

В. В. Новокрещенов

ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ В ЗАДАЧАХ ДИНАМИКИ И ПРОЧНОСТИ ТУРБОМАШИН

Описаны преимущества суперэлементного подхода к решению задач статики и динамики деталей и узлов турбомашин. Приведено описание программного комплекса, решающего указанные задачи с помощью метода конечных элементов и суперэлементов. Приведены результаты тестирования комплекса.

E-mail: novvit@mail.ru

Ключевые слова: суперэлемент, программный комплекс, прочностной анализ.

Введение. Последнее десятилетие характеризуется принципиальной сменой парадигмы наращивания мощности вычислительных средств. Конец XX века ознаменовался фактическим достижением границ миниатюризации техпроцесса изготовления электронных микросхем и дальнейшее увеличение мощности происходит ныне за счет использования многоядерных и многопроцессорных систем на всех уровнях — от мобильных телефонов до суперкомпьютеров. Кроме того, в обиход вошли инженерные вычисления с использованием GPU (Graphics Processing Unit), фактически представляющие собой миниатюрные суперкомпьютеры, построенные по схеме SIMD (Single Instruction Multiple Data).

Производители коммерческих программных комплексов для расчетов, связанных с инженерной деятельностью в областях газо- и гидродинамики, прочности, теплофизики и т.д., приняли вызов и начали активно внедрять возможности параллельных вычислений в свои продукты. Однако жесткая конкуренция на рынке соответствующего программного обеспечения (ПО) не позволяет им взять паузу и полностью изменить архитектуру комплекса, чтобы наиболее полно вписаться в изменившуюся архитектуру аппаратного обеспечения. Производители ПО вынуждены по сути применять тактику “заплаток”, заменяя от версии к версии то один, то другой вычислительный блок. В результате сложилась ситуация, когда среди крупных игроков рынка специализированного ПО нет систем, полностью и изначально построенных на суперэлементной основе.

Одно из наиболее успешных применений эта технология может найти в области прочностных и теплофизических инженерных расчетов, которые традиционно используют метод конечных элементов (МКЭ), в частности, при расчете деталей и узлов турбомашин. Чтобы пояснить актуальность рассматриваемого вопроса именно в указанной области, рассмотрим результаты тестирования производительности программного комплекса Ansys 13 при расчете узла авиационного

двигателя. Трехмерная модель содержала 392 608 неизвестных перемещений; расчет выполнялся в геометрически нелинейной постановке; итерационный процесс сходился за 16 итераций; оперативной памяти компьютера было достаточно для размещения всех необходимых данных модели; рабочая частота процессора 2880 МГц. Результаты тестирования приведены в таблице.

Таблица

Тестирование Ansys 13 на многоядерном ПК с GPU

| | Без GPU, с | C GPU Nvidia Tesla C2070, с |
|---------------------------------------|------------|-----------------------------|
| Расчет с использованием 1-го ядра GPU | 4009 | 1848 |
| Расчет с использованием 4-х ядер GPU | 1870 | 1344 |

Таким образом, применение GPU на рабочих станциях при решении задач прочности следует признать нецелесообразным на текущем этапе, принимая во внимание, что модуль GPU по цене соответствует трем 4...6-ядерным ПК. Объясняется это в основном тем, что алгоритмы МКЭ редко используют массивно параллельные вычисления и плохо переносятся на GPU.

Настоящая статья посвящена методам параллельных вычислений с применением суперэлементного (СЭ) подхода при решении задач прочности и теплопроводности в деталях и узлах турбомашин. Описан создаваемый на основе указанных методик программный комплекс и приведены некоторые результаты тестирования его производительности.

Способы повышения эффективности вычислений при использовании СЭ. Распараллеливание алгоритмов метода конечных элементов для кластеров с использованием СЭ основано на идее разбиения области решения задачи на набор непересекающихся подобластей, в каждой из которых вводится своя сетка конечных элементов таким образом, что узлы, попадающие на смежные границы областей, совпадают. Каждая из подобластей образует суперэлемент. Суперэлементы могут быть образованы путем поглощения ими других суперэлементов, в этом случае создается иерархическая структура СЭ.

Распараллеливание алгоритма осуществляется путем распределения суперэлементов по клиентам вычислительной сети. Объем данных, передаваемых между клиентами в процессе решения задачи статического анализа конструкций, имеет порядок n_e^2 , где n_e — число внешних степеней свободы СЭ (образованных внешними узлами).

