

## МОДИФИКАЦИИ АЛГОРИТМА УПРОЩЕНИЯ ПОВЕРХНОСТНОЙ ТРЕУГОЛЬНОЙ СЕТКИ В РАЗНЫХ ПОДХОДАХ ПЕРЕСТРОЕНИЯ ФАСЕТОЧНОЙ МОДЕЛИ В ПРЕПРОЦЕССОРЕ ЛОГОС АЭРО-ГИДРО

*Евстифеева Елена Олеговна (EOEvstifeeva@vniief.ru)*

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

В настоящее время для генерации поверхностной треугольной сетки по модели в фасеточном представлении в препроцессоре пакета программ ЛОГОС для решения задач аэро- и гидродинамики реализовано два независимых подхода. Каждый из подходов использует алгоритм упрощения сетки в своей последовательности этапов. Алгоритм упрощения сетки в зависимости от контекста применения имеет определенную цель: в одном из подходов – это максимально допустимое сокращение количества ячеек без потери кривизны поверхности исходной модели, в другом – получение сетки из ячеек высокого качества, размеры которых удовлетворяют кривизне исходной поверхности модели и значениям, заданным пользователем. Для достижения поставленной цели в каждом из подходов было разработано две модификации алгоритма упрощения сетки, описанные в данной работе.

**Ключевые слова:** пакет программ ЛОГОС, программный модуль ЛОГОС АЭРО-ГИДРО, препроцессор, генератор поверхностной треугольной сетки.

## MODIFICATIONS OF THE ALGORITHM FOR SURFACE TRIANGULAR MESH SIMPLIFICATION IN DIFFERENT FACET MODEL REMESHING APPROACHES IN LOGOS CFD PREPROCESSOR

*Evstifeeva Elena Olegovna (EOEvstifeeva@vniief.ru)*

FSUE «RFNC-VNIIEF», Sarov Nizhny Novgorod region

There are currently two approaches in the preprocessor of LOGOS CFD program module of LOGOS software package for surface triangular mesh generation for facet model representation. Each of these two approaches uses a mesh simplification algorithm in its stage sequence. The mesh simplification algorithm has a specified purpose that is defined by the context of its implementation: in the first approach, it is the maximal permissible cells reduction without the loss of initial model surface curvature; in the second, it is the achievement of high quality mesh with cells, the sizes of which are appropriate to initial model surface curvature and user-defined values. To achieve the specified purpose for each of the approaches, two modifications of the simplification algorithm were created. These modifications of simplification algorithm are described in this work.

**Key words:** LOGOS software package, LOGOS CFD program module, preprocessor, surface triangular mesh generator.

### Введение

В настоящее время для генерации поверхностной треугольной сетки по модели в фасеточном представлении в препроцессоре пакета программ ЛОГОС [1] для решения задач аэро- и гидродинамики (препроцессор ЛОГОС АЭРО-ГИДРО) реализовано два независимых подхода [2, 3]. Выбор подхода

для генерации сетки осуществляется посредством предварительного анализа исходной сетки. Несмотря на то, что для генерации поверхностной сетки каждый из подходов использует свою технику, а именно выполняет определенную последовательность этапов, в каждой из них применяется алгоритм упрощения сетки, причем в определенной модификации.

Алгоритм упрощения сетки в зависимости от контекста его применения имеет определенную цель: в одном из подходов – это максимально допустимое сокращение количества ячеек без потери кривизны поверхности исходной модели, в другом – получение сетки из ячеек высокого качества, размеры которых удовлетворяют кривизне исходной поверхности модели и значениям, заданным пользователем. Для достижения поставленной цели для каждого из подходов была разработана своя модификация алгоритма упрощения сетки.

Актуальность разработки и применения алгоритма упрощения и его модификаций обоснована развитием областей промышленности, в которых активно внедряется и используется 3D-моделирование, как в целях быстрого построения сеточных моделей высокого качества, так и в целях ускорения визуализации графики. Исследование выполнено в рамках научной программы Национального центра физики и математики (направление «Математическое моделирование на супер-ЭВМ экс- и зетафлопсной производительности»).

В данной работе будут описаны модификации алгоритма упрощения сетки, их цели и области применения. Визуальные результаты работы различных модификаций алгоритма будут представлены на рисунках с фрагментами сеток. Также будут приведены скоростные и численные характеристики построения сеток для оценки результатов работы алгоритма в разных модификациях.

