УДК 539.3 DOI: 10.32326/1814-9146-2023-85-4-522-538

ОБЗОР ЛИТЕРАТУРЫ ПО РАСПАРАЛЛЕЛИВАНИЮ КОНЕЧНО-ЭЛЕМЕНТНЫХ СХЕМ РЕШЕНИЯ НЕЛИНЕЙНЫХ ЗАДАЧ ДЕФОРМИРОВАНИЯ КОНСТРУКЦИЙ НА СУПЕРЭВМ*

Кибец А.И., Калинина Ю.А.

Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского, Нижний Новгород, Российская Федерация

kibec@mech.unn.ru

Поступила в редакцию 27.06.2023

Приведен обзор публикаций, посвященных особенностям использования параллельных вычислительных систем для решения задач механики сплошных сред методом конечных элементов. В этих публикациях обсуждаются условия успешного применения алгоритмов и программного обеспечения, разрабатываемых для эффективного функционирования параллельных вычислительных систем. Изучаются способы ускорения решения задачи за счет учета ее специфики и приводятся оценки их эффективности. Рассмотрены вопросы применения: итерационных методов с предобусловливанием для решения разреженных систем линейных алгебраических уравнений с нерегулярной структурой как симметричных, так и несимметричных; конечно-элементных аппроксимаций на неструктурированных и квазиструктурированных расчетных сетках; конвейеризации вычислительного процесса при пространственной декомпозиции расчетной области; метода параллельных вычислений с явной и неявной схемами интегрирования по времени для крупномасштабных задач механики конструкций и твердых тел; конечно-элементного моделирования контактно-ударных задач с помощью графического процессора; безматричного подхода к конечно-элементным вычислениям, предназначенного для графических процессоров; явного метода асинхронных пошаговых параллельных вычислений для сокращения времени счета при уточнении сетки в некоторой локальной области; метода дискретных и конечных элементов и графического процессора для моделирования взаимодействия твердых частиц, включая обнаружение контакта, расчет силы и обновление информации; методов распараллеливания циклических участков последовательных программ, основанных на определении возможности параллельного выполнения итераций цикла, поиске и распараллеливании линейных участков внутри циклов с неизвестным количеством итераций.

Ключевые слова: нелинейный, нестационарный, деформирование, метод конечных элементов, параллельные вычислительные системы, суперЭВМ.

*Выполнено в рамках научной программы Национального центра физики и математики, направление №2 «Математическое моделирование на суперЭВМ экса- и зеттапроизводительности. Этап 2023–2025».

Введение

В программном документе «Эксафлопсные технологии. Концепция по развитию технологии высокопроизводительных вычислений на базе суперЭВМ эксафлопсного класса (2012–2020 гг.)» содержатся предложения по созданию отечественных эксафлопсных технологий. В документе выделены ближнесрочные и среднесрочные этапы работ по развитию суперкомпьютерных технологий и изложены актуальные задачи, соответствующие этапам, с указанием необходимых для их решения вычислительных мощностей. Концепция основывается на результатах работ, выполняемых госкорпорацией «Росатом» в широкой кооперации с предприятиями высокотехнологичных отраслей промышленности, ведущими научными школами и образовательными учреждениями по созданию суперкомпьютерных технологий предсказательного моделирования тера- и петафлопсного класса в интересах атомной энергетики, авиации, автомобилестроения и космоса, и предусматривает поэтапное создание на этой основе принципиально новых технологий моделирования многомасштабных процессов и взаимодействий эксафлопсного уровня сложности. При создании концепции использован опыт, накопленный в области фундаментальных исследований и других направлений работ. В концепции отмечается, что достижение эксафлопсной производительности требует разработки и применения технологий, реализующих принципиально различные дисциплины вычислений на качественно более высоких уровнях параллелизма, сложности и неоднородности вычислительных систем. В частности, требуются: а) фундаментальные исследования математических методов, алгоритмов и архитектур, позволяющих достигнуть необходимые значения производительности; б) создание и освоение прикладного программного обеспечения для имитационного моделирования на суперЭВМ эксафлопсного класса.

Анализ проблем и тенденций развития вычислительных систем максимального уровня производительности и возможные подходы к их решению, а также требования, предъявляемые к программному обеспечению для систем эксафлопсного уровня, достигающих рубежа скорости вычислений 10^{18} операций с плавающей запятой в секунду, представлен в публикациях [1–3]. В них отмечается, что условиями успешного применения алгоритмов и программного обеспечения, разрабатываемых для эффективного функционирования эксафлопсных систем, являются:

– значительная вычислительная трудоемкость решаемых задач;

– высокий запас параллелизма (масштабируемость) вычислений (вплоть до использования 10^9 процессоров);

– низкая интенсивность информационных взаимодействий (локальность) параллельно выполняемых вычислений;

– устойчивость вычислений к аппаратным сбоям вычислителей, которые неизбежно будут происходить при столь больших количествах вычислительных элементов.