Необходимо разбивать исходную область решения задачи на СЭ таким образом, чтобы минимизировать суммарный объем передава-

емого по сети трафика $\left(\sum_e n_e^2 \rightarrow \min \right)$. Обычно вместо задачи $\sum_e n_e^2 \rightarrow \min$ решают задачу $\sum_e n_e \rightarrow \min$, стремясь при этом создавать СЭ примерно равных размеров для обеспечения оптимальной загрузки клиентов вычислительной сети.

Количество суперэлементов иногда определяется спецификой задачи и не может быть изменено. На разбиение также могут накладываться условия, диктуемые некоторыми соображениями касательно повышения эффективности алгоритма решения задачи. Например, если в рассматриваемой конструкции могут быть выделены одинаковые подобласти, то вполне естественно взять их в качестве СЭ задачи. Матрицы жесткости будут для таких СЭ одинаковыми и их вычисление и факторизацию потребуется произвести только один раз.

В конструкции, относительно части которой предполагается, что она будет работать в пластической области, в то время как остальная конструкция остается упругой, разбиение на СЭ производится таким образом, чтобы суперэлементы целиком лежали в одной из обозначенных областей. На каждом шаге итерационного решения задачи необходимо пересчитывать только касательные матрицы жесткости для СЭ, лежащих в пластической области.

Существенный выигрыш во времени решения контактных задач теории упругости можно получить, построив разбиение таким образом, чтобы узлы, по которым осуществляется контакт, лежали на границе суперэлемента. Тогда итерационный алгоритм решения контактной задачи будет оперировать только с матрицами отцовских суперэлементов, содержащими контактные узлы в качестве внутренних.

В задачах оптимизации конструкций желательно построить разбиение так, чтобы при малых возмущениях параметров модели возникала необходимость пересчитывать матрицы как можно меньшего количества СЭ и чтобы эти матрицы были по возможности небольшого размера.

Структура комплекса. Создаваемый комплекс параллельных вычислений имеет структуру, показанную на рис. 1.

Вычислительная среда образована набором вычислительных узлов в общем случае различной производительности, объединенных в сеть, позволяющую реализовать взаимодействие “каждый с каждым”. Каждая из программных единиц предполагается выполняющейся на отдельном узле, хотя это условие и не обязательно. Клиенты в начале своей работы соединяются с сервером и регистрируют свои данные. Один из клиентов загружает базу данных (БД) задачи, либо скрипт с описанием задачи, на основе которого сам генерирует БД, затем

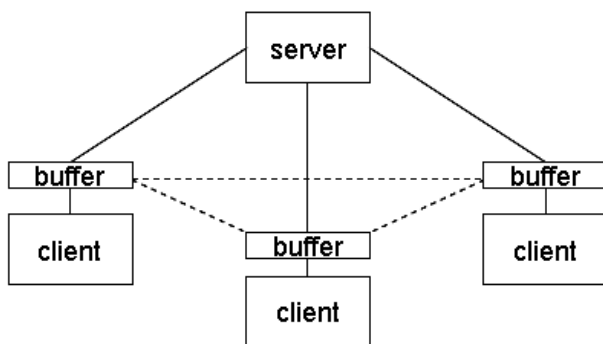


Рис. 1. Структура комплекса

дает информацию о БД задачи на сервер. После завершения рассылки сервером БД задачи по всем подключенным на данный момент к системе клиентам один из них формирует задачи верхнего уровня и отправляет их на сервер.

Задача представляет собой набор следующих данных:

- идентификаторы данных (в соответствии с БД задачи) и сами данные, необходимые клиенту для выполнения расчета;
- текстовую строку, задающую последовательность действий для клиента (тело задачи), а также определяющую набор данных (в виде их идентификаторов), которые необходимо вернуть клиенту, поставившему задачу;
- текстовую строку с описанием задачи.

Данное представление задачи позволяет легко работать с иерархически организованными структурами суперэлементов, поскольку тело задачи может содержать команды генерации задач следующего уровня.

Гибкость и эффективность системы централизованного распределения вычислений на сервере базируется на механизме описания задач. Описание может содержать следующие поля данных:

- требование выполнить задачу на том же клиенте, что и задачу X (позволяет резко снизить объем передаваемого трафика);
- требование выполнить задачу только после завершения списка задач $X_1 \dots X_n$ (задача считается завершенной, когда клиент, ее поставивший, получит результаты ее выполнения);
- группирование задач (позволяет производить действия по окончании выполнения всех задач из группы);
- информацию о трудоемкости задачи;
- информацию о приоритете задачи;

Каждый из клиентов имеет буфер, необходимый для обмена данными с сервером и другими клиентами в то время, пока вычислительный модуль клиента выполняет расчет.