### Подходы к перестроению сетки

Генерация поверхностной треугольной сетки по модели в фасеточном представлении в препроцессоре пакета программ ЛОГОС для решения задач аэро- и гидродинамики возможна с помощью одного из двух независимых подходов.

Первый (стандартный) подход представлен в работе [2]. До реализации и внедрения алгоритма упрощения сетки построение занимало достаточно длительное время для сеток высокой детализации, состоящих более чем из 1 миллиона ячеек. Построение сетки на каждой выделенной границе происходит в пространстве  $R^2$  методом подвижного фронта после предварительного перестроения кривых между выделенными границами. Затем полученная сетка переносится в пространство  $R^3$  с помощью заданной функции отображения. Данный способ построения не всегда устойчив к сеткам с ошибками, например, с наличием пересечений, а также с нетривиальным описанием поверхности модели, например, небольшими выступами произвольной формы, скруглениями и т. д., где не удается получить сетку на границе. В таких случаях увеличивается количество итераций построения для получения приемлемого результата, что в свою очередь увеличивает и время, ощутимо возрастающее для моделей высокой детализации.

Целью внедрения алгоритма упрощения в цепочку этапов первого подхода было ускорение подготовки результирующей сетки за счет сокращения количества ячеек до-перестроения, что по предположению должно было снизить сложность выполнения внутренних вычислительных операций.

Второй (новый) подход к перестроению сетки [3] был разработан для перестроения фасеточных моделей высокой детализации, например, подготовленных с помощью генератора замкнутой оболочки [4]. Перестроение сетки происходит в пространстве  $R^3$ , основные этапы второго подхода следующие:

1. Упрощение сетки.
2. Оптимизация построенной сетки.

Основополагающим этапом нового подхода является алгоритм упрощения сетки, а сам подход в свою очередь был разработан, исходя из предположения, что результирующая сетка может быть получена путем итерационного сокращения ячеек без переноса сетки в пространство  $R^2$ , следовательно, без выполнения трудоемких операций, затратных по времени. Таким образом, основной задачей реализации алгоритма упрощения в данном контексте являлось создание нового подхода к генерации поверхностной сетки.

Непосредственно целью алгоритма упрощения в рамках второго подхода является быстрое перестроение большей части исходной модели с получением результирующей сетки, состоящей из ячеек высокого качества.

В обоих случаях алгоритм упрощения сетки учитывает кривизну [5] исходной поверхности модели и не допускает ее искажения. В первом подходе ввиду того, что впоследствии сетка будет перестроена, не накладываются ограничения на получение ячеек высокого качества, не считая запрета на появление вырожденных треугольников с качеством ниже порогового значения. Во втором подходе получение ячеек высокого качества и заданного размера – одно из приоритетных условий работы алгоритма.

Далее рассмотрим общий алгоритм упрощения сетки и его модификации в контексте применения в каждом из подходов.

### Общая схема алгоритма упрощения сетки

Общая идея алгоритма упрощения сетки взята из работы [6], на ее основе разработан оригинальный алгоритм, состоящий из следующих этапов:

- вычисление и сглаживание метрик<sup>1</sup>;
- упрощение характерных кривых;
- упрощение сетки границ.

<sup>1</sup> Метрика в вершине – желаемый размер ребер, инцидентных данной вершине, при котором поверхность в окрестности данной вершины будет аппроксимирована с заданной точностью.

Вычисление метрик производится на основе кривизны исходной поверхности. Далее метрики ограничиваются размерами, заданными пользователем для каждой из границ, и производится сглаживание для недопущения резкого изменения значений.

В полный набор характерных кривых для упрощения на втором этапе включены ребра, выделенные по острому углу между треугольниками сетки, характерные кривые, выделенные пользователем, а также ребра, разделяющие разные патчи сетки. Этап упрощения характерных кривых выполняется вторым для обеспечения распараллеливания упрощения границ, что возможно при условии неизменности ограничивающих их ребер.

