpha_i(e_i, e_{i+1})$ – угол между соседними ребрами треугольников, сходящихся в узле v (рис. 2,б).

Дефицит угла близок по смыслу к гауссовой кривизне, а его значение показывает отличие поверхности в рассматриваемой точке от плоскости. Если рассчитанный для узла сетки дефицит угла превышает заданное предельное значение (по умолчанию 72°), то такой узел считается особенностью и будет учитываться при определении характерных кривых.

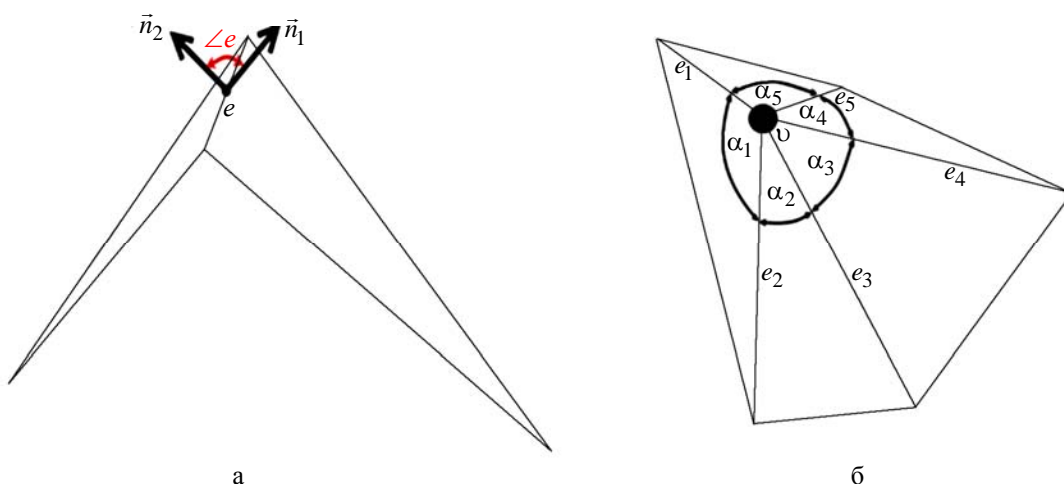


Рис. 2. Иллюстрация параметров для определения характерных кривых: а – угол между нормальными к плоскостям треугольников; б – углы между ребрами треугольников для вычисления дефицита угла узла v

На основании вышеуказанных критериев выполняется формирование списков участков характерных кривых (ребер) и узлов, для которых выполнено условие дефицита угла. Сформированный список ребер (список 1) будет использоваться для построения характерных кривых, а список узлов (список 2) – для проверки выполнения одного из приведенных ниже критериев фильтрации.

Алгоритм формирования характерной кривой заключается в последовательном добавлении в кривую ребер из списка 1, пока не будет выполнено одно из условий:

– узел, который является концом характерной кривой, имеет дефицит угла, т. е. принадлежит списку 2;

– кривая замкнулась;

– нет следующего ребра, выходящего из узла.

Результат выполнения алгоритма – набор характерных кривых, образованных ребрами треугольников поверхностной сетки. Среди этих кривых могут быть так называемые *ложные*, которые удовлетворяют совокупности следующих критериев фильтрации:

– кривая состоит из небольшого количества ребер;

– кривая не содержит узлов с дефицитом угла;

– кривая не является замкнутой.

Ложные кривые удаляются.

На рис. 3 жирными линиями показаны *ложные* кривые, подлежащие удалению.

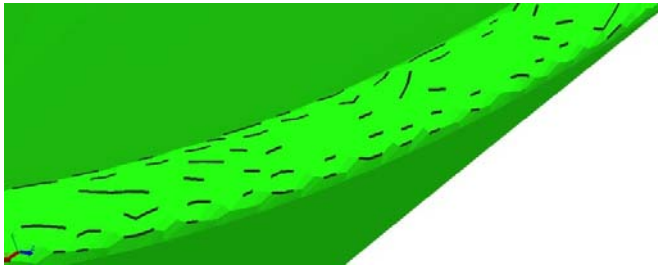


Рис. 3. Удаляемые ложные кривые

Результат работы алгоритма определения характерных кривых представлен на примере крыла самолета DLR F6 (рис. 4).

В дальнейшем будем различать два типа узлов, принадлежащих характерной кривой: обычные и особые. Особый узел – это узел, в котором сходятся две или более характерных кривых. По умолчанию речь идет об обычном узле.



Рис. 4. Крыло самолета DLR F6 с выявленными характерными кривыми (выделены жирными линиями)

Построение смещенной поверхности

После определения характерных кривых выполняется построение поверхностной сетки из треугольников, смещенной относительно исходной, – так называемой *смещенной поверхности*. Методы построения смещенной поверхности, учитывающие характерные особенности модели, при генерации объемных сеток с призматическими ячейками представлены в работах [17, 18]. Построение смещенной поверхности в данной работе основывается на геометрическом методе и состоит из двух основных этапов:

- 1) определение оптимального направления сдвига в каждом узле сетки;
- 2) определение максимально возможного, не превышающего заданного, смещения* вдоль направления сдвига.

Набор треугольников с нулевым смещением будем называть неподвижной границей, набор с ненулевым смещением – подвижной границей (рис. 5).

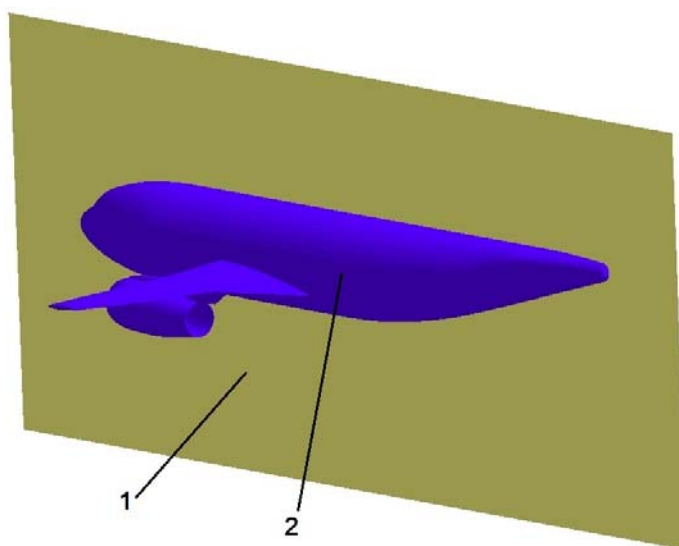


Рис. 5. Неподвижная (1) и подвижная (2) границы

Направление сдвига для узла на гладкой поверхности подвижной границы определяется по формуле

$$\vec{n}_i = \frac{\sum_j \vec{n}_{ij} \alpha_{ij} d_{ij}}{\left[\sum_j \vec{n}_{ij} \alpha_{ij} d_{ij} \right]},$$

где N_i – количество треугольников, примыкающих к i -му узлу; α_{ij} – угол в j -м треугольнике при i -м узле, d_{ij} – заданное смещение j -го треугольника, \vec{n}_{ij} – нормаль плоскости j -го треугольника.

