

УДК 519.6

ФУНКЦИОНАЛЬНЫЙ БЛОК ГЕНЕРАЦИИ ОБЪЕМНЫХ НЕСТРУКТУРИРОВАННЫХ СЕТОК НА ОСНОВЕ РЕГУЛЯРНОГО ДРОБНО-АДАПТИВНОГО ШАБЛОНА В МЕТОДИКЕ "ТИМ"

В. А. Никитин
(ФГУП "РФЯЦ-ВНИИЭФ", г. Саров Нижегородской области)

В РФЯЦ-ВНИИЭФ ведется разработка методики ТИМ, предназначенной для решения многомерных нестационарных задач механики сплошной среды на неструктурированных подвижных сетках произвольного вида. При сложных расчетах трехмерных задач в лагранжевых переменных, связанных с вихревыми и струйными течениями, происходят большие деформации вещества, что, в свою очередь, искажает сетку в трех направлениях. При сильных искажениях сетки локальная коррекция может занимать продолжительное время, и в таких случаях целесообразнее произвести замену сетки.

Для получения новой сетки разработан функциональный блок, позволяющий генерировать объемную неструктурированную сетку со сгущениями в локальных зонах. Генерация возможна как в ручном, так и в автоматическом режиме.

В качестве базы для генерации используется мелко-адаптивный регулярный шаблон, который преобразуется в неструктурированную сетку. При построении шаблона в качестве входных параметров используется набор контрольных объемов — пространственные геометрические фигуры, в которых производится сгущение сетки по заданным размерам.

Приведено краткое описание входных параметров, примеры построенных сеток, описывается метод построения сетки, в том числе в параллельном режиме с использованием интерфейса MPI, и даны некоторые характеристики работы функционального блока.

Ключевые слова: генератор объемной сетки, неструктурированные сетки, мелко-адаптивный шаблон, методика ТИМ.

Введение

При математическом моделировании физических процессов методом конечных разностей генераторы сеток играют ключевую роль. Они используются как для подготовки сетки для задачи на старте, так и в ходе проведения расчета, для перестроения искаженной сетки, на которой дальнейший счет невозможен либо использование которой приводит к увеличению времени расчета.

Методика ТИМ [1] предназначена для решения многомерных нестационарных задач механики сплошной среды на неструктурированных подвижных сетках произвольного вида. По этой методике проводятся сложные расчеты в лагран-

жевых переменных трехмерных задач, связанных с вихревыми и струйными течениями. Такие расчеты приводят к большим деформациям вещества, что, в свою очередь, искажает сетку в трех направлениях (рис. 1).

В связи с этим в методике ТИМ реализован широкий набор алгоритмов, которые позволяют проводить расчеты с автоматическим контролем состояния и поддержания качества сетки на протяжении всего расчета [2]. Но при большом количестве исправляемых элементов заметно увеличивается время счета задачи. В таких случаях целесообразнее построить новую сетку со сгущением в локальных зонах (там, где это необходимо для корректного описания решения) и выпол-

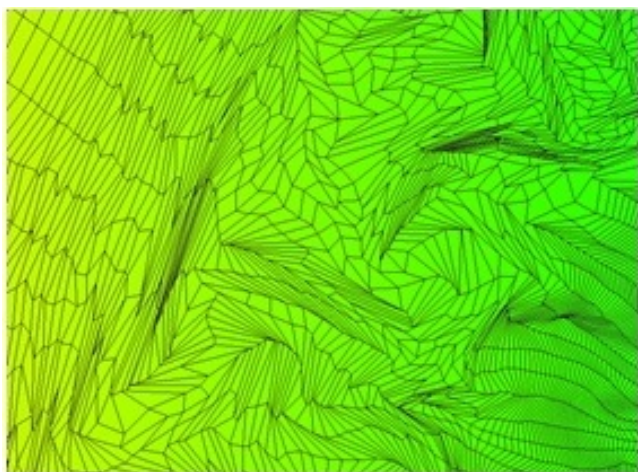


Рис. 1. Искаженная сетка

нить на нее переинтерполяцию физических величин.

Использование для указанных целей аналогичных генераторов сеток, применяемых в других методиках РФЯЦ-ВНИИЭФ [3, 4], достаточно проблематично из-за специфики методики ТИМ. Нет также уверенности, что алгоритмы имеющихся генераторов можно эффективно реализовать в параллельном режиме с использованием интерфейса MPI.