На вход алгоритма передаются параметры, соответствующие этапу, а также сеточные данные, по которым формируется список начальных ребер для упрощения, и списки всех характерных кривых. Опуская различие в некоторых входных параметрах для выбранного этапа, алгоритмы упрощения характерных кривых и упрощения сетки границ имеют одинаковую структуру, которая заключается в следующем:

1) поиск оптимальной точки для каждого ребра из начального списка;

2) составление очереди из всех рассматриваемых ребер, сортированной по возрастанию веса ребра, вычисленного по определенному закону;

3) повторение итерации упрощения:

○ завершение алгоритма упрощения, если для всех ребер установлен специальный вес;

○ замена первого в очереди ребра  $e$  вершиной  $v$  (или операция «схлопывания» ребра), координаты которой совпадают с координатами оптимальной точки, вычисленной для ребра  $e$ ;

○ проверки геометрической и топологической допустимости [3];

○ проверка выполнения целевого критерия;

○ в случае успешного прохождения проверок с предыдущих шагов – обновление информации для всех ребер, инцидентных новой вершине, обновление очереди; в случае неуспешной проверки – установление для ребра специального веса и откат операции схлопывания;

○ переход к следующей итерации.

Основные различия модификаций алгоритма упрощения заключаются в выборе оптимальной точки для операции схлопывания ребра, а также в проверках целевого критерия, что будет описано ниже.

### Поиск оптимальной точки

Для выбора позиции новой вершины реализовано несколько режимов операции схлопывания ребра: в оптимальную точку и в одну из вершин ребра. Режим выбирается в зависимости от текущего этапа упрощения.

Поиск оптимальной точки заключается в решении системы линейных уравнений, т. е.  $v$  - является пересечением трех непараллельных плоскостей в

пространстве  $R^3$ . При составлении системы уравнений  $a_i^T v = b_i$  для поиска оптимальной точки каждое новое ограничение проверяется на линейную независимость с уже добавленными. В работе рассматривается более чем три ограничения, чтобы в случае линейной зависимости при добавлении нового ограничения сохранилась возможность получения оптимальной точки. После того, как были найдены все ограничения, новая вершина  $v$  вычисляется как:

$$v = A^{-1}b. \quad (1)$$

Некоторые из ограничений получены минимизацией квадратичной целевой функции  $f(e, v)$ , т. е. решением задачи квадратичного программирования [6]. В данном случае целевая функция может быть приведена к форме:

$$f(e, v) = \frac{1}{2}v^T H v + c^T v + \frac{1}{2}k, \quad (2)$$

где  $H$  – симметричный гессиан;  $Hv + c$  – градиент функции  $f$ ;  $k$  – константа.

Дано  $n$  ограничений  $(A, b)$ , пусть столбцы  $Z$  – базис в  $R^3$  с первыми  $n$  равными  $A^T$ . Остальные  $3-n$  столбца  $Z$  заданы ортогональными векторам  $a_i$ . Тогда дополнительные  $3-n$  ограничений задаются в виде:

$$I_{(3-n,3)}Z^{-1}(Hv + c) = 0, \quad (3)$$

где  $I_{(3-n,3)}$  – подматрица размерности  $3-n$  на  $3$ , сформированная удалением верхних  $n$  строк из единичной матрицы  $I$ .

Первое ограничение отвечает за сохранение объема исходной модели:

$$\sum_i V(v, v_0^i, v_1^i, v_2^i) = \sum_i \frac{1}{6} \begin{vmatrix} v_x & v_{0x}^i & v_{1x}^i & v_{2x}^i \\ v_y & v_{0y}^i & v_{1y}^i & v_{2y}^i \\ v_z & v_{0z}^i & v_{1z}^i & v_{2z}^i \\ 1 & 1 & 1 & 1 \end{vmatrix} = 0. \quad (4)$$

Следующее условие направлено на оптимизацию формы треугольников, полученных в окрестности ребра после его схлопывания, таким образом, что равносторонние треугольники являются предпочтительнее вытянутых. Для измерения величины качества треугольника по форме было выбрано следующее выражение:

$$f_S(e, v) = \sum_i L(v, v_i)^2, \quad (5)$$

правосторонняя часть которого есть сумма квадратов длин ребер, инцидентных новой вершине  $v$ .

Еще одним условием является оптимизация объема, которое заключается в минимизации суммы беззнаковых объемов тетраэдров, определяющих значение локального отклонения для соответствующих треугольников в окружении удаляемого ребра. Для минимизации отклонения ищется минимум следующей функции:

$$f_V(e, v) = \sum_i V(v, v_0^i, v_1^i, v_2^i)^2. \quad (6)$$

Вес ребра вычисляется подстановкой оптимальной точки в целевую функцию (6).