* Под смещением узла или треугольника здесь и далее подразумевается скалярная величина – расстояние между смещенным и исходным положением этого узла или треугольника.

Если узел принадлежит ребру характерной кривой, то формируется двугранный угол между «средними плоскостями» треугольников, примыкающих к этому ребру с двух сторон. Сдвиг рассматриваемого узла направлен вдоль биссекторной плоскости двугранного угла и определяется по формуле

$$\vec{n}_i = \frac{\frac{\vec{n}_1 + \vec{n}_2}{1 + \vec{n}_1 \vec{n}_2}}{\left[\frac{\vec{n}_1 + \vec{n}_2}{1 + \vec{n}_1 \vec{n}_2} \right]},$$

где \vec{n}_1, \vec{n}_2 – нормали к плоскостям двугранного угла.

Если узел принадлежит треугольникам как подвижной, так и неподвижной границы, то он объявляется граничным. Направление сдвига граничного узла корректируется по формуле

$$\vec{n}_{i_bord} = \vec{n}_i - \vec{n}_0 (\vec{n}_i \vec{n}_0),$$

где в скобках – скалярное произведение векторов, \vec{n}_i – направление сдвига вдоль биссекторной плоскости, \vec{n}_0 – средняя нормаль к плоскостям треугольников неподвижной границы.

Если узел является особым узлом характерной кривой, то для него строится многогранный угол, ребра которого являются ребрами характерных кривых, сходящихся в рассматриваемом узле. В многогранный угол вписывается шар единичного радиуса таким образом, чтобы он оказался как можно ближе к узлу. Сдвиг особого узла направлен вдоль линии, соединяющей узел и центр шара. Задача оптимального расположения шара в многогранном угле сводится к канонической задаче квадратичного программирования.

После вычисления направлений сдвига проводится их коррекция, которая осуществляется итерационно методом подвижного фронта, начиная от узла характерных кривых, далее вдоль ребер, а затем вглубь гладких поверхностей.

Чтобы избежать нежелательных деформаций геометрической модели, коррекции подвергаются только направления сдвига в узлах гладких поверхностей. При их коррекции используется метод Гаусса – Зейделя: направление сдвига узла находится как нормализованная сумма направлений сдвигов соседних узлов, взятых с соответствующими коэффициентами [19]:

$$\vec{n}'_i = \vec{n}_i (1 - \alpha) + \frac{\sum_j \vec{n}_{ij} \omega_{ij}}{\sum_j \omega_{ij}} \alpha. \quad (1)$$

Для достижения достаточной гладкости поверхности оптимальным считается выбор весового коэффициента $\alpha = 0,3$. При коррекции нормали в i -м узле весовой коэффициент ω_{ij} для j -го узла, соединенного ребром с i -м, рассчитывается по формуле

$$\omega_{ij} = \frac{\operatorname{tg} \frac{\alpha_{j+1}}{2} + \operatorname{tg} \frac{\alpha_j}{2}}{l_{ij}}, \quad (2)$$

где l_{ij} – длина ребра, соединяющего i -й и j -й узлы; α_{j+1} и α_j – углы при узле j , примыкающие к этому ребру.

В особых узлах характерных кривых направления сдвигов не корректируются.

Если узел принадлежит ребру характерной кривой, то при коррекции рассматриваются только два соседних узла, также лежащие на этой кривой:

$$\vec{n}'_i = \vec{n}_1 \frac{l_2}{l_1 + l_2} + \vec{n}_2 \frac{l_1}{l_1 + l_2}, \quad (3)$$

где \vec{n}_1 и \vec{n}_2 – нормали в узлах, соседних для i -го на характерной кривой; l_1 и l_2 – длины ребер характерной кривой, соединяющих эти узлы с узлом i .

На следующем шаге итерационно определяются узлы с некорректными направлениями сдвига, заданные смещения вдоль которых приводят к развороту нормалей к плоскостям треугольников на 90° . Для каждого такого узла (он находится в невыпуклой области модели) выделяется окрестность радиусом, не превышающим смещения данного узла. Некорректное направление сдвига узла изменяется путем осреднения с направлениями сдвига узлов в выделенной окрестности.

На втором этапе определяются смещения вдоль рассчитанных направлений сдвига. Смещение i -го узла рассчитывается по формуле

$$d_i = \min \left\{ \frac{d_{ij}}{\vec{n}_{ij} \vec{l}_i} \right\}, \quad (4)$$

где \vec{l}_i – направление сдвига i -го узла; d_{ij} – заданное смещение примыкающего к i -му узлу треугольника, \vec{n}_{ij} – нормаль к плоскости j -го треугольника.

Для каждого j -го треугольника исходной поверхностной сетки рассчитывается допустимое смещение \tilde{d}_{ij} , при котором треугольник может повернуться не более чем на 90° по отношению к исходному положению. Исходя из этого значения корректируются смещения в узлах треугольников по формуле

$$d'_i = \min \left\{ d_i, 0,9 \frac{\tilde{d}_{ij}}{\vec{n}_{ij} \vec{l}_i} \right\}, \quad (5)$$

где d'_i , d_i – новое и исходное смещения i -го узла j -го треугольника; \vec{l}_i – направление сдвига i -го узла; \tilde{d}_{ij} – допустимое смещение j -го треугольника, \vec{n}_{ij} – нормаль к плоскости j -го треугольника.

Практика показала, что использование формулы (5) для расчета смещения узлов, расположенных в острых углах исходной модели, приводит к снижению качества смещенной поверхности: не сохраняется форма исходной поверхности, на характерных кривых появляются узкие вытянутые треугольники. Поэтому в узлах характерной кривой, лежащих на общем ребре треугольников, двугранный угол между которыми не превышает 30° , смещение устанавливается равным $1/3$ от заданного значения (рис. 6).

Исходя из рассчитанных смещений и направлений сдвига вычисляются положения узлов смещенной поверхности.

Далее проверяется сохранение формы исходной поверхности. С этой целью для каждого треугольника вычисляется допустимый угол между нормальными к его плоскостям в исходном и смещенном положениях. Этот угол плавно меняется от 17° в областях, близких к характерным особенностям модели, до 60° в областях, удаленных от характерных кривых. Если угол превышает допустимое значение, то считается, что для смещенной поверхности рассматриваемого треугольника с номером j требуется приближение к исходному положению, т. е. уменьшение смещения.