С целью решения указанной проблемы для методики ТИМ создан функциональный блок (модуль), позволяющий генерировать объемную неструктурированную сетку как в ручном режиме с заданием требуемых параметров при подготовке сетки для проведения расчета, так и в автоматическом, когда подготовка параметров генерации проводится без участия пользователя, по данным об исходной сетке.

Входные параметры

Модуль генерации объемной неструктурированной сетки на основе регулярного мелко-адаптивного шаблона позволяет создавать сетку, состоящую из многогранных ячеек, имеющих форму параллелепипедов. Каждая из шести граней ячейки может быть дополнительно разбита на четыре части при формировании соседних ячеек меньшего размера. Области сгущения сетки задаются с помощью набора контрольных объемов или на основании данных об исходной сетке.

Контрольный объем представляет собой геометрическую фигуру в пространстве, внутри ко-

торой будут формироваться ячейки требуемого размера. Реализовано шесть типов контрольных объемов:

- 1) параллелепипед — задается как габарит, т. е. точками пространства с минимальными и максимальными координатами;
- 2) цилиндр — задается двумя точками пространства, формирующими ось цилиндра, и радиусом основания;
- 3) усеченный конус — задается двумя точками пространства, формирующими ось конуса, и двумя радиусами оснований;
- 4) сфера — задается координатами центра и радиусом;
- 5) труба — задается двумя точками пространства, формирующими ось трубы, и радиусами основания — внешним и внутренним;
- 6) комбинированный контрольный объем — задается двумя типами контрольных объемов. Внутри первого заданного объема строится адаптивная сетка с требуемым размером до тех пор, пока не встретится второй заданный объем. Адаптивная сетка внутри второго контрольного объема не строится (рис. 2).

При генерации сетки допускается любое количество и сочетание контрольных объемов (рис. 3). Возможно также их пересечение и наложение. В этом случае алгоритм модуля работает так, чтобы в пространстве пересечения стремиться к созданию ячеек из пересекающихся контрольных объемов с наименьшими требуемыми размерами.

Для работы модуля необходимо задать следующую информацию:

- 1) габарит, в котором необходимо построить сетку, — указывается заданием точек пространства с минимальными и максимальными координатами;

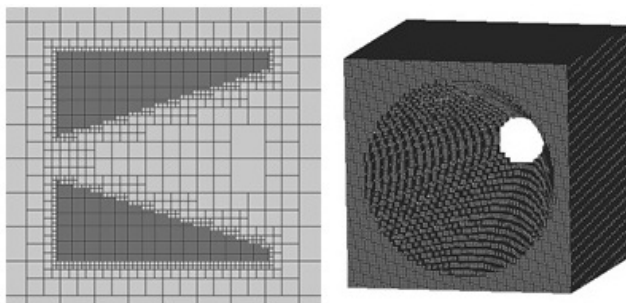


Рис. 2. Построение объема в виде комбинации куба и усеченного конуса

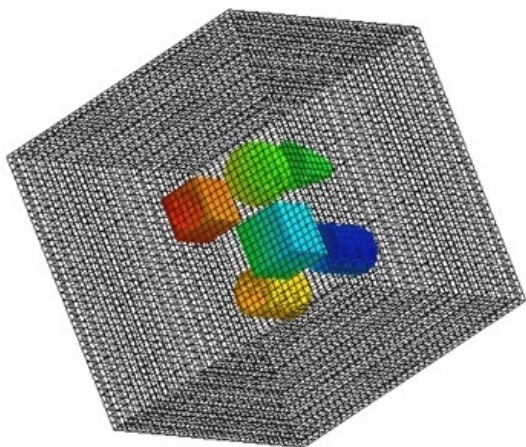


Рис. 3. Пример построения различных контрольных объемов

- 2) базовый размер ячейки — требуемая длина ребра ячейки исходной сетки, т. е. сетки, с которой начинается работа модуля;
- 3) номера граничных условий — дополнительная информация для граней ячеек, лежащих на внешних поверхностях, необходимая для определения, к какой из шести внешних поверхностей принадлежит грань;
- 4) набор контрольных объемов, которые задают область пространства со своей геометрией и требуемым в этой области размером ячеек.

