

# АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ НЕСТРУКТУРИРОВАННЫХ СЕТОК НА МОДЕЛЯХ В АНАЛИТИЧЕСКОМ ПРЕДСТАВЛЕНИИ В «ЛОГОС. ПРЕПОСТ»

*А. Н. Лукичев, А. Н. Соловьев, В. В. Лазарев, Д. В. Логинов*

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

## Введение

В современной промышленности все большую роль играют компьютерное моделирование и инженерный анализ. Их использование позволяет во много раз сократить затраты времени и средств на разработку, создание и модернизацию различных технических устройств, от простейших конструкций до самолетов и атомных электростанций. Для обеспечения полного цикла инженерного анализа в настоящее время используется широкий спектр вычислительных систем и программных комплексов. Развитие последних позволяет ускорить время расчетов, повысить точность вычислений и увеличить масштабы моделирования, позволяя производить расчеты все более сложных задач.

В Институте теоретической и математической физики в рамках проекта «Суперкомпьютерные технологии» ведется разработка программного комплекса «Логос», который представляет собой пакет кроссплатформенных программных модулей, обеспечивающих поддержку полного цикла инженерного анализа – от загрузки CAD моделей до анализа результатов расчета. Одним из компонентов комплекса является программный продукт «Логос. ПреПост», который обеспечивает задание начальных данных и генерацию сеток в задачах инженерного анализа, подготовку и запуск счета с помощью методик, реализованных в программном комплексе «Логос», проведение пред- и постобработки сеточной модели и результатов счета, визуализацию данных на всех этапах моделирования и анализа.

Важной составляющей «Логос. ПреПост» являются модули генерации расчетных сеток. От качества построенных сеток, правильного подбора параметров и размеров элементов зависят скорость и точность расчета, и успех моделирования в целом. В данном докладе рассматривается структура и реализация модуля автоматической генерации конечно-элементных неструктурированных сеток на CAD моделях в аналитическом представлении.

## Представление аналитической геометрии в «Логос. ПреПост»

В «Логос. ПреПост» для хранения аналитических геометрических моделей используется граничное представление деталей или BREP представление

(рис. 1), поскольку является основным способом представления геометрических форм в современных системах геометрического моделирования, таких как Parasolid и ACIS, лежащих в основе многих коммерческих САПР.



Рис. 1. BREP представление твердого тела

Граничное представление (англ. Boundary Representation, BREP) – это метод представления объемной формы путем описания ее границ. В этом методе трехмерное тело представляется набором связанных друг с другом граней, задающих границу между представляемым телом и остальным пространством. Каждая грань описывается через поверхность грани и контуров – замкнутых наборов ребер, образующих на поверхности ограниченную многосвязную область. Каждое ребро разделяет две соседние грани. Ребра представляют собой ограниченные трехмерные кривые. В большинстве случаев при использовании BREP представления используются аналитические кривые и поверхности, т. е. такие объекты, которые можно описать математическими уравнениями. Чаще всего, для описания кривых и поверхностей используют параметрическое задание.

В «Логос. ПреПост» деталь может состоять из нескольких фрагментов, каждый из которых представляет собой BREP модель. Каждый фрагмент состоит из граней, каждая грань представляет собой описание поверхности и массив ребер, ограничивающих ее. В свою очередь ребра содержат в себе описание кривой и пару точек – вершин, ограничивающих кривую.

Для построения сеток на аналитических геометриях необходим ряд геометрических функций, таких как получение координат точки на поверхности или кривой, получение параметров кривой или поверхности в точке: первой, второй производной по направлениям, кривизны, направление нормалей. Также необходимы функции проецирования точки на по-

верхность и кривую и нахождение точки поверхности, ближайшей к заданной точке. Все эти функции реализованы в используемом геометрическом ядре 3DTV, поставляемом компанией ТЕСИС.

## Построение неструктурированных сеток в «Логос. ПреПост»

Для моделирования и расчетов протекания физических процессов используют численные методы моделирования, которые оперируют с дискретными моделями, на которых построены расчетные сетки.

Как и аналитические модели, сетки имеют базовые топологические элементы:

- Узел, или элемент нулевой размерности.
- Ребро, соединяющее два узла, – одномерный элемент.
- Грань – часть плоскости, ограниченной ребрами, – двумерный элемент. При рассмотрении поверхностных сеток грань является ячейкой сетки.
- Трехмерная ячейка (или просто ячейка, при рассмотрении объемных сеток) – часть пространства, ограниченная набором граней.