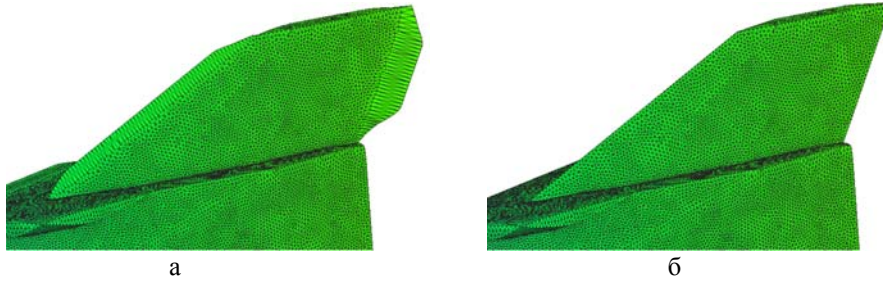


Рис. 6. Фрагмент смещенной поверхности: а – с расчетом сдвига по формуле (4), б – со скорректированным смещением

Для этого j -го треугольника находится узел с наименьшим смещением d_{\min} . Для двух других узлов треугольника смещения корректируются по формуле

$$d'_i = \min \left\{ d_i, \frac{1}{2} \left(d_i + \frac{d_{\min}}{\bar{n}_{ij} \bar{l}_i} \right) \right\},$$

где \bar{l}_i – направление сдвига i -го узла; d'_i , d_i – новое и исходное смещения i -го узла j -го треугольника; \bar{n}_{ij} – нормаль к плоскости j -го треугольника.

Если хотя бы для одного треугольника проводилась коррекция смещения, то она выполняется и для всей поверхности методом подвижного фронта при продвижении от характерных кривых вглубь области. При этом используется метод Гаусса – Зейделя с весовыми коэффициентами, аналогичными приведенным в формулах (1)–(3) для коррекции направлений сдвига. Новое смещение i -го узла вычисляется по формуле

$$d'_i = \min \left\{ d_i, d_i (1 - d_{\max_i}) + d_{i_smooth} d_{\max_i} \right\},$$

где d_{i_smooth} – усредненное в результате коррекции смещение, d_{\max_i} – максимальная относительная разница смещения i -го узла до и после коррекции.

В процессе построения могут возникнуть пересечения исходной и смещенной поверхностей или самопересечения. В этом случае также выполняется приближение смещенной поверхности к исходной. Уменьшение смещения осуществляется не для всей смещенной поверхности, а только для узлов треугольников, которые были детектированы как пересекающиеся. При этом смещение сокращается в N_{decr} раз:

$$N_{\text{decr}} = 0,7 - 0,001 N_{\text{iter}},$$

где N_{iter} – номер текущей итерации

Формирование и коррекция формы смещенной поверхности, проверка на самопересечение и пересечение и при необходимости приближения смещенной поверхности к исходной осуществляются итерационно до тех пор, пока не перестанут детектироваться пересечения.

Построение дробной сетки

Следующим этапом генерации является построение дробной сетки [20]. Параметрами для построения являются максимальный размер ячейки и размер дробной ячейки у поверхности.

Дополнительные условия определяются размерами дробных ячеек для контрольных областей. Размер любой ячейки дробной сетки, попавшей в контрольную область полностью или частично, не может быть больше, чем заданный для этой области. В настоящее время могут быть контрольные области следующих типов: прямоугольный параллелепипед, сфера, конус, цилиндр, а также их тонкостенные варианты (рис. 7, 8).

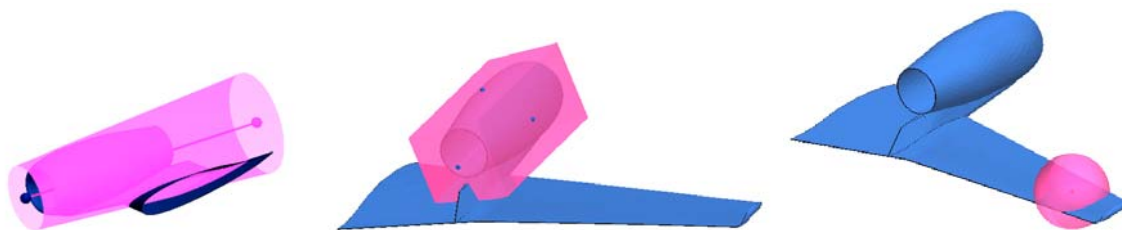


Рис. 7. Примеры контрольных областей (объекты с прозрачностью)

Алгоритм построения дробной сетки можно разделить на несколько этапов. На первом этапе на основе габаритов поверхностной сетки выполняется построение объемных ячеек максимального размера. После этого проводится распределение треугольников поверхностной сетки по ячейкам объемной сетки. Все объемные ячейки, в которые попал хотя бы один треугольник, должны быть отмечены, поскольку они будут использоваться на этапе отсечения треугольниками поверхностной сетки ячеек дробной сетки. Чтобы выявить ячейки, не принадлежащие области моделирования, для остальных ячеек определяются их положения относительно поверхностной сетки.

Задача определения положения объемной ячейки относительно треугольной сетки решается с использованием следующего алгоритма. Из центра ячейки параллельно одной из осей координат выпускается луч до пересечения с треугольниками. Если такой треугольник найден, анализируется косинус угла между нормалью к плоскости треугольника и лучом. Если косинус меньше 0, то это означает, что ячейка находится внутри модели, ограниченной поверхностной сеткой, если больше 0 – снаружи. Если нормаль к плоскости треугольника ортогональна лучу либо точка пересечения луча и треугольника совпадает с узлом или попадает на ребро треугольника, то такой случай считается особым и не рассматривается. При этом изменяется направление, по которому строится луч (выбирается другая ось координат), после чего процедура поиска пересечения луча и треугольников повторяется.

На следующем этапе итерационно проводится дробление ячеек, содержащих треугольники. Цель разбиения – добиться требуемого размера дробной ячейки вблизи поверхности, который вычисляется по формуле

$$S_{ad} = \frac{S_{reg}}{2^L}, \quad L = \left\lceil \log_2 \frac{S_{reg}}{\left(\sum_{i=1}^n w_i \right) / n} \right\rceil, \quad (6)$$

где L – требуемый уровень дробления ячейки (≤ 20), S_{reg} – размер (длина ребра) ячейки, $w_i = \sqrt{S_i}$ – «вес» i -го треугольника, n – число треугольников в дробной ячейке; запись $\lceil \]$ означает целую часть числа.

В каждой итерации дробления участвуют только те ячейки и подъячейки, в которые попали треугольники. Проводится перераспределение треугольников по восьми новым подъячейкам. Так как в общем случае количество треугольников, попавших в ячейку, изменилось, вычисляется тре-

буемый уровень разбиения по второй формуле (6). Если текущий уровень дробления ячейки меньше требуемого, то выполняется следующая итерация.