Автоматическая подготовка начальных данных

На текущий момент основная задача модуля генерации неструктурированной трехмерной сетки — это создание в методике ТИМ новой сетки на основе данных о старой деформированной сетке в ходе счета задачи. Алгоритмы, анализирующие качество сетки, дают сигнал о необходимости перестроения счетной сетки. Производится подготовка параметров для генерации новой сетки: вычисляются габариты ячеек старой сетки и определяется размер для новых ячеек. Вокруг каждой ячейки перестраиваемой неструктурированной сетки формируется контрольный объем *параллелепипед* с габаритами этой ячейки. В качестве требуемого размера задается исходный размер ячейки. Набор контрольных объемов и базовый размер ячейки исходной сетки передаются в модуль генерации сетки. В модуль также передаются габариты задачи.

Генерация сетки

Генерация объемной сетки разбивается на два этапа: 1) генерация трехмерной матричной сетки с возможностью вставки адаптивной сетки в ячейки с целью получения дробно-адаптивного шаблона; 2) преобразование дробно-адаптивного шаблона в неструктурированную сетку формата ТИМ [5] на этапе записи файла-разреза в формате ЕФР [6].

Этап генерации трехмерной матричной сетки разбивается на несколько шагов. На первом шаге формируется матричная сетка по заданным параметрам. Выбор такого формата представления сетки обусловлен простотой поиска соседей для элементов сетки (узлов и ячеек), а также минимальным набором дополнительной информации для доступа к элементам.

На следующем шаге проверяется пересечение каждой ячейки матричной сетки с набором заданных контрольных объемов. Если ячейка имеет пересечение с одним из объемов, проводится анализ на необходимость дробления этой ячейки. Если размер ячейки больше требуемого в контрольном объеме, то она дробится. Заметим, что как такового дробления не происходит, т. е. матричная сетка никаких изменений не претерпевает. Вместо этого формируется новая, аналогичная исходной матричная сетка, габаритами и местоположением в пространстве совпадающая с обрабатываемой ячейкой и имеющая $2 \times 2 \times 2$ ячеек. В ячейку исходной сетки записывается ссылка на новую сетку. Нумерация узлов такой ячейки согласовывается с нумерацией крайних узлов новой сетки. Для новой сетки также формируется информация о соседних ячейках исходной сетки и аналогичных матричных сетках в этих ячейках.

В результате обработки всех ячеек исходной сетки появляется набор матричных сеток, которые образуют *первый уровень адаптивности*. Далее анализируются ячейки из этого набора сеток на пересечение с контрольными объемами. Если есть ячейка, удовлетворяющая критериям дробления, то для нее, аналогично описанному выше методу, формируется матричная сетка. Таким образом формируется набор матричных сеток *второго уровня адаптивности*.

Третий и последующий уровни адаптивности формируются аналогично. Количество уровней адаптивности не ограничено (рис. 4). Совокупность матричных сеток всех уровней образует дробно-адаптивный шаблон.