Различают структурированные (регулярные) и неструктурированные (нерегулярные) сетки. Особенностью структурированных сеток является возможность индексации ячеек сетки и по известному индексу одной ячейки точно определять все соседние ячейки. Преимуществом данного вида сеток является отсутствие необходимости хранения информации о соседних элементах, что уменьшает затрачиваемую на хранение память, а также уменьшает время расчетов за счет более простых и оптимальных алгоритмов. Недостатком структурированных сеток является то, что не на любой геометрии можно построить структурированную сетку. Кроме того, регулярные расчетные сетки подходят только для задач с отсутствием или незначительной деформацией сеток.

Для задач деформации и разрушения, а также для задач со сложной геометрией используют неструктурированные сетки. Их преимуществом является большая область применимости и меньшее количество элементов при дискретизации сложных областей. К недостаткам относятся большие, по сравнению с регулярными сетками, затраты памяти на хранение, более сложные алгоритмы построения и поиска элементов, более сложные алгоритмы счета [1].

Для хранения и визуализации обрабатываемых сеток в «Логос. ПреПост» используется свободно распространяемая библиотека VTK [2]. В данной библиотеке для хранения сеток используется ячеечно-узловая структура. В ней хранится общий массив точек сетки и массив ячеек. Каждая ячейка содержит свой тип и вектор индексов точек из общего массива. Форма ячейки определяется типом ячейки и порядком точек. Кроме того, можно использовать особый тип ячеек – «многогранная ячейка», позволяющих реализовать реберно-граневую структуру, добавляя в ячейку список граней, образующих ее, тем самым увеличивая

объем затрачиваемой памяти, но уменьшая время доступа и поиска соседних элементов. Достоинством данного формата хранения являются малые затраты памяти – нет необходимости хранить информацию о соседях, быстрый доступ к элементам сетки по индексу в массиве, быстрые сохранение и загрузка сетки. К недостаткам относятся сложность модификации топологии сетки и сложность поиска соседних элементов.

Для генерации неструктурированной сетки в «Логос. ПреПост» используется библиотека CM2 MeshTools [3], которая предоставляет возможность генерации поверхностных треугольных и четырехугольных, и объемных тетраэдрных и преимущественно шестигранных неструктурированных сеток. В данном генераторе объемные сетки строятся на основе поверхностных сеток фронтальным методом. Поверхностные сетки с требуемыми параметрами размера и качества ячеек строятся на основе триангуляционной сетки, аппроксимирующей геометрию с заданной точностью.

Для построения сетки на аналитических поверхностях библиотека CM2 MeshTools предоставляет функцию `mesh_surface_param()`, которая позволяет построить треугольную или преимущественно четырехугольную сетку, аппроксимирующую поверхность с заданной точностью. На вход подаются следующие данные:

- Объект, реализующий доступ к основным функциям работы с аналитической поверхностью, описанным ранее, необходимым для оперирования с поверхностью при построении сетки.
- Контуры, ограничивающие поверхность, в виде массива трехмерных точек и массива связности этих точек.

Таким образом, сценарий построения неструктурированной сетки состоит из следующих этапов. Вначале строится разбиение ребер детали. Далее строится поверхностная сетка путем построения предварительной аппроксимирующей сетки и дальнейшего ее перестроения с заданными параметрами. После чего для каждого фрагмента детали сетки, построенные на гранях, сшиваются и, при необходимости, строится объемная сетка.

## Реализация

Рассмотрим общую архитектуру классов хранения и автоматической генерации сетки (рис. 2).

Объект «Деталь» (`uppPart`) содержит в себе объекты геометрии и сетки. В объекте геометрии (`uppPartGeometry`) хранится массив фрагментов, каждый из которых содержит в себе массивы граней, ребер и вершин, из которых состоит данный фрагмент.

Объект «Сетка» в общем случае представляет собой хранилище сетки. Автоматический генератор сеток оперирует с типом сетки, основанной на геометрии, `uppGeometryBasedMesh`. Объект данного класса хранит сетку в виде массивов связей геометрических элементов с конструкторами сеток.

Конструктор сеток `uppMeshElementBuilder` (рис. 3) представляет собой объект, хранящий в себе ссылку на геометрический элемент, построенную сетку и объект-алгоритм `uppMeshingAlgorithm`, с помощью которого данная сетка была сгенерирована и с использованием которого конструктор может обновить или перестроить сетку с новыми параметрами. Задача конструктора сеток – подготовить необходимые

входные данные, запустить алгоритм и обработать результат, сохранив полученную сетку.