После завершения дробления ячейки определяются положения ее дробных подъячеек, не содержащих треугольники, относительно треугольной сетки и выполняется обработка контрольных областей (все дробные ячейки, принадлежащие области, дополнительно дробятся до заданного для области размера).

Заключительным шагом построения дробной сетки является балансировка объемов соседних ячеек: они должны отличаться не более чем в 8 раз; ребра соседних ячеек должны отличаться не более чем в 2 раза. Анализ ячеек начинается от треугольников поверхности, далее делается проход по соседним ячейкам внутрь сетки. Если объем ячейки не удовлетворяет указанным выше условиям, проводится дополнительное ее дробление.

На рис. 8 приведены примеры построенных сеток с использованием контрольных областей в виде тонкостенных фигур (для наглядности представлены сечения плоскостями).

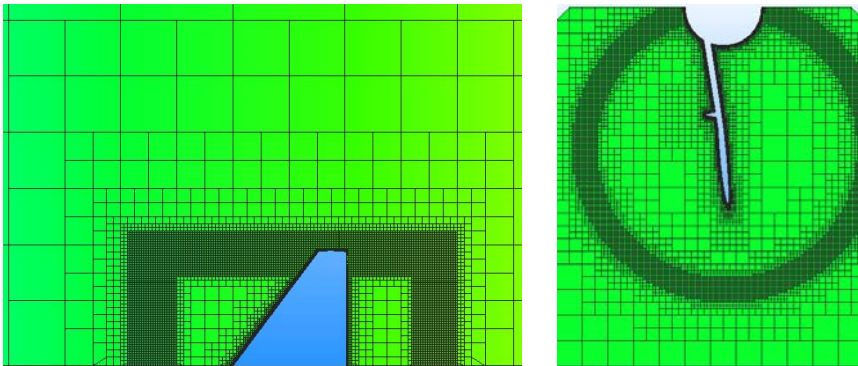


Рис. 8. Примеры сеток с контрольными областями в виде тонкостенных фигур

Отсечение дробных ячеек треугольниками поверхностной сетки

Начальными данными для этого этапа [21–26] являются: поверхностная сетка S из треугольников, дробная сетка C , а также данные, приписанные каждой ячейке сетки C , – набор треугольников и описание характерных кривых. Результатом будет многогранная сетка P , полученная в результате отсечения дробной сетки треугольниками исходной поверхностной сетки.

Для построения сетки P проводятся следующие операции:

- определение положения узлов сетки C относительно сетки S (внутри, снаружи или на поверхности модели). Краткий обзор применяемых методов можно найти в [27];
- нахождение пересечений ребер дробной сетки с треугольниками поверхностной сетки;
- выполнение операции объединения найденных точек пересечения друг с другом, если расстояние между точками мало ($< 10^{-6}$ м);
- построение внутренних граней многогранной сетки P на основе графовых алгоритмов, подробно описанных в [28];
- построение внешних граней многогранной сетки P .

Построенная сетка P обладает следующими свойствами: каждое ребро в ячейке разделяет только две грани; грани ячейки могут быть как выпуклыми, так и невыпуклыми; ячейки могут быть как выпуклыми, так и невыпуклыми.

В областях поверхностной сетки, не содержащих характерных кривых, будут построены многогранниками простой формы (рис. 9). В областях, содержащих характерные кривые, напротив, могут быть построены многогранные ячейки достаточно сложной формы (рис. 10).

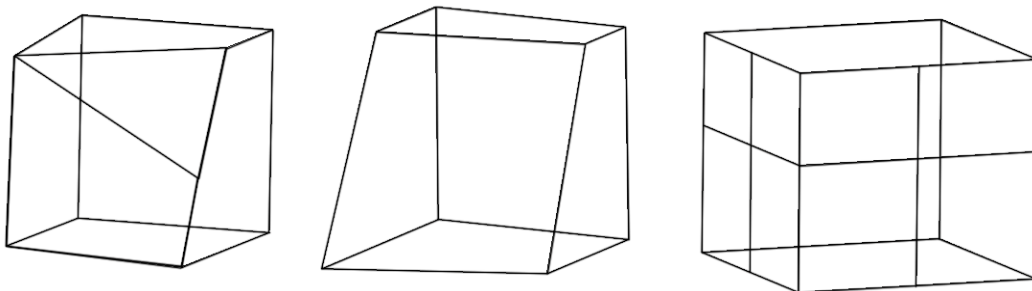


Рис. 9. Примеры многогранных ячеек простой формы

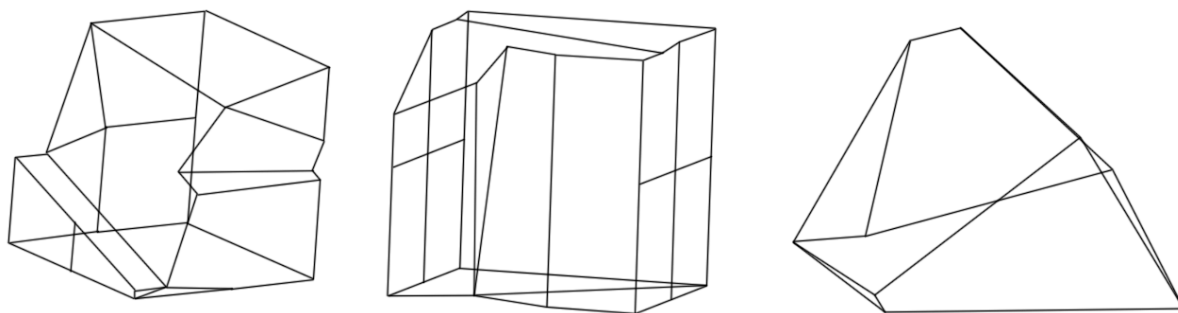


Рис. 10. Примеры невыпуклых ячеек

Особенностью применяемого подхода к отсечению дробных ячеек является явный поиск пересечений дробной сетки и треугольников смещенной поверхности, что позволяет более подробно описать все характерные особенности исходной модели (углы, щели и т. п.) в отличие от широко известного алгоритма *марширующих кубов* [29]. Однако следует отметить, что именно этот факт значительно повышает вычислительную сложность алгоритма.

Улучшение качества отсеченных ячеек

После этапа отсечения ячеек проводится улучшение качества ячеек сетки [30]: разрезание невыпуклых ячеек, слияние ячеек малого объема с соседними ячейками, объединение поверхностных граней ячейки, разрезание невыпуклых граней.

Разрезание невыпуклых ячеек. В процессе отсечения могут быть сформированы невыпуклые ячейки, способные в дальнейшем негативно повлиять на численное моделирование. Для решения данной проблемы реализован алгоритм их разрезания. Алгоритм разработан с учетом условий минимизации добавления новых узлов в сетку и оптимизации распределения объемов между новыми ячейками.