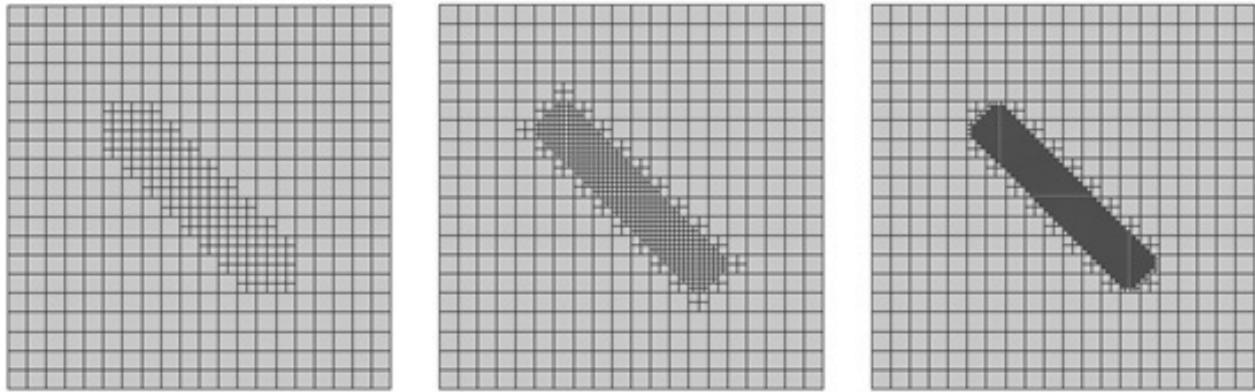


Рис. 4. Формирование первых трех уровней адаптивности

После построения всех уровней адаптивности происходит сглаживание переходов между ними (рис. 5). Для этого анализируются все не подвергшиеся дроблению ячейки дробно-адаптивного шаблона. Для каждой такой ячейки формируется список ячеек всех уровней адаптивности, которые соседствуют с обрабатываемой через грань, после чего анализируются все ячейки из этого списка.

Если в списке есть ячейка с уровнем адаптивности, превышающим уровень обрабатываемой ячейки больше чем на один, то обрабатываемая ячейка дробится аналогично описанному выше методу. Такой анализ проводится до тех пор, пока максимальная разница между уровнями адаптивности не подвергшихся дроблению ячеек и их соседей по граням не станет равной единице (рис. 6).

Формирование неструктурированной объемной сетки происходит в формате ЕФР [6] на стадии записи файла-разреза. Для приведения дробно-адаптивного шаблона к требуемому формату необходимо сформировать информацию о

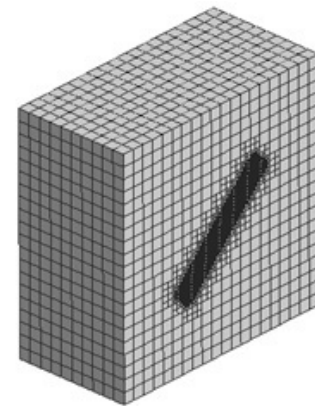


Рис. 6. Результат генерации дробно-адаптивного шаблона

гранях — номера разделяемых ячеек и список узлов. Для каждой ячейки формируется список граней, из которых она состоит.

Нумеруются только все не подвергшиеся дроблению ячейки всех уровней адаптивности. Это позволяет сразу формировать для каждой грани конечную информацию о разделяемых ею ячейках.

При формировании списка узлов для грани необходимо соблюдать правильный порядок их следования. Согласно формату ЕФР узлы грани должны нумероваться в порядке против часовой стрелки, если смотреть на эту грань из первой разделяемой ячейки.

Обработка сетки начинается с последнего уровня адаптивности. Таким образом, ячейки меньшего размера формируют набор граней, относящийся к одной из сторон более крупной ячейки. Это позволяет экономить время на вычислениях соседства между ячейками и опреде-

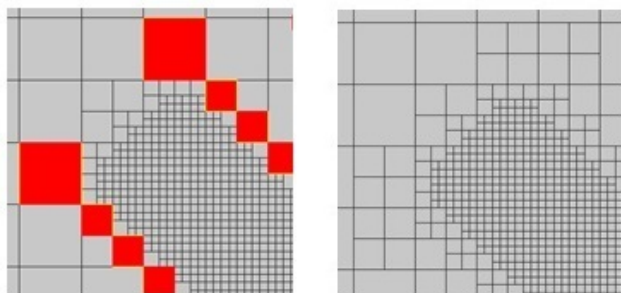


Рис. 5. Сглаживание переходов между уровнями адаптивности

лении количества граней для конкретной ячейки (рис. 7).

Так как преобразуемая сетка является матричной, то при формировании узлов неструктурированной сетки координаты вычисляются через индексы и смещение относительно точки с минимальными координатами габарита матричной сетки.

После окончания работы генератора результат сохраняется в файле формата ЕФР [6], который передается в методику ТИМ для загрузки и переинтерполяции величин со старой сетки на новую (рис. 8).

Все операции по генерации новой сетки проводятся полностью в автоматическом режиме, без остановки счета задачи.