### Первая модификация алгоритма

Выбор оптимальной точки схлопывания в первой модификации алгоритма выполняется путем решения системы уравнений, добавленных в порядке (4), (6), (5). Как было сказано выше, обязательным и самым приоритетным условием при составлении системы является сохранение объема исходной модели. Далее добавляются ограничения, оптимизирующие локальное изменение объемов тетраэдров, и на последнем шаге, если в случае линейной зависимости добавляемых ранее ограничений их количество недостаточно, рассматривается условие оптимизации формы модифицированных треугольников.

Стоит отметить, что ограничение, оптимизирующее форму треугольников, добавляется последним и может вообще не участвовать при составлении системы для поиска оптимальной точки.

Целевым критерием для проверки успешности итерации является выполнение условий, которые отслеживают как изменение размеров ребер, так и изменение кривизны исходной модели. Если длина  $l_i^1$  измененного ребра не превышает среднее значение метрик в его вершинах, то есть выполняется условие (7), то операция считается допустимой и выполняется переход к следующей итерации.

$$l_i^1 \leq \alpha_{lower} \frac{m_i^1 + m_i^2}{2}, \quad (7)$$

где  $m_i^1, m_i^2$  – метрики в вершинах ребра;  $\alpha_{lower}$  – масштабирующий коэффициент, допускающий отклонение размера.

Если условие (7) не выполнено, необходимо проверить дополнительные параметры изменения поверхности. В начале алгоритма для каждого треугольника сетки инициализируется накопительная величина изменения объема  $tol_i = 0,0$ . На каждой операции схлопывания для  $k$ -го модифицированного треугольника вычисляется вызванное его смещением знаковое изменение объема  $tol_{curr}^k$  и суммируется с текущим значением  $tol_k$ . Если величина суммы  $tol_k + tol_{curr}^k$  для каждого из измененных треугольников не превышает глобального порогового

значения  $\epsilon_{tol}$ , то операция считается допустимой. Глобальное пороговое значение рассчитывается по формуле:

$$\epsilon_{tol} = (km_{min})^3/6, \quad (8)$$

где  $m_{min}$  – минимальное значение метрики на границе;  $k$  – константа, при реализации экспериментальным путем было выбрано значение 0,01;

Дополнительно значение  $\epsilon_{tol}$ , полученное по формуле (8) ограничивается по формуле (9):

$$1e - 22 \leq \epsilon_{tol} \leq 1e - 9. \quad (9)$$

### Вторая модификация алгоритма

Во второй модификации алгоритма порядок добавления ограничений следующий: (4), (5), (6). Таким образом после добавления приоритетного ограничения, сохраняющего объем исходной модели, добавляются ограничения, оптимизирующие форму измененных треугольников. На последнем шаге в случае недостаточного для составления матрицы количества ограничений, добавляются ограничения, оптимизирующие локальное изменение объемов тетраэдров.

Целевым критерием для выполнения итерации упрощения является проверка размеров измененных ребер в окружении новой вершины. Длина  $l_i^1$  измененного ребра не должна превышать среднего значения метрик в его вершинах, следовательно если неравенство (7) выполнено для всех измененных ребер, то итерация упрощения считается успешной и осуществляется переход к следующей, иначе проверяется выполнение дополнительных условий:

$$l_i^1 \leq \alpha_{upper} \frac{m_i^1 + m_i^2}{2}, \quad (10)$$

$$q_{avg}^1 > q_{avg}, \quad (11)$$

где  $\alpha_{upper}$  – масштабирующий коэффициент, допускающий отклонение размера;  $q_{avg}$  и  $q_{avg}^1$  – среднее значение качества треугольников в окрестности заменяемого ребра до и после итерации упрощения.

Если выполнены оба неравенства (10) и (11), то итерация упрощения считается успешной.

При выборе масштабирующих коэффициентов нужно учитывать, что  $1 \leq \alpha_{lower} < \alpha_{upper}$ . Например, при установке значений  $\alpha_{lower} = 1,1$ ;  $\alpha_{upper} = 1,25$  гарантируется, что размер полученных ребер не превысит значений, вычисленных по кривизне, более чем на 25 %.