Разрезание проводится по биссектрисе угла между двумя поверхностными (аппроксимирующими смещенную поверхность) гранями, образующими ребро невыпуклости, или по плоскостям

этих граней. В качестве оптимальной плоскости разрезания выбирается плоскость с наибольшим значением коэффициента качества потенциальных ячеек, определяемого по формуле

$$K = 1 - \frac{N_{п.у}}{N_{р.г}} + \frac{S_1}{S_2},$$

где $N_{п.у}$ – количество «притянутых» узлов ячейки, т. е. не лежащих в плоскости разреза, но близких к ней; $N_{р.г}$ – количество разрезанных граней исходной ячейки; S_1 , S_2 – площади неразрезанных граней, лежащих по разные стороны от плоскости разреза.

На рис. 11 на примере показано выполнение алгоритма разрезания невыпуклых ячеек. На первом этапе проводится разрезание по ребру невыпуклости JQ , после которого остаются многогранники $PJQION$ (выпуклый) и $ABCRDEFGHIQJL$ (невыпуклый). На втором этапе происходит разрезание по ребру невыпуклости DJ . Исходя из условий оптимального соотношения объемов плоскость разреза проходит по плоскости грани $DRQJ$. В итоге образуется выпуклый многогранник $SBCRDTHIQJ$ и невыпуклый $ASDEFGTJLM$, при этом добавлены новые узлы S и T . На последнем этапе выполняется разрезание по ребру невыпуклости EL . Плоскость разреза проходит по плоскости грани $ELMF$.

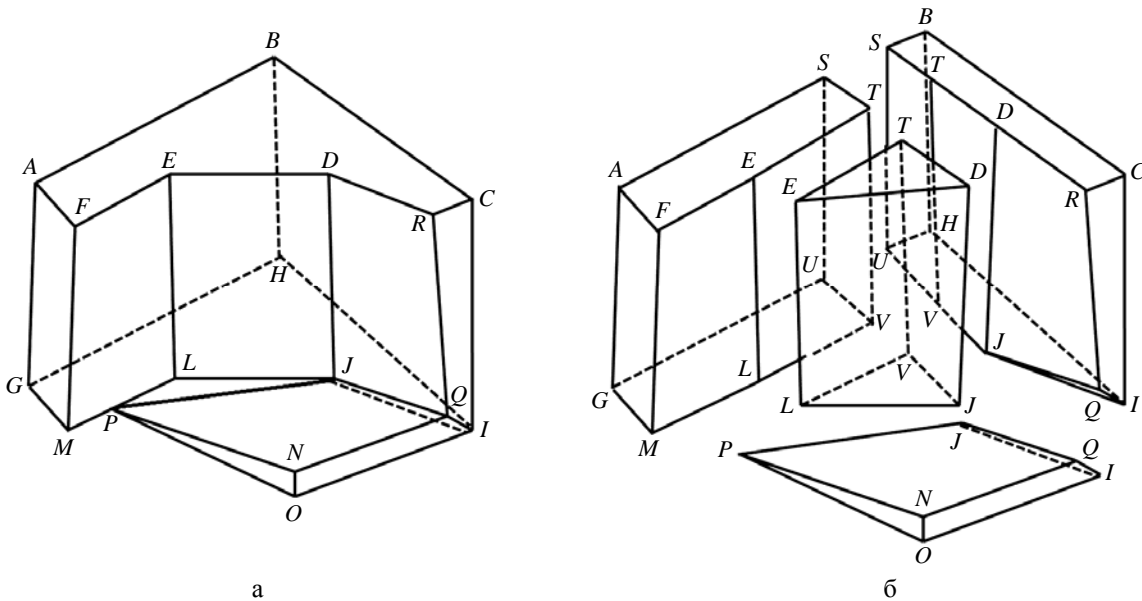


Рис. 11. Пример выполнения алгоритма разрезания невыпуклых ячеек: а – исходная ячейка; б – совокупность ячеек после разрезания

Слияние ячеек малого объема с соседними ячейками. Для поддержания равномерности объемов отсеченных ячеек разработана операция объединения ячеек, имеющих малый объем по сравнению с соседними ячейками. На рис. 12 приведен один из возможных вариантов расположения ячеек сетки, когда такая операция необходима.

Пусть дана многогранная отсеченная ячейка малого объема и известны ее соседние ячейки. Нужно из множества всех соседних ячеек выбрать такую, чтобы в результате объединения была образована выпуклая ячейка и разница объемов между ячейками на данном участке сетки была минимальной.

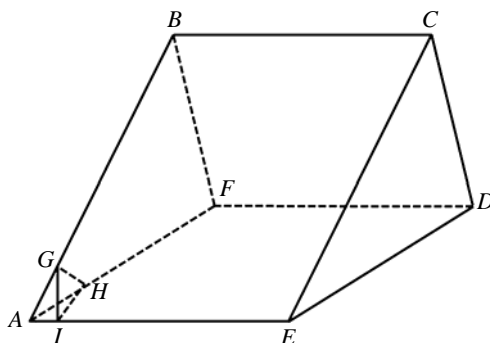


Рис. 12. Пример необходимости слияния ячейки малого объема ($AGIH$) с соседней ячейкой ($GBCEIHF$)

Для определения наиболее подходящего варианта объединения из всех соседей выбирается ячейка, имеющая большую площадь грани, общую с малой ячейкой, больший объем и меньшее число поверхностных граней.

Объединение поверхностных граней ячейки. Поскольку на основе поверхностных граней отсеченных ячеек формируются призматические ячейки, необходимо исключить значительную разницу между площадями соседних граней. Одним из способов решения данной проблемы в рамках одной отсеченной ячейки является операция объединения граней.

На рис. 13,*а* показана исходная ячейка, а на рис. 13,*б* – ячейка с объединенными поверхностными гранями. В результате выполнения алгоритма поверхностные грани $ABEF$ и $BCDE$ объединены в грань $ABCDEF$.

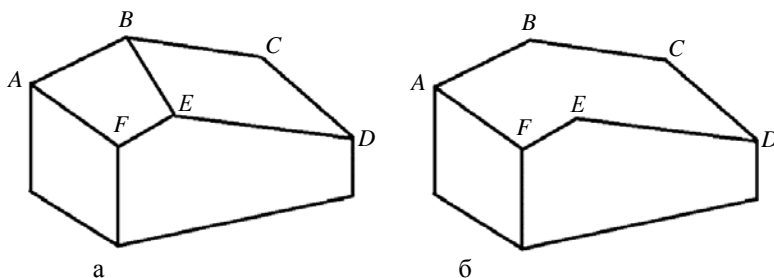


Рис. 13. Пример выполнения алгоритма объединения поверхностных граней: а – исходная ячейка; б – отсеченная ячейка с объединенными поверхностными гранями

Сначала в ячейке проводится поиск поверхностных граней с сильно различающимися площадями. Если такие грани присутствуют в ячейке и ребро, их разделяющее, не принадлежит характерной кривой, то эти грани будут объединяться. Далее все грани ячейки сортируются в зависимости от их площади и определяются все варианты их объединения (сочетания без повторов). После этого выбирается наиболее благоприятный вариант из всех возможных. Требования, предъявляемые к наилучшему варианту объединения, – выпуклость, минимальная разница периметров полученных граней.