Класс `uppMeshingAlgorithm` реализует алгоритм генерации сетки. Он предоставляет интерфейс для задания начальных параметров и метод `Execute()` для запуска генерации. На данный момент реализованы алгоритмы генерации сетки на ребре, грани и фрагменте.

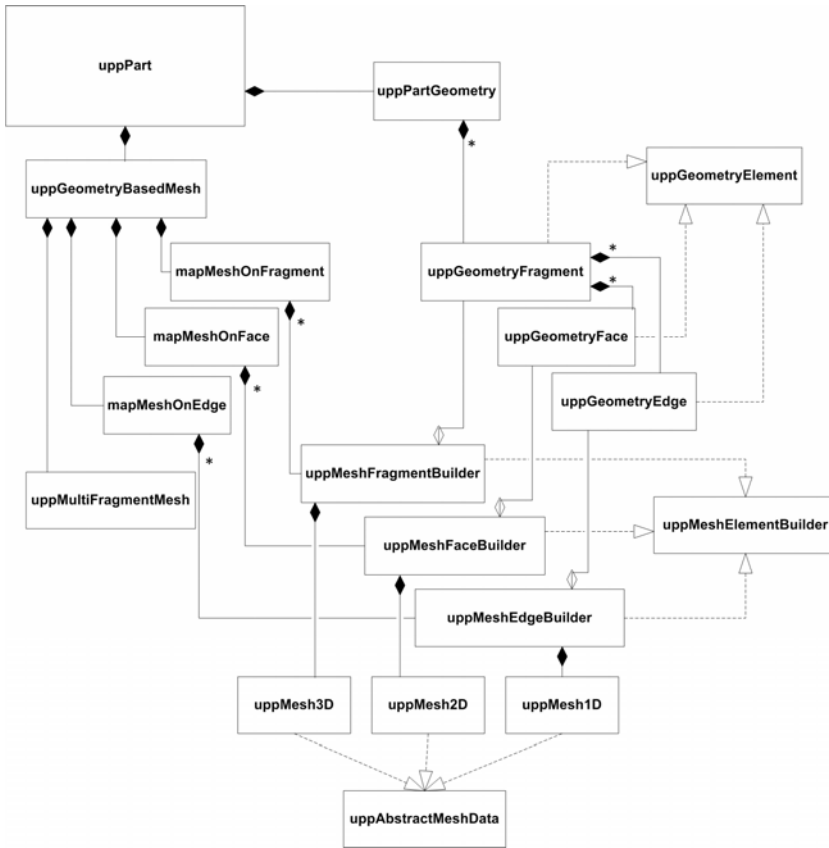


Рис. 2. Диаграмма классов: хранение сетки в классе `uppGeometryBasedMesh`

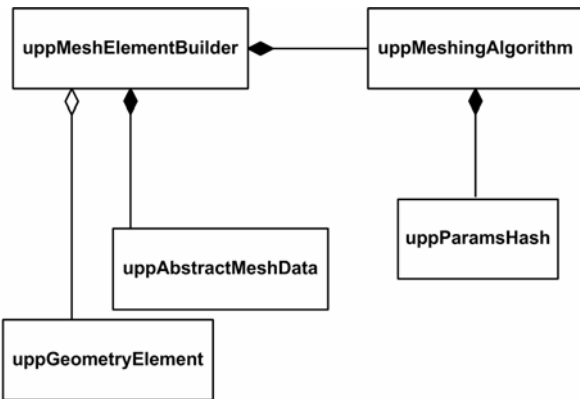


Рис. 3. Диаграмма классов: конструктор сеток

Алгоритм генерации сетки на ребре использует класс `mgEdgeDiscretization`, который по выбранному закону и параметрам разбиения выдает коэффициенты разбиения ребра. На данный момент доступны следующие законы разбиения:

- Равномерный.
- Геометрическая прогрессия.
- Двойная геометрическая прогрессия.

Алгоритмы генерации сетки на грани и фрагменте готовят необходимые для работы генератора данные, переводят их в формат генератора `MeshTools` и запускают генератор заданного типа с заданными параметрами.