## Результаты

Для визуальной и численной оценки результатов обратимся к цели каждой из модификаций алгоритма, поставленной в описании подходов.

Цель первой модификации – ускорение подготовки результирующей сетки за счет сокращения количества ячеек, при этом не должно происходить искажение исходной поверхности, выраженное в сильном изменении ее кривизны.

Целью второй модификации является быстрое перестроение большей части исходной модели с получением результирующей сетки, состоящей преимущественно из ячеек высокого качества.

Результаты упрощения сетки для визуальной оценки работы алгоритма в каждой из модификаций представлены на примере исходной сетки [7] с рис. 1.



Рис. 1. Фрагмент исходной сетки

Результат упрощения сетки с рис. 1 с помощью первой модификации показан на рис. 2. Визуальная оценка результата показывает, что сетка после упрощения аналогична результату извлечения тесселяции [8], при котором выполняется замощение параметрической модели треугольниками, исходя из кривизны поверхности. При этом по направлению малого изменения кривизны могут быть получены длинные вытянутые треугольники (в пределах заданного порогового значения качества), которые аппроксимируют исходную поверхность с высокой точностью.



Рис. 2. Фрагмент сетки после алгоритма упрощения в первой модификации

Результат упрощения сетки с рис. 1 с помощью второй модификации показан на рис. 3. Сетка, полученная таким образом, удовлетворяет значениям вычисленных метрик [3] и при этом треугольники сетки преимущественно хорошего качества.



Рис. 3. Фрагмент сетки после алгоритма упрощения во второй модификации

Для проверки достижения поставленной цели для первой модификации обратимся к диаграмме с рис. 4, на которой на примере тестовых моделей различной сложности показано сравнение общего времени построения сетки с использованием алгоритма упрощения и без его использования. Над вторым столбцом пары указано относительное время построения сетки с использованием упрощения относительно времени построения без его использования  $t_j$ .

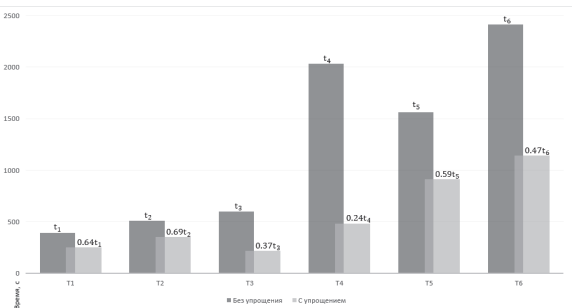


Рис. 4. Сравнение времени построения сеток с использованием первой модификации алгоритма упрощения и без его использования

По результатам, приведенным на рис. 4, видно, что с использованием первой модификации алгоритма упрощения на представленной выборке задач различной сложности время построения было сокращено до 76 %.

Наряду со скоростными характеристиками сравнивалось отклонение результирующих сеток, построенных с использованием алгоритма упрощения и без его использования. Сравнение поверхностных сеток проводилось с помощью программы metro (см. [9]), измеряющей расстояние Хаусдорфа. По результатам сравнения отклонение не превышает  $0,0006d$ , где  $d$  – длина диагонали габаритной коробки, описывающей модель.

Для проверки выполнения цели для второй модификации алгоритма упрощения было измерено качество результирующих сеток, подготовленных с помощью нового подхода, реализованного на основе алгоритма упрощения, и с помощью стандартного подхода (с использованием алгоритма упрощения). Результаты сравнения среднего и минимального качества для выборки тестовых моделей различной сложности представлены на рис. 5, 6.