Разрезание невыпуклых граней. Одним из условий получения призматической ячейки приемлемого качества является ее формирование исходя из выпуклой грани отсеченной ячейки. С этой целью добавлена возможность разрезания невыпуклых граней. В генераторе реализованы три алгоритма разрезания грани в зависимости от количества и расположения на грани *узлов*

невыпуклости. На рис. 14 представлены примеры ячеек для каждого типа разрезания с выделенными узлами невыпуклости.

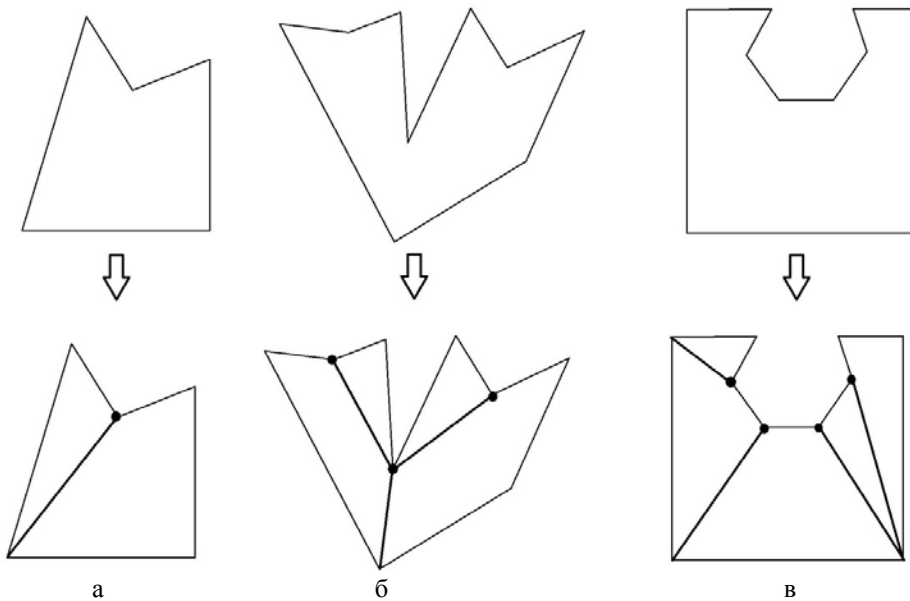


Рис. 14. Пример разрезания невыпуклых граней: а – одноточечное разрезание; б – многоточечное разрезание; в – разрезание по секторам; сверху – грани до разрезания, снизу – грани после разрезания

Одноточечное разрезание грани (см. рис. 14,а) выполняется в том случае, если грань имеет только один узел невыпуклости. С помощью графовых алгоритмов находят все возможные варианты разрезания грани. Из полученных вариантов удаляются комбинации, содержащие невыпуклые и вырожденные многоугольники. Из оставшихся выбирается комбинация с минимальным соотношением периметров.

Многоточечное разрезание грани (см. рис. 14,б) используется в случае, если грань имеет несколько непоследовательно расположенных узлов невыпуклости. Соединяя такие узлы по направлению обхода грани и против него, выбираем грань с минимальной мерой качества

$$M = N + \frac{L}{P},$$

где N – количество узлов новой грани, не лежащих на одной прямой; L – длина нового ребра отрезаемой части; P – периметр отрезаемой части.

Разрезание по секторам (см. рис. 14,в) используется, если грань имеет несколько последовательных узлов невыпуклости. Сектор разреза определяется пересечением полупространств, образованных ребрами, исходящими из узла невыпуклости.

Грань формируется по узлу невыпуклости и узлу, найденному следующим способом. Если в сектор разреза не попал ни один узел грани, то берем ближайший узел в направлении обхода. Если в сектор попал один узел, то берем его. Если в сектор разреза попало несколько узлов, то выбираем узел с минимальной мерой качества, которая определяется по формуле

$$M = \left| S_{loc} - \frac{S_{face}}{N_{SP} + 1} \right| \frac{1}{S_{face}} + \begin{cases} 0, & \text{если в новой грани 4 узла;} \\ 0,5, & \text{если в новой грани 3 узла;} \\ 10 & \text{в любом другом случае;} \end{cases} + \begin{cases} 0, & \text{если новая грань выпуклая;} \\ 100, & \text{если новая грань невыпуклая.} \end{cases}$$

Здесь S_{loc} – площадь новой грани, S_{face} – площадь исходной грани, N_{SP} – количество узлов невыпуклости в исходной грани.

Построение призматических ячеек

Внешние грани ячеек объемной сетки, построенной на этапе отсечения, образуют поверхность сетку, которая используется для построения призматических ячеек.

Для формирования призматических ячеек вычисляются барицентрические координаты узлов поверхностных граней отсеченных ячеек в треугольниках смещенной поверхности. С использованием этих координат узлы со смещенной поверхности переносятся на исходную поверхность. Далее по заданным параметрам (коэффициент сгущения, количество слоев) строится набор ячеек.

В случае близкого расположения исходной и смещенной поверхностей осуществляется «схлопывание» ребер ячеек и изменение количества слоев. Это позволяет избежать создания малых ребер при формировании призматических ячеек.

Анализ качества

В генераторе реализована проверка качества отсеченных и призматических ячеек [31]. Качество для каждого типа ячеек анализируется непосредственно после выполнения соответствующего этапа построения сетки (отсечения, формирования призматических ячеек) по критериям, учитывающим: наличие замкнутых ячеек и свободных ребер, объемы ячеек и соотношения объемов соседних ячеек, скошенность поверхностных граней, соотношение длин ребер, количество узлов поверхностных граней, количество невыпуклых граней, соотношение площадей поверхностных граней.

Пример построения сеточной модели автоматическим генератором

Построение объемной многогранной сетки продемонстрировано на примере двухзвенного профиля крыла самолета. Общий вид геометрии и зона вблизи области моделирования изображены на рис. 15. На рис. 16–20 показаны этапы построения сеточной модели.

Количество ячеек составляет 3,3 млн. Время расчета сетки – 4 минуты (Intel Core i7, 2,79 ГГц, 8 процессоров).