Параметры алгоритмов хранятся в переменной типа `uppParamsHash` в виде массива пар: ключ – значение, где в качестве ключа выступает имя параметра. Для инициализации алгоритму передается общий массив со всеми параметрами, из которого он выбирает параметры, необходимые для своей работы.

Сеточные данные хранятся в классах-обертках на VTK структуры: `uppMesh1D`, `uppMesh2D` и `uppMesh3D` с общим интерфейсом `uppAbstractMeshData`. Это сделано для возможности реализации различных форматов хранения сеток. приме-

ром может служить хранение трехмерной сетки, построенной на многофрагментной детали.

Для хранения общей трехмерной сетки детали служит класс `urpMultiFragmentMesh`. Объект данного класса создается и хранится в сетке `urpGeometryBasedMesh`. При построении сетки на детали для каждого фрагмента строится отдельная сетка, которая хранится в объекте класса `urpMesh3D`. После завершения генерации на всех фрагментах части сетки «подшиваются» к общей сетке. При этом в объекте класса `urpMesh3D` остается ссылка на общую сетку, а при вызове метода получения сетки `GetData()` вызывается соответствующий метод класса `urpMultiFragmentMesh`, который по ID фрагмента формирует выборку соответствующих ему ячеек. Благодаря такой структуре, можно работать как с сеткой на отдельном фрагменте, так и с общей сеткой на детали. Причем в последнем случае узлы и ячейки сетки находятся в одном пространстве индексов, что облегчает поиск и выборку сеточных элементов, позволяя более удобным способом формировать наборы ячеек и узлов.

## Сценарий построения автоматической сетки

Процесс построения сетки автоматическим генератором в общем случае состоит из двух шагов.

На первом шаге задаются пользовательские параметры построения сетки на выбранных геометрических элементах. Конфигурацией сеток занимается класс `urpConfigureMeshBuilderOperation`, который по заданным параметрам и геометрическому объекту создает конструкторы и алгоритмы, заполняет их необходимыми для их работы параметрами и записывает в объект сетки `urpGeometryBasedMesh`.

При конфигурировании сеток на гранях и фрагментах автоматически задаются параметры сеток на всех элементах, необходимых для построения сетки. Так, при задании параметров сетки на грани автоматически задаются параметры на входящих в нее ребрах, при задании параметров на фрагменте – на входящих в него гранях и, соответственно, ребрах.

При задании параметров на определенном элементе – ребре, грани или фрагменте, в соответствующем конструкторе выставляется метка, что параметры заданы пользователем. Впоследствии заданные параметры изменяются только при непосредственном изменении параметров на этом элементе пользователем.

После задания всех необходимых параметров имеется два варианта построения сетки. В первом случае вызывается обновление сетки на элементах, на которых были заданы параметры (и, соответственно, созданы конструкторы сеток). Этот вариант позволяет строить сетку на отдельных фрагментах или гранях, не вызывая генерацию сетки на всей детали. Второй вариант – вызов генерации сетки на детали. В этом случае используется класс `urpMesh`

`Generator`, основная задача которого – обеспечить последовательную генерацию сетки на всех геометрических элементах детали с целью построить сетку на всей детали. Элементы, на которых не заданы параметры построения, используются параметры по умолчанию.

При необходимости можно, пропустив первый шаг, сразу вызвать генерацию сетки на детали. В этом случае по умолчанию будет построена тетраэдрная сетка, а размер элемента будет задан относительно габаритов соответствующего фрагмента.

## Автоматическое построение сетки через графический интерфейс «Логос. ПреПост»

На рис. 4 показано главное окно «Логос. ПреПост» с загруженной деталью. Слева от окна визуализации находится вспомогательная панель с функциональными кнопками. В данном случае активирован модуль «Сетка» и отображены кнопки, отвечающие за построение автоматической сетки. В группе «Авто» находятся кнопки, открывающие доступ к функционалу задания параметров построения сетки и удаления сетки на элементах, а также кнопки «Обновить сетку» и «Сгенерировать сетку». При активации режима задания параметров сетки появляется группа кнопок «Создать», позволяющих задать параметры на ребрах, гранях, фрагментах геометрии, а также построить начальное разбиение всех ребер детали по желаемому размеру получаемых элементов.

На рис. 5 показан диалог задания параметров сетки на фрагменте. В окне визуализации выделен фрагмент, на котором задаются параметры.

Кратко о параметрах построения.

*Минимальный и максимальный размеры элемента* – задают предпочтительные размеры получаемых элементов. Фактические размеры элементов могут варьироваться.