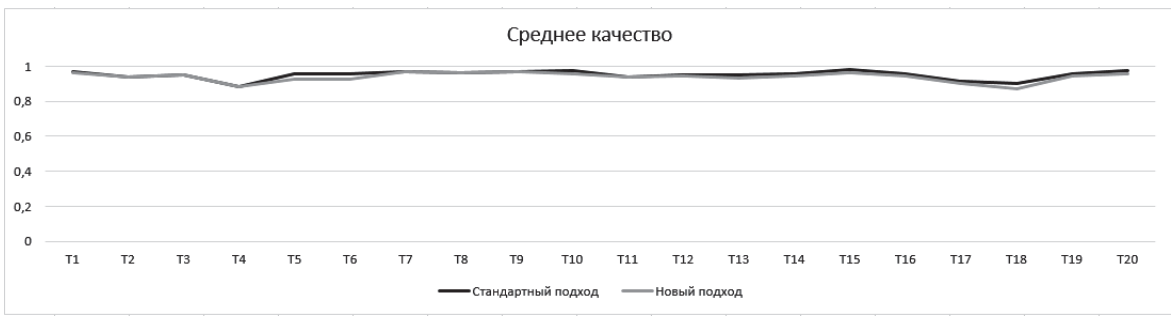


Рис. 5. Сравнение среднего качества результирующих сеток

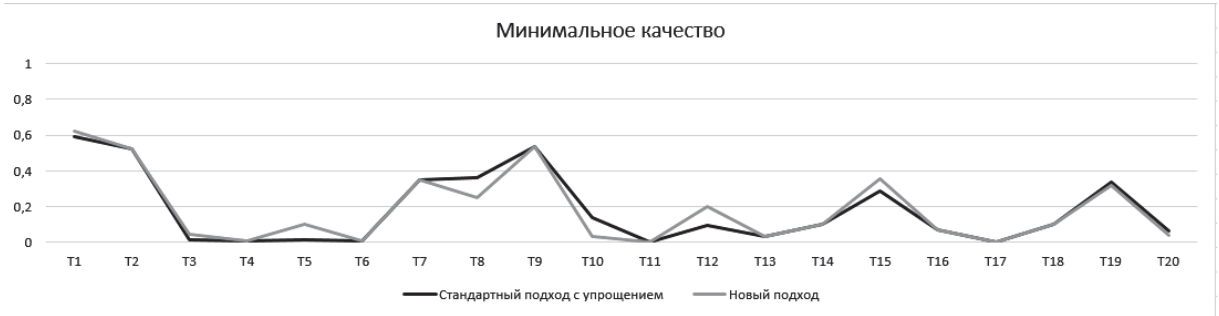


Рис. 6. Сравнение минимального качества результирующих сеток

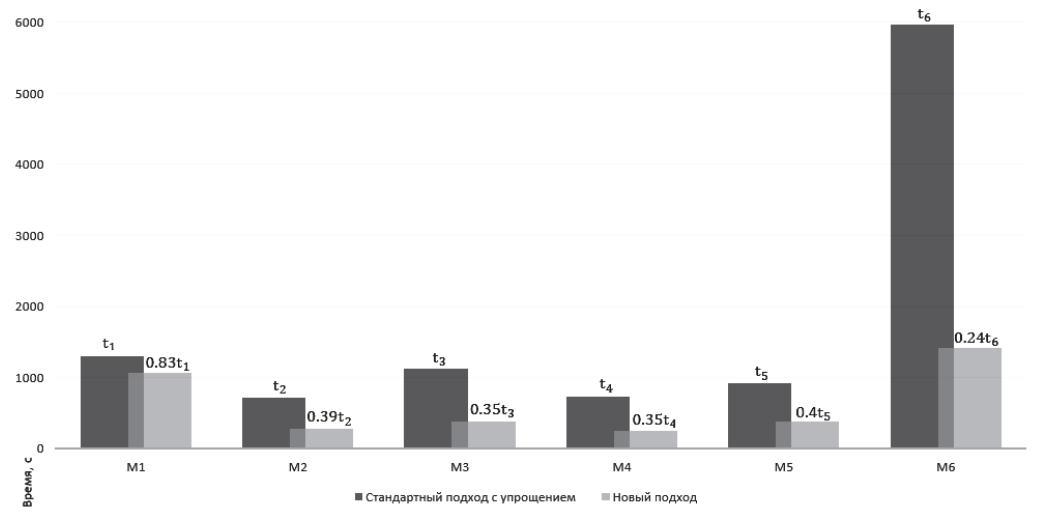


Рис. 7. Сравнение времени подготовки сеток разными подходами

Результаты графика с рис. 5 демонстрируют, что отклонение среднего качества результирующих сеток, подготовленных стандартным подходом (с алгоритмом упрощения) и новым подходом, минимально. Значение среднего качества в обоих случаях близко к 1.0, что характеризует идеальный по качеству (равносторонний) треугольник. По результатам графика с рис. 6 видно, что минимальное качество в обоих случаях имеет небольшое отклонение, это связано с различием в обработке модели, но в целом получено хорошее согласие.