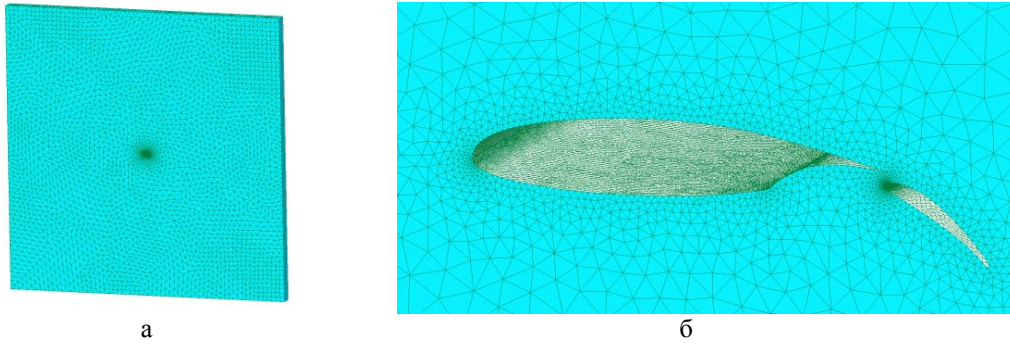


Рис. 15. Общий вид исходной поверхностной сетки для двухзвенного профиля крыла самолета (а) и зона моделирования в увеличенном масштабе (б)

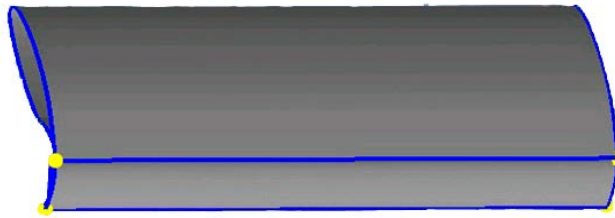


Рис. 16. Построение характерных кривых и вершин

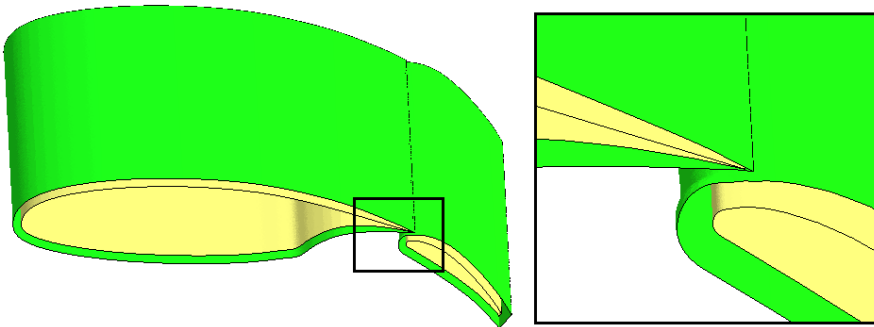


Рис. 17. Построение смещенной поверхности

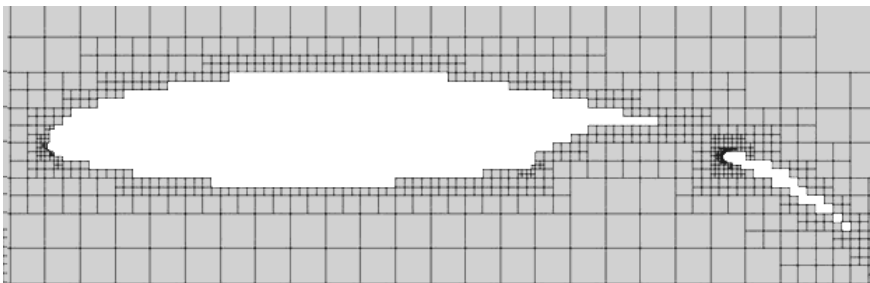


Рис. 18. Построение дробной сетки

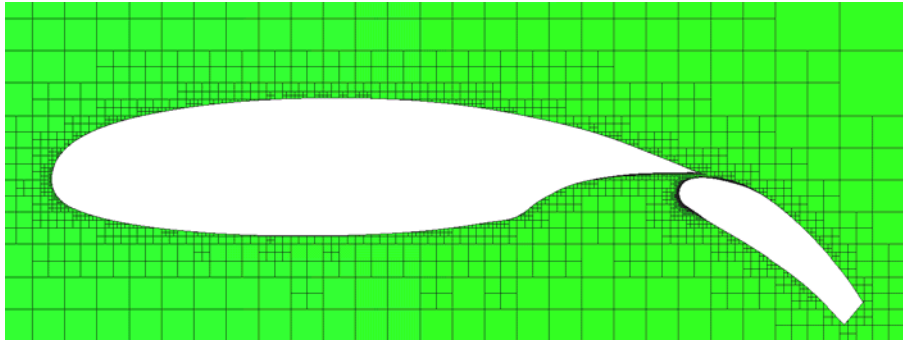


Рис. 19. Отсечение дробной сетки треугольниками поверхностной сетки

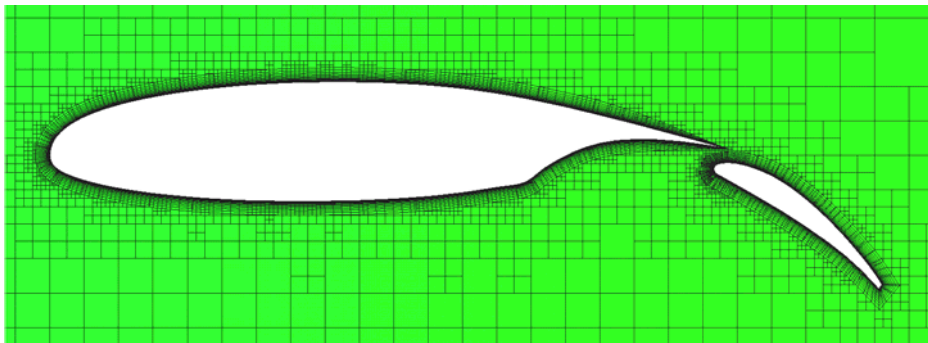


Рис. 20. Построение сетки с призматическими ячейками

Заключение

Рассмотрены основные этапы работы генератора неструктурированных многогранных сеток, реализованного в препроцессоре программного комплекса ЛОГОС. Приведены результаты построения сеточной модели для двухзвенного профиля крыла самолета.

Список литературы

1. Blacker T. D., Meyers R. J. Seams and wedges in plastering: a 3D hexahedral mesh generation algorithm // Engineering with computers. 1993. Vol. 9(2). P. 83–93.
2. Tautges T. J., Timothy J., Mitchell S. A. Progress report on the whisker weaving all-hexahedral meshing algorithm // Proc. of the 5th Int. conf. on Numerical Grid Generation in Computational Field Simulation. – Mississippi State University, 1996. P. 659–670.
3. Li T. S., Armstrong C. G., McKeag R. M. Automatic partitioning of analysis models using the medial axis transform // Advances in Parallel and Vector Processing for Structural Mechanics. – Edinburgh, UK: Civil-Comp Press, 1994. P. 165–171.