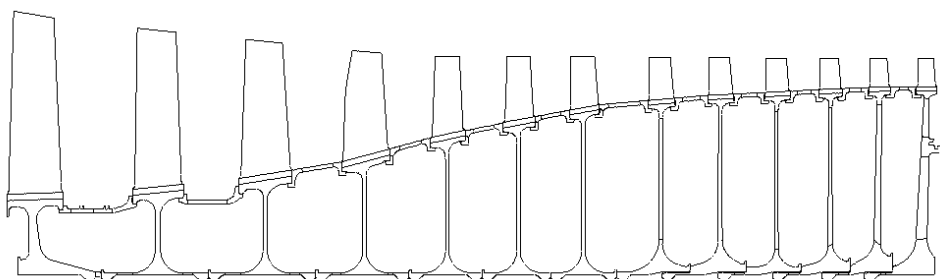


Рис. 2. Расчетная схема ротора КВД

Одной из важных задач при проектировании распределенной вычислительной системы является обеспечение ее отказоустойчивости. Программный или аппаратный сбой на одном или нескольких клиентах не должны приводить к необходимости перезапуска расчета. Более того, система должна автоматически отслеживать появление сбоев и принимать меры для перераспределения данных и задач, обеспечивать корректное повторное вычисление необходимых для дальнейших расчетов утерянных данных.

Решение задачи определения НДС ротора КВД авиационного газотурбинного двигателя (ГТД). Созданный параллельный вычислительный комплекс был применен для расчета напряженно-деформированного состояния (НДС) дисков компрессора высокого давления (КВД) в двумерной постановке. Расчетная схема конструкции показана на рис. 2.

Каждый из 13 дисков и 12 распорных проставок имели возможность быть загруженными для расчета по отдельности. Подобный способ сборки конструкции “по кубикам” позволил легко разбить расчетную схему на суперэлементы оптимальным образом и оказался очень полезным для тестирования системы путем постепенного наращивания размерности задачи. Контактное взаимодействие между деталями не моделировалось, сетки различных деталей сопряжены с помощью множителей Лагранжа.

После построения достаточно качественной сетки расчетная схема конструкции без суперэлементов содержала 185 465 неизвестных. Для хранения матрицы жесткости всей конструкции потребовалось 369 Мб. Относительно небольшой объем передаваемых по сети данных (12,8 Мб), обусловленный правильным выбором способа разбиения на суперэлементы, позволил добиться хороших показателей ускорения решения задачи за счет ее распараллеливания. Иллюстрирующий сказанное график приведен на рис. 3.

Расчет собственных частот и форм колебаний блиска. Задача расчета собственных колебаний облопаченных дисков ГТД и построения частотных диаграмм очень важна при их проектировании. Не-

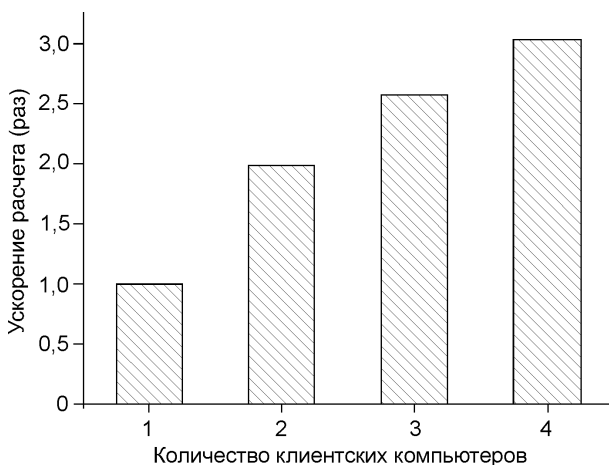


Рис. 3. Уменьшение времени расчета за счет распараллеливания

обходимо исключить совпадения на установившихся режимах работы собственных частот и частот, возбуждаемых в двигателе за счет вращения ротора, взаимодействия лопаточных венцов и многих других факторов. Применение в современных конструкциях разночастотных лопаток и наличие смешанных лопаточно-дисковых форм колебаний требует анализа колебаний полной трехмерной расчетной схемы узла с учетом ее взаимодействия с другими элементами конструкции.

Созданная в целях тестирования модель блиска, приведенная на рис. 4, топологически эквивалентна моделям реальных блисков и потому результаты тестирования алгоритмов распараллеливания с ее применением могут считаться верными и для других, более точных моделей.