Цель настоящей статьи – обзор публикаций, содержащих описание методов повышения эффективности конечно-элементных схем решения задач механики деформируемого твердого тела, ориентированных на суперЭВМ. При отборе материала предпочтение отдавалось исследованиям, опубликованным за предыдущие 15 лет. Ссылки на более ранние публикации, а также на публикации, не вошедшие в эту статью из-за ограниченности ее объема, можно найти в использованных здесь источниках.

Поскольку конечно-элементное решение статических и динамических задач имеет свои особенности, последующее изложение материала разделено соответственно на две части.

1. Применение метода конечных элементов для решения статических задач механики деформируемых сред и конструкций на суперЭВМ

В обзоре S. Georgescu, P. Chow, H. Okuda [4] представлено исследование внедрения графических процессоров (GPU) в качестве ускорителей в области анализа деформирования конструкций методом конечных элементов (МКЭ). Излагается работа, проделанная для ускорения наиболее трудоемких этапов конечно-элементного решения задачи, указывается достигнутое ускорение и, по возможности, описываются программные библиотеки.

В диссертации M.E. Guney [5] предлагается процедура решения линейной системы уравнений, которая лежит в основе программного обеспечения для анализа МКЭ и использует параллелизм, существующий в современных многоядерных процессорах с симметричной многопроцессорной обработкой (SMP). Разработано несколько алгоритмов для повышения производительности конечно-элементного решения задач. Описывается высокопроизводительный решатель, который позволяет экспериментировать с недавно разработанными и существующими алгоритмами. Приводится метод оценки количества операций для дальнейшего исследования их эффективности. Работоспособность предлагаемого решателя продемонстрирована на большом количестве тестовых задач.

Исследование S. Sanfui, D. Sharma [6] направлено на ускорение решения задачи за счет использования симметричного характера матрицы жесткости: вычисляется и собирается только нижняя треугольная часть локальной матрицы жесткости одного конечного элемента (КЭ) в глобальной матрице жесткости. Для обработки состояния гонки используется метод окраски. Установлено, что предложенная методика примерно в два раза быстрее стандартной реализации как для расчета одного КЭ, так и для этапа сборки глобальной матрицы жесткости. Кроме того, преимущество предлагаемого подхода заключается в меньшем объеме требуемой памяти, так как записывается только симметричная часть матрицы.

С.В. Поляков с соавторами [7] рассмотрели проблему разработки параллельных приложений для решения задач механики сплошной среды на вычислительных системах с гибридной архитектурой, включающей в себя центральные и графические процессоры. Для их решения сформулирована концепция гибридных параллельных вычислений, содержащая анализ особенностей гибридного вычислителя и его программного оснащения, а также предложения по реализации параллельных программ. Предложены три основные модели параллельных вычислений, использующие графические вычислители эпизодически, постоянно или в тесной связке с центральными процессорами.

В статье J.S. Mueller-Roemer, A. Stork [8] представлен метод сборки матриц жесткости для КЭ произвольного полиномиального порядка на симплексных сетках для графических процессоров. При применении метода не возникает необходимости в промежуточной разреженной матрице, и сборка выполняется непосредственно в окончательную структуру данных, оптимизированную для графического процессора. В результате сокращаются накладные расходы на память от 180 до 600% в зависимости от полиномиального порядка и связанные с этим затраты времени. При этом упрощается программный код и используется более компактное представление сетки. Выполнено сравнение предлагаемого метода с существующими алгоритмами, которое продемонстрировало значительное ускорение.

В статьях Д.А. Губайдуллина, А.И. Никифорова, Р.В. Садовникова [9, 10] исследуются особенности использования параллельных вычислительных систем для решения задач механики сплошных сред и, в частности, итерационных методов подпространств Крылова с предобусловливанием для решения разреженных систем линейных алгебраических уравнений (СЛАУ) с нерегулярной структурой как симметричных, так и несимметричных. Представлено расширение библиотеки GPU_SPARSE итерационных методов подпространств Крылова с предобусловливанием для использования на нескольких графических устройствах одновременно, а также на гибридных вычислительных системах, состоящих из нескольких вычислительных узлов, каждый из которых снабжен, как минимум, одним графическим устройством.

В статье S.Y. Fialko, F. Zeglen [11] рассматривается метод предобусловленных сопряженных градиентов (PCG), реализованный на GPU и предназначенный для решения больших задач строительной механики методом конечных элементов. Математическая постановка задачи приводит к решению систем линейных уравнений с разреженными симметричными положительно определенными матрицами. Авторы используют метод неполной факторизации Холецкого по значениям, основанный на технике разреженных матриц, для создания эффективного предобусловливания, обеспечивающего устойчивую сходимость для упомянутых выше слабо обусловленных задач.

Публикации С.П. Копысова с соавторами [12], C. Cecka, A.J. Lew, E. Darve [13, 14], J. Wong, E. Kuhl, E. Darve [15], T. Zegard, G.H. Paulino [16] посвящены конечно-элементным аппроксимациям на неструктурированных расчетных сетках. Такой подход приводит к нерегулярному доступу к сеточным данным, существенно уменьшающим эффективность параллельных вычислений. Узким местом поэлементных конечно-элементных схем является суммирование компонент из поэлементных векторов. При параллельном суммировании компонент, соответствующих общим узлам сетки, возникает так называемое состояние гонки, когда несколько параллельных вычислительных процессов обращаются к одной ячейке памяти. Эта операция суммирования включает в себя: чтение текущего значения компоненты, чтение компоненты из поэлементного вектора, сложение текущего значения с поэлементной компонентой и запись результата. Состояние гонки приводит к вычислительным ошибкам, которые могут быть связаны как с чтением текущего значения компоненты вектора, так и с записью результата. В статье С.П. Копысова с соавторами [12] предложены алгоритмы к разделению триангулированной многосвязной области на связные подобласти без ветвления внутренних границ. Приведены оценки сложности алгоритмов и проведено сравнение качества разделения триангулированных многосвязных областей.

C. Cecka, A.J. Lew, E. Darve в [13, 14] проанализировали несколько подходов к сборке и решению разреженных линейных систем и представили несколько стратегий эффективного использования глобальной, разделяемой и локальной памяти, методов объединения памяти и оптимального выбора параметров. Установлено, что оптимальная стратегия сборки зависит от порядка полиномов, используемых в дискретизации МКЭ.

J. Wong, E. Kuhl, E. Darve [15] предложили несколько вариантов нового алгоритма умножения разреженных матриц на вектор (sparse matrix vector multiplication SPMV) для неструктурированных сеток КЭ на графических процессорах. В этой статье анализируется эффективная пропускная способность существующих алгоритмов для графического процессора и предложенных алгоритмов для разреженных матриц разного размера и разной структуры разреженности. Тесты эффективной пропускной способности показывают, что новые алгоритмы могут дать заметные преимущества (коэффициенты ускорения до 12 раз для реальных приложений КЭ).

В статье T. Zegard, G.H. Paulino [16] исследуется возможность применения МКЭ и оптимизация топологии для неструктурированных сеток в массивно-параллельных компьютерных архитектурах, в частности, на графических процессорах. Обсуждаются проблемы в параллельной реализации, такие как условие гонки параллельной сборки.

И.А. Климонов, В.Д. Корнеев, В.М. Свешников провели исследование эффективности применения графических ускорителей при распараллеливании решения трехмерных краевых задач на квазиструктурированных сетках [17]. Основой распараллеливания решения трехмерных краевых задач в этом случае является метод декомпозиции расчетной области на параллелепипедальные подобласти, сопрягаемые без наложения. В каждой такой подобласти строится своя равномерная параллелепипедальная подсетка. Совокупность подсеток образует квазиструктурированную сетку. Подсетки группируются в объединения, содержащие приблизительно одинаковое количество узлов, для балансировки загрузки процессоров суперЭВМ.

Исследования U. Kiran, S.S. Gautam, D. Sharma [18], K. Ljungkvist [19, 20], M. Kronbichler et al. [21] посвящены развитию и применению безматричного подхода к конечно-элементным вычислениям. Безматричные решатели для МКЭ позволяют избежать сборки элементарных матриц и заменяют умножение разреженной матрицы на вектор произведением плотной матрицы на вектор на уровне элемента. Устранение необходимости хранить матрицу жесткости и требование малого объема памяти делают безматричные методы привлекательными.

В статье [18] предлагается новая безматричная стратегия для МКЭ, которая вычисляет матрично-векторное произведение на уровне элементов, используя только симметричную часть элементарных матриц. Вводится уникальная структура данных, которая обеспечивает локализованный и объединенный доступ к памяти, подходящий для графического процессора, сохраняя