4. Sheffer A., Etzion M., Rappoport A., Bercovier M. Hexahedral mesh generation using the embedded Voronoi graph // Proc. of the 7th Int. Meshing Roundtable. – Dearborn, Michigan, USA, 1998. P. 347–364.
5. Schneider R. A grid-based algorithm for the generation of hexahedral element meshes // Engineering with computer. 1996. Vol. 12, N 3–4. P. 168–177.
6. Weiler F., Shindler R. Automatic geometry-adaptive generation of quadrilateral and hexahedral element meshes for the FEM // Proc. of the 5th Int. conf. on Numerical Grid Generation in Computational Field Simulation. – Mississippi State University, 1996. P. 689–697.
7. Фархутдинов В. Ф., Тарасов В. И., Соловьев А. Н. и др. Обзор возможностей по обработке геометрических и построению сеточных моделей в ЛОГОС-Препост // Тр. XIV Международ. конф. «Супервычисления и математическое моделирование». – Саров: РФЯЦ-ВНИИЭФ, 2013. С. 585–592.
8. Погосян М. А., Савельевских Е. П., Шагалиев Р. М. и др. Применение отечественных суперкомпьютерных технологий для создания перспективных образцов авиационной техники // Вопросы атомной науки и техники. Сер. Матем. моделирование физ. процессов. 2013. Вып. 2. С. 3–17.
9. Сафронов А. В., Дерюгин Ю. Н., Жучков Р. Н. и др. Результаты валидации многофункционального пакета программ ЛОГОС при решении задач аэрогазодинамики старта и полета ракет-носителей // Математическое моделирование. 2014. Т. 26, № 9. С. 83–95.
10. STAR-CCM+ Siemens PLM Software [Electronic resource]. – <https://mdx.plm.automation.siemens.com/star-ccm-plus>.
11. Jiangtao Hu, Lee Y. K., Blacker T., Jin Zhu. Overlay grid based geometry cleanup // Proc. of the 11th Int. Meshing Roundtable. – Ithaca, New York, USA, 2002. P. 313–324.
12. Karlsson J. Implementing anisotropic adaptive mesh refinement in OpenFOAM // Master's Thesis in Computer Science, Algorithms, Languages and Logic. – Chalmers University of Technology, Gothenburg, Sweden, 2012.
13. Козелков А. С., Жучков Р. Н., Уткина А. А., Володченкова К. Б. Моделирование турбулентных течений на сетках гибридной структуры с использованием схем высокого порядка точности // Вопросы атомной науки и техники. Сер. Матем. моделирование физ. процессов. 2014. Вып. 3. С. 18–31.
14. Jiao X., Heath M. T. Feature detection for surface meshes // Proc. of the 8th Int. Conf. on Numerical Grid Generation in Computational Field Simulations. – Honolulu, Hawaii, USA, 2002. P. 705–714.
15. Baker T. Identification and preservation of surface features // Proc. of the 13th Int. Meshing Roundtable. – Williamsburg, Virginia, USA, 2004. P. 299–310.
16. Jiao X., Alexander P. Parallel feature-preserving mesh smoothing // Proc. of the Int. conf. on Computational Science and Applications. – Singapore, 2005. P. 1180–1189.
17. Tomac M., Eller D. Towards automated hybrid-prismatic mesh generation // Procedia Engineering. 2014. Vol. 82. P. 377–389.
18. Kallinderis Y., Ward S. Prismatic grid generation for three-dimensional complex geometries // AIAA Journal. 1993. Vol. 31, N 10. P. 1850–1856.
19. Floater M. S. Mean value coordinates // Computer Aided Geometric Design. 2003. Vol. 20(1). P. 19–27.

20. Fujimoto K., Kozo Fujii, Wang Z. J. Improvements in the reliability and efficiency of body-fitted cartesian grid method // 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition. – Orlando, Florida, USA, 2009.
21. Aftosmis M. J. Solution adaptive cartesian grid methods for aerodynamic flows with complex geometries // Von Karman Institute for Fluid Dynamics, Lecture Series. – Rhode-Saint-Genese, Belgium, 1997.
22. Berger M. J., Aftosmis M. J. Aspects (and aspect ratios) of cartesian mesh methods // Proc. of the 16th Int. conf. on Numerical Methods in Fluid Dynamics, to Appear in «Lecture Notes in Physics» – Berlin: Springer, 1998. Vol. 515. P. 1–12.
23. Liang Q., Zang J., Borthwick A. G. L., Taylor P. H. Shallow flow simulation on dynamically adaptive cut-cell quadtree // Int. J. for Numerical Methods in Fluids. 2007. Vol. 53(12). P. 1777–1799.
24. Kovalev K. Unstructured hexahedral non-conformal mesh generation: Ph. D. Dissertation. University of Brussel, 2005.
25. Oaks W., Paoletti S. Polyhedral mesh generation // Proc. of the 9th Int. Meshing Roundtable. – New Orleans, LA, USA, 2000. P. 57–67.
26. Zavattieri P. Optimization strategies in unstructured mesh generation // Int. J. for Numerical Methods in Engineering. 1996. Vol. 39. P. 2055.
27. Handbook of Grid Generation / Eds. J. F. Thompson, B. K. Soni, N. P. Weatherill. – Boca Raton, Florida, USA: CRC PRESS, 1999. Chap. 21. P. 7–18.
28. Kruse R. L., Bruce P., Tondo C. L. Data structure and program design in C. – Upper Saddle River, New Jersey, USA: Prentice-Hall, 1991.
29. Lorensen W. E., Cline H. E. Marching cubes: a high resolution 3D surface reconstruction algorithm // ACM Computer Graphics. 1987. Vol. 21(4). P. 163–169.
30. Jasak H., Tukovi Z. Automatic mesh motion for the unstructured finite volume method // Transactions of FAMENA. 2006. Vol. 30(2). P. 1–20.
31. Голованов Н. Н. Геометрическое моделирование. – М.: Физматлит, 2002. С. 391.

An Automatic Generator of Unstructured Polyhedral Grids in the LOGOS Software Preprocessor

D. N. Smolkina, O. N. Borisenko, M. V. Cherenkova, A. G. Giniyatullina,
M. V. Kuz'menko, N. V. Chukhmanov, E. V. Potekhina, N. V. Popova, M. R. Turusov

The paper describes an automatic generator of unstructured grids used to solve problems of aero- and hydromechanics implemented in LOGOS software complex. Stages of generating a 3D grid are considered, an example of the stage-by-stage construction of a grid is given.