*Уровень оптимизации* определяет степень оптимизации получаемой сетки.

*Параметры поверхностной сетки* задают специфичные для поверхностных сеток параметры: угол для формирования топологических ребер при перестроении аппроксимирующей сетки, соотношение качества по размеру к качеству по форме получаемых элементов, и т. д.

*Параметры предварительной сетки* определяют точность аппроксимации аналитической поверхности и метод расчета отклонения сетки от поверхности.

Диалог задания параметров на грани аналогичен, но в списке генераторов доступны только генераторы поверхностных сеток.

На рис. 6 представлен процесс задания параметров разбиения ребра.

Можно выбрать закон разбиения и задаваемые пользователем параметры. Остальные параметры пересчитываются автоматически. После задания параметров в окне визуализации можно включить режим

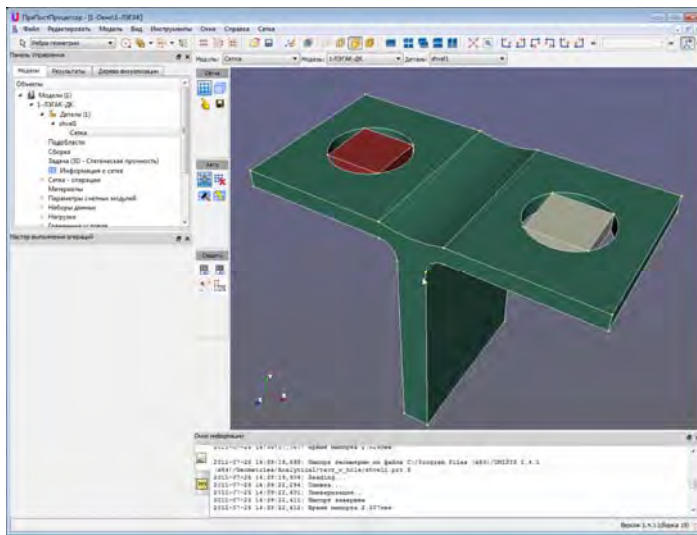


Рис. 4. Главное окно «Логос. ПреПост» с загруженной деталью

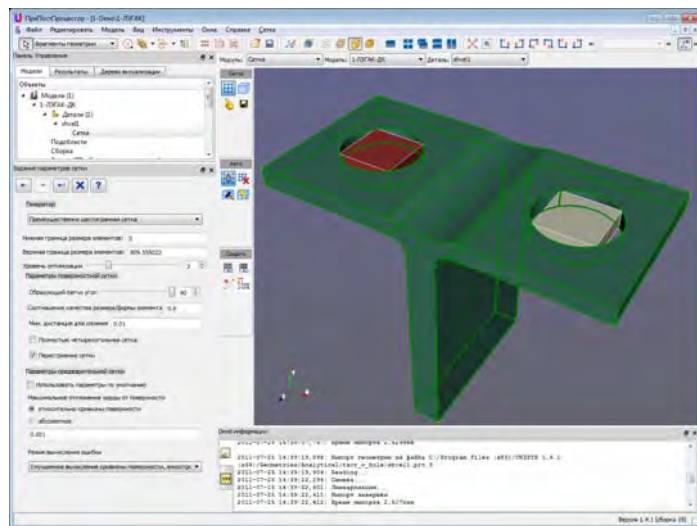


Рис. 5. Задание параметров генерации сетки на фрагменте

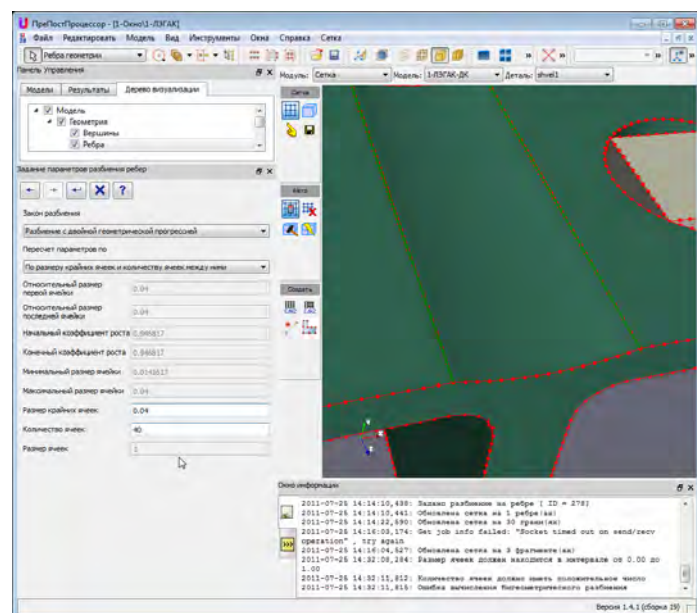


Рис. 6. Задание параметров разбиения ребра

отображения разбиений на ребрах для просмотра результата разбиения.