Также сравнивалось время построения сеток с помощью разных подходов, поскольку ускорение подготовки дискретной модели являлось основной задачей создания нового подхода на основе второй модификации алгоритма упрощения. Результаты показаны на

рис. 7, где над вторыми столбцами пары указано относительное время выполнения нового подхода относительно времени стандартного подхода  $t_i$ .

По результатам диаграммы с рис. 7 можно сделать вывод, что для приведенной выборки моделей высокой детализации цель разработки нового подхода к построению сетки на основе второй модификации алгоритма упрощения достигнута, время построения результирующей сетки сократилось в среднем на 50%.

В дополнение к сравнению качества проводилось сравнение [9] поверхностных сеток. Максимальное отклонение среди тестовых моделей не превышает  $0,0009d$ , где  $d$  - длина диагонали габаритной коробки, описывающей модель.

## Заключение

По результатам проведенных работ реализовано две модификации алгоритма упрощения поверхностной сетки, которые на данный момент внедрены в цепочки этапов различных подходов к построению поверхностной сетки в препроцессоре пакета программ «Логос» для решения задач аэро- и гидродинамики.

Использование первой модификации алгоритма в стандартном подходе способствовало ускорению работы подготовки результирующей сетки в целом. Результаты тестирования показали, что на моделях, состоящих из большого количества ячеек, было достигнуто ускорение до 76 %. При этом кривизна исходной поверхности не искажается в результате упрощения, а результирующие сетки, построенные с использованием алгоритма упрощения и без его использования, имеют малое отклонение друг от друга.

Реализация второй модификации алгоритма упрощения положила основу для разработки нового подхода к построению поверхностных сеток, что позволило производить подготовку сеток хорошего качества с заданными размерами непосредственно в пространстве  $R^3$  и ускорило генерацию сетки для моделей высокой детализации по сравнению со стандартным подходом с использованием алгоритма упрощения. При этом сравнение результирующих сеток, подготовленных различными подходами, показало их минимальное отклонение.

Работа выполнена в рамках научной программы Национального центра физики и математики по направлению «Математическое моделирование на супер-ЭВМ экса- и зеттафлопсной производительности».

## Список литературы

1. Козелков А. С., Лашкин С. В., Куркин А. А. и др. Параллельная реализация метода SIMPLE на основе

многосеточного метода // Сибирский журнал вычислительной математики. 2020. Т. 23, № 1. С. 1–22.

2. Лукичев А. Н., Цалко Т. В., Панкратов Д. М. и др. Особенности построения триангуляционных сеток на поверхностях в аналитическом и фасеточном представлении / 15-я научно-техническая конференция «Молодежь в науке» // Сборник докладов. Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2017. С. 73–78.

3. Евстифеева Е. О. Алгоритм упрощения поверхностной треугольной сетки при подготовке задач аэрогидродинамики в пакете программ ЛОГОС / 19-я научно-техническая конференция «Молодежь в науке» // Сборник тезисов. Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2021. С. 18–19.

4. Никитин В. А., Шурыгин А. В., Новиков И. Г. и др. Программный модуль генерации замкнутой поверхностной триангуляционной сетки в пакете программ «Логос» // ВАНТ. Сер. Математическое моделирование физических процессов. 2021. Вып. 2. С. 70–79.

5. Белова А. И., Лукичев А. Н., Борисенко О. Н. Вычисление кривизны дискретных поверхностей в генераторе поверхностных триангуляционных сеток «Логос. Препост» / 15-ая научно-техническая конференция «Молодежь в науке» // Сборник докладов. Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2017. С. 34–40.

6. Lindstrom P., Turk G. Fast and Memory Efficient Polygonal Simplification // Proc. Visualization '98, IEEE Computer Soc. Press. Oct. 1998. P. 279–286.

7. 3rd AIAA CFD Drag Prediction Workshop, San Francisco, June 2006 // URL: <http://aiaa-dpw.larc.nasa.gov/Workshop3/workshop3.html> (дата обращения: 08.07.2021).

8. Sadreghighi I. Mesh generation in CFD. A review-CFD Open Series, Patch 1.86.5, 2019. P. 405.

9. Cignoni P., Rocchini C., Scopigno R. Metro: Measuring Error on Simplified Surfaces. // Technical Report B4-01-01-96, Istituto I. E. I.-C. N. R. – Pisa, Italy – January 1996.