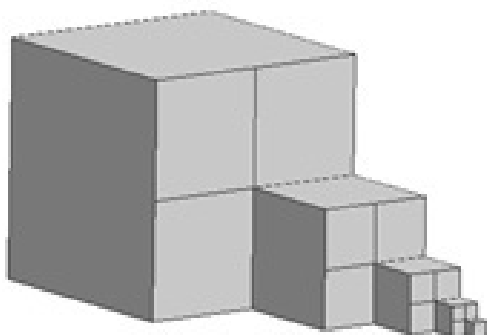


Рис. 7. Формирование ячеек

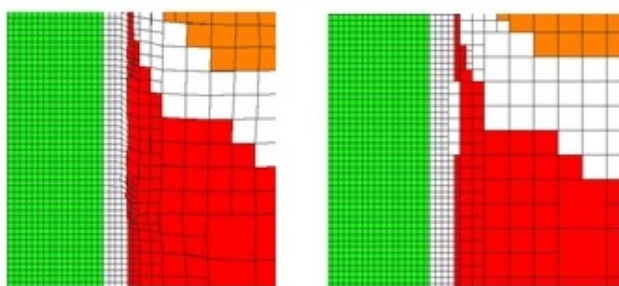


Рис. 8. Перестроение фрагмента сетки: *a* — перестраиваемая; *b* — новая

Работа в параллельном режиме

Для обеспечения функционирования модуля в параллельном режиме была проведена работа по декомпозиции задачи генерации сетки по процессорам с использованием интерфейса MPI.

Тип декомпозиции выбран матричный, так как он позволяет оптимально разбить сетку на фрагменты и лучше сбалансировать нагрузку на процессоры. Разбиение происходит по основной матричной сетке в трех направлениях до построения уровней адаптивности (рис. 9).

При разбиении также учитывается нагрузка на каждый процессор, связанная с генерацией уровней адаптивности. Часть нагрузки может быть передана соседним процессорам через межпроцессорную границу. Но при этом размер минимальной рассчитываемой на каждом процессоре области не может быть меньше одного слоя ячеек стартовой матричной сетки (рис. 10).

Межпроцессорная граница каждого фрагмента дополняется фиктивным слоем ячеек. В этот слой помещается информация об уровнях адаптивного соседнего фрагмента при синхронизации межпроцессорных границ для сглаживания переходов. В построении итоговой сетки

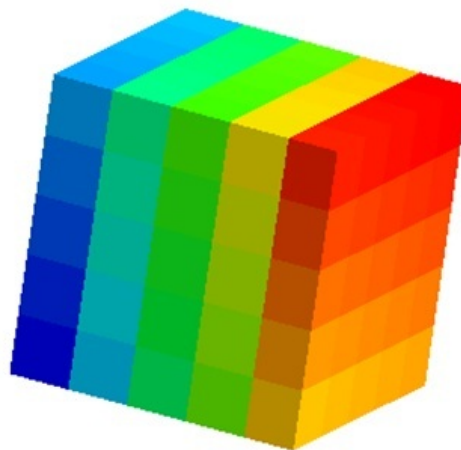


Рис. 9. Пример матричной декомпозиции

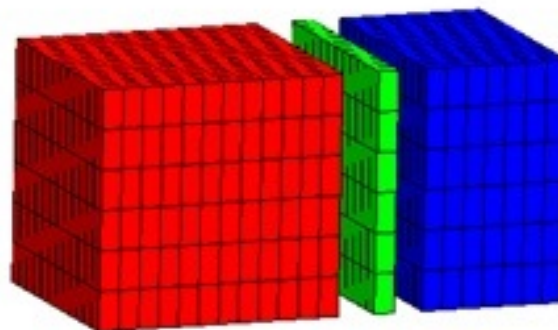


Рис. 10. Пример балансировки нагрузки в одном из направлений пространства

этот слой не участвует. Для синхронизации информации между процессорами каждому фрагменту присваивается один из двух цветов, так чтобы они чередовались в шахматном порядке (рис. 11).

Синхронизация происходит в три подхода, по одному на каждое из трех направлений пространства. При каждом подходе синхронизация происходит за два шага. Если в рамках направления выделить условно левые и правые границы фрагментов, то, исходя из раскраски на рис. 11, на первом шаге происходит синхронизация правых синих границ с левыми красными, на втором шаге — наоборот (рис. 12).

Таким образом, синхронизация межпроцессорных границ выполняется параллельно. При синхронизации передается информация об уровнях адаптивности на крайних слоях ячеек, примыкающих к межпроцессорной границе. По этой информации проводится сглаживание переходов между уровнями. Синхронизация проводится на итерациях до тех пор, пока переходы во всех фрагментах не будут выровнены. После этого фрагменты дробно-адаптивного шаблона пре-

образуются во фрагменты неструктурированной сетки без учета слоев фиктивных ячеек.

Чтобы увязать все фрагменты в единую итоговую сетку, проводится перенумерация всех элементов сетки (узлов, граней, ячеек) с использованием единой глобальной нумерации. Для этого определяется количество элементов в каждом фрагменте. Эта информация рассылается на все процессоры, что позволяет вычислить стартовые глобальные номера для каждого фрагмента. При этом учитывается дублирование некоторых элементов сетки на межпроцессорной границе — их номера должны быть одинаковыми для левой и правой границ.

Каждый результирующий фрагмент сохраняется в своем файле формата EFR и формирует многофайловое представление итоговой сетки, что позволяет провести запись в параллельном режиме.

Дополнительные возможности модуля

В рамках описываемого модуля реализованы дополнительные возможности, улучшающие качество результата и расширяющие область применения.

Одной из таких возможностей является увеличение минимальной зоны перехода от одного уровня адаптивности к другому. По умолчанию минимальный переход между уровнями адаптивности осуществляется за одну ячейку (см. рис. 6). Пользователю предоставлена возможность задавать другое количество ячеек при переходе (рис. 13).

Еще одной дополнительной возможностью является конвертация сеточных данных методик, по которым проводятся расчеты на матричных

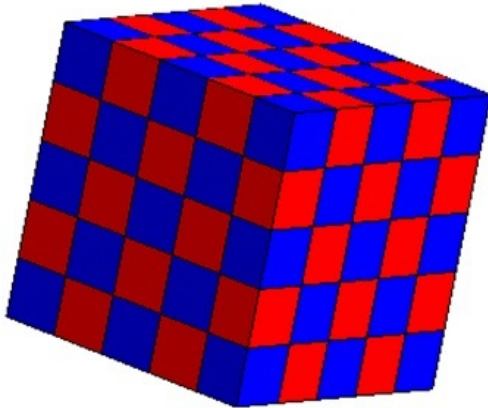


Рис. 11. Пример раскраски фрагментов для синхронизации межпроцессорных границ

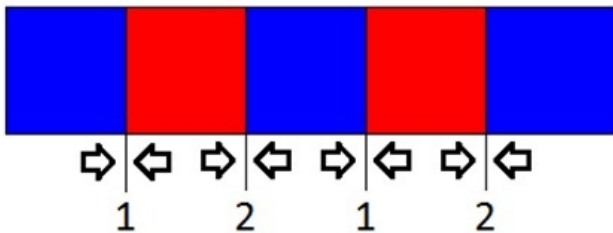


Рис. 12. Пример межпроцессорных обменов в одном из направлений пространства

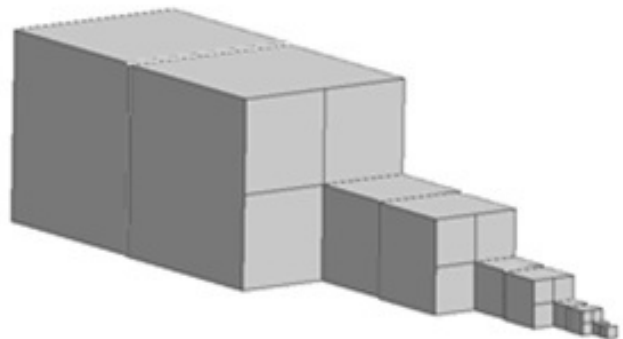


Рис. 13. Пример перехода между уровнями в две ячейки

сетках, в неструктурированное представление для методики ТИМ. Используемый в генерации неструктурированной сетки дробно-адаптивный шаблон без уровней адаптивности задается по параметрам конвертируемой матричной сетки. После формирования неструктурированной сетки при наличии связи через шаблон между номерами ее элементов и индексами элементов матричной сетки данные (координаты, физические величины, дополнительные массивы и т. д.) переносятся на новую сетку.

Описанные дополнительные возможности поддерживаются при работе модуля в параллельном режиме.

Результаты тестирования

Для оценки работоспособности функционального блока было проведено тестирование на одной из задач, рассчитываемых по методике ТИМ. В ходе расчета из методики в автоматическом режиме вызывался функциональный блок для подготовки новой сетки. Цель теста — анализ зависимости времени работы модуля от количества генерируемых ячеек. Построение сетки проводилось, начиная с 60 млн ячеек, с 5 уровнями адаптивности. Анализ результатов показал, что время построения практически линейно зависит от числа ячеек (рис. 14).

Для проверки работоспособности модуля в параллельном режиме было проведено тестирование на 150 процессорах без построения уровней адаптивности (рис. 15). Как видно из графика, в целом зависимость количества генерируемых ячеек от времени близка к линейной. Отклонения от прямой связаны с аппаратными задержками на многопроцессорной ЭВМ.

На рис. 16 приведена зависимость времени генерации сетки, содержащей 62 млн ячеек и 6

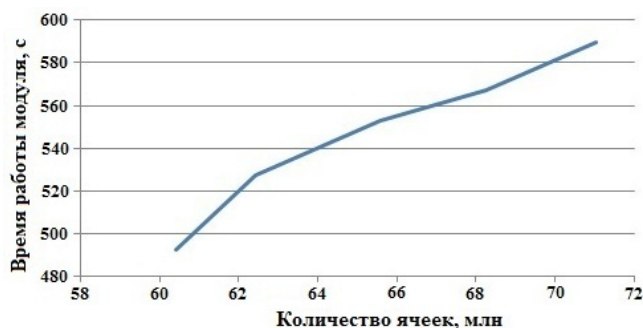


Рис. 14. Зависимость времени работы модуля от количества ячеек



Рис. 15. Зависимость времени работы модуля в параллельном режиме от количества ячеек

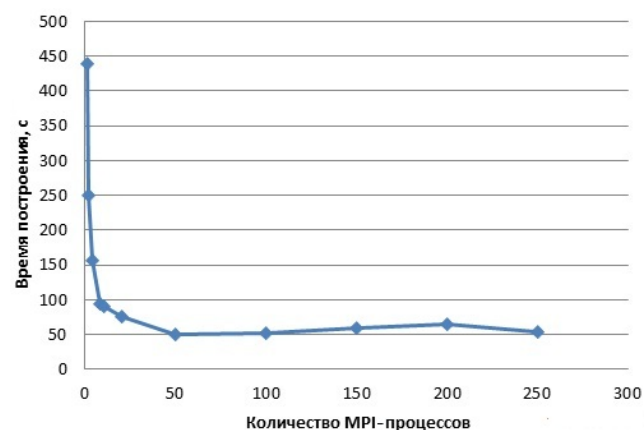


Рис. 16. Зависимость времени работы модуля в параллельном режиме от количества MPI-процессов

уровней адаптивности, от числа используемых MPI-процессов.

Заключение

Разработанный функциональный блок расширил возможности методики ТИМ, позволяя генерировать объемную неструктурированную сетку в ручном и автоматическом режимах с использованием многопроцессорных ЭВМ. С реализацией автоматического режима получена возможность проводить по методике ТИМ сложные расчеты с перестройкой всей счетной сетки без остановки счета задачи. Дополнительные возможности расширили область применимости модуля, сделав его более универсальным инструментом.

Список литературы

1. Соколов С. С., Панов А. И., Воропинов А. А., Новиков И. Г., Соболев И. В., Ялозо А. В. Методика ТИМ расчета трехмерных задач механики сплошных сред на неструктурированных многогранных лагранжевых сетках // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2005. Вып. 3. С. 37–52.
Sokolov S. S., Panov A. I., Voropinov A. A., Novikov I. G., Sobolev I. V., Yalozo A. V. Metodika TIM raschyeta tryekhmernykh zadach mekhaniki sploshnykh sred na nestrukturovannykh mnogogrannykh lagranzhevyykh setkakh // Voprosy atomnoy nauki i tekhniki. Ser. Matematicheskoe modelirovanie fizicheskikh protsessov. 2005. Vyp. 3. S. 37–52.
2. Новиков И. Г., Панов А. И., Соколов С. С. Способ коррекции нерегулярной лагранжевой сетки методом наложения дифференцируемых связей // Журнал вычисл. мат. и мат. физ. 2005. Т. 45, № 8. С. 1487–1500.
Novikov I. G., Panov A. I., Sokolov S. S. Sposob korrektsii neregulyarnoy lagranzhevoy setki metodom nalozheniya differentsiruemykh svyazey // Zhurnal vychisl. mat. i mat. fiz. 2005. T. 45, № 8. S. 1487–1500.
3. Смолкина Д. Н., Борисенко О. Н., Черенкова М. В., Гиниятуллина А. Г., Кузьменко М. В., Чулманов Н. В., Потехина Е. В., Попова Н. В., Турусов М. Р. Автоматический генератор неструктурированных многогранных сеток в препроцессоре пакета программ "ЛОГОС" // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2018. Вып. 2. С. 25–39.
Smolkina D. N., Borisenko O. N., Cherenkova M. V., Giniyatullina A. G., Kuzmenko M. V., Chukhmanov N. V., Potekhina E. V., Popova N. V., Turusov M. R. Avtomaticheskii generator nestrukturovannykh mnogogrannykh setok v preprotssessore paketa programm "LOGOS" // Voprosy atomnoy nauki i tekhniki. Ser. Matematicheskoe modelirovanie fizicheskikh protsessov. 2018. Vyp. 2. S. 25–39.
4. Потехина Е. В., Ребров С. В., Логинов Д. В. Методы расчета трехмерных разностных сеток и распределения частиц в препроцессоре 3D-РНД // Тр. межд. науч.-тех. конф. "Компьютерное моделирование 2009". Санкт-Петербург, 2009. С. 158–164.
Potekhina E. V., Rebrov S. V., Loginov D. V. Metody raschyeta tryekhmernykh raznostnykh setok i raspredeleniya chastits v preprotssessore 3D-RND // Tr. mezhd. nauch.-tekh.konf. "Kompyuternoe modelirovanie 2009". Sankt-Peterburg, 2009. S. 158–164.
5. Воропинов А. А., Соколов С. С., Панов А. И., Новиков И. Г. Формат для описания нерегулярной многогранной сетки произвольной структуры в методике ТИМ // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2007. Вып. 3-4. С. 55–63.
Voropinov A. A., Sokolov S. S., Panov A. I., Novikov I. G. Format dlya opisaniya neregulyarnoy mnogogrannoy setki proizvolnoy struktury v metodike TIM // Voprosy atomnoy nauki i tekhniki. Ser. Matematicheskoe modelirovanie fizicheskikh protsessov. 2007. Vyp. 3-4. S. 55–63.
6. Олесницкая К. К., Антипин И. А., Петрова М. А. Коллективный доступ к файловым данным на вычислительных системах с различной архитектурой в библиотеке ЕФР // XVII Межд. конф. "Супервычисления и математическое моделирование": Сб. докладов. Саров: РФЯЦ-ВНИИЭФ, 2019. С. 392–399.
Olesnitskaya K. K., Antipin I. A., Petrova M. A. Kollektivny dostup k faylovym dannym na vychislitelnykh sistemakh s razlichnoy arkhitekturoy v biblioteke EFR // XVII Mezhd. konf. "Supervychisleniya i matematicheskoe modelirovanie": Sb. dokladov. Sarov: RFYaTs-VNIIEF, 2019. S. 392–399.

Статья поступила в редакцию 06.11.20.

A FUNCTIONAL MODULE IN TIM CODE FOR GENERATION OF UNSTRUCTURED VOLUME GRIDS USING A REGULAR ADAPTIVE REFINEMENT PATTERN / Nikitin V. A. (FSUE "RFNC-VNIIEF", Sarov, Nizhniy Novgorod region).

RFNC-VNIIEF has been developing the TIM code to solve multidimensional unsteady problems of continuum mechanics on unstructured moving arbitrarily-shaped grids. In complex simulations for 3D problems in Lagrangian variables, with eddy flows and jet flows, large material deformations causing distortions of the grid in three directions take place. Local corrections in a severely distorted grid may consume much time and, therefore, it is reasonable to change the grid.

To generate a new grid, a functional module was developed that allows generating an unstructured locally-condensed 3D grid. Such grid can be generated both manually and automatically.

A regular adaptive refinement pattern is used for the grid generation and transformed into an unstructured grid. A set of control volumes, i.e. solid geometric figures in which the grid is condensed according to the specified sizes, is used as input parameters.

A brief description of input parameters and examples of the generated grids are presented, a grid generating method is described including that used in parallel mode with MPI interface, and some functional module performance data is given.

Keywords: generator of a volume grid, unstructured grids, an adaptive refinement pattern, the TIM code.