После задания параметров построения на всех интересующих нас элементах вызываем построение сетки нажатием на кнопку «Сгенерировать сетку».

Пример построения преимущественно шестигранной сетки с заданными разбиениями на ребрах можно увидеть на рис. 7.

Результаты построения сеток с разными параметрами на одной геометрии представлены в таблице.

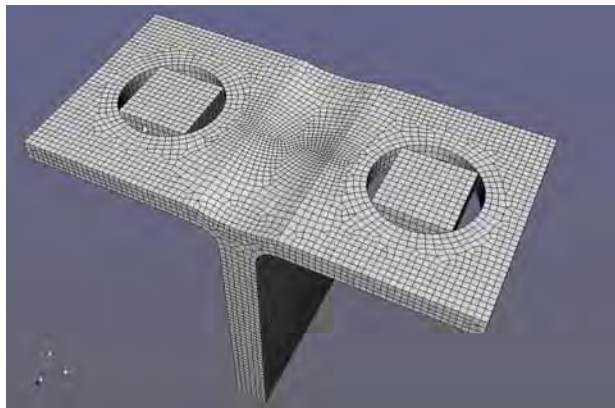


Рис. 7. Пример построения преимущественно шестигранной сетки

Результаты работы автоматического генератора неструктурированных сеток

П/п	Тип сетки	Количество ячеек	Время построения	Скорость построения, элементов/с
1	Преимущественно шестигранная	230551	1219 с	~200
2	Тетраэдрная	1 548127	152 с	~10000

### Планы развития

На данном этапе генерация сетки на многофрагментной детали происходит без учета взаимного расположения фрагментов, что делает невозможным построение согласованной сетки на контактных границах и может привести к пересечениям сеток разных фрагментов. В дальнейшем планируется ввести анализ геометрии для обнаружения контактных границ и «узких» мест между фрагментами и использовать полученные результаты для построения качественной, корректной и согласованной сетки.

Кроме того, в «Логос. ПреПост» реализована возможность построения вспомогательных геометрических элементов на гранях. Планируется использование их для задания дополнительных параметров построения сетки, таких как сгущение или разрежение сетки в определенной области, задание жестких ребер сетки и т. д.

Также в соответствии с планами ведется разработка функционала автоматической генерации сетки на балочных структурах.

### Заключение

Итак, в данный момент в «Логос. ПреПост» реализована возможность автоматической генерации неструктурированных поверхностных и объемных сеток. Реализован функционал для задания параметров размера и законов разбиения на различных геометрических элементах.

Кроме автоматического генератора неструктурированных сеток в рамках проекта «Логос. ПреПост» ведутся разработки собственных генераторов сеток. Это генератор блочно-регулярных сеток, генераторы сеток методом отсечения, экструзии, трансформации сеток, генератор сеток с пристеночными слоями и пр.

Использование автоматического генератора позволяет покрыть около 50 % задач, решаемых в рамках комплекса ЛОГОС. Реализация поддержки балочных элементов позволит покрыть еще 20–25 %. Остальные задачи рассчитаны на построение блочно-регулярных сеток, что обусловлено спецификой задач и геометрии.

### Литература

1. Owen, S. Non-Simplicial Unstructured Mesh Generation. / A Dissertation Submitted to The Department of Civil and Environmental Engineering Carnegie Mellon University, Pittsburgh, PA. U.S.A. in Partial Fulfillment of Requirements for the Degree of Doctor of Philosophy, 1999.
2. Shroeder, W. Visualization Toolkit, An Object-Oriented Approach to 3D Graphics, 3rd edition // Pearson Education, Inc. 2004.
3. CM2 MeshTools Documentation // Режим доступа: [http://www.computing-objects.com/downloads/mt420/common/mt\\_doc\\_html.zip](http://www.computing-objects.com/downloads/mt420/common/mt_doc_html.zip), свободный.