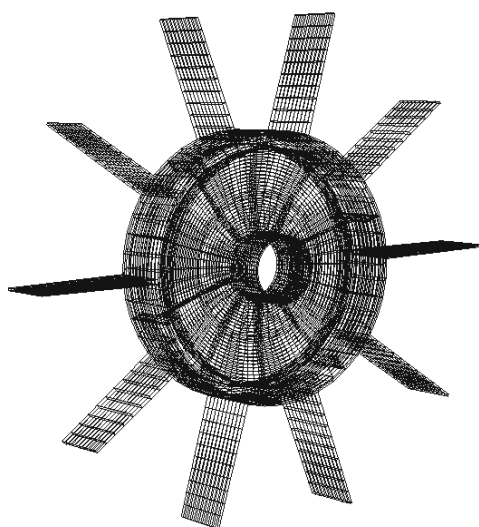


Рис. 4. Модель блиска

Реальные конструкции блисков содержат обычно более 20 лопаток, однако объемы данных, хранящихся в памяти, и время расчета таких моделей на одном компьютере очень велики (порядка суток), поэтому было решено ограничиться 10-ю лопатками и провести тестирование на 1-й, 2-х и 3-х достаточно мощных клиентских машинах с тем условием, чтобы все данные задачи находились в оперативной памяти компьютера. Это ограничение позволяет провести тесты в максимально корректной постановке.

Известно, что для блиска с 10 лопатками существуют семейства, состоящие из 10 форм колебаний лопаток, перемежаемые отдельно стоящими (зачастую парными) формами колебаний диска. Предварительные оценочные расчеты показали, что для рассматриваемой модели частоты располагаются следующим образом: 10 лопаточных (первая форма изгиба), затем 6 дисковых, затем снова 10 лопаточных (первая крутильная форма). Поэтому было решено провести расчет для 26 первых частот и форм колебаний блиска. При этом общее число итерационных векторов было взято равным 30.

Для оценки эффективности распараллеливания алгоритма поиска собственных частот и форм колебаний (СЧ и СФ) конструкция блиска была разрезана на 10 одинаковых секторов, каждый из которых сформировал свой суперэлемент. Матрица жесткости всей конструкции заняла 406 Мб, матрицы каждого дочернего СЭ — по 6 Мб, матрица главного СЭ — 42 Мб. Результаты тестирования приведены на рис. 5.

Несмотря на небольшие значения ускорения процесса решения распараллеливать задачу нахождения СЧ и СФ блисков необходимо.

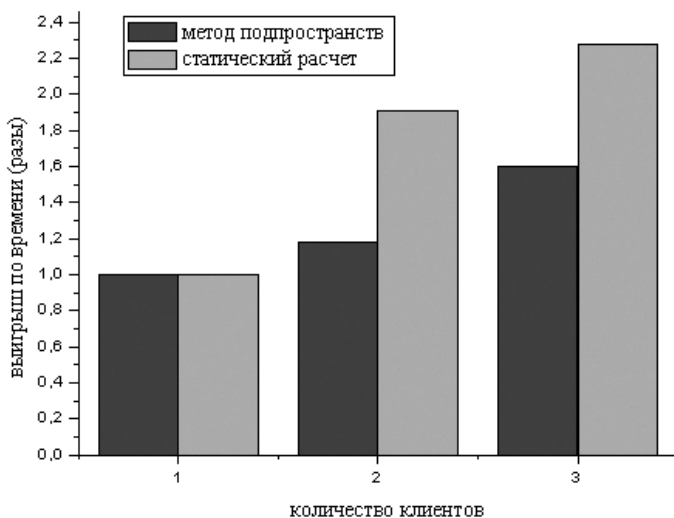


Рис. 5. Эффективность распараллеливания метода подпространств и решения задачи статики для расчетной схемы блиска с 10 лопатками

Реальные расчетные схемы на одном компьютере требуют постоянной работы с виртуальной памятью системы, что приводит к резкому (в десятки раз) увеличению времени расчета. Применение методики распараллеливания и продуманное разбиение конструкции на суперэлементы может и для таких случаев организовать вычисления без обращений к дисковой подсистеме компьютеров.

Выводы. Рассмотрены факторы, обуславливающие эффективность применения суперэлементов для расчетов прочности и теплопроводности деталей и узлов турбомашин, а также их оптимизации. Описан создаваемый на изложенных в статье принципах программный комплекс. Приведены результаты тестирования комплекса на типовых задачах проектирования узлов газотурбинных двигателей. Показана возможность существенного ускорения расчетов за счет распараллеливания на основе суперэлементного подхода.

Автор выражает благодарность московскому отделению компании Nvidia за любезно предоставленный для тестирования образец GPU Nvidia Tesla C2070.

Работа выполнена при поддержке гранта Президента РФ для государственной поддержки ведущих научных школ (проект НШ-255.2012. 8)

Статья поступила в редакцию 05.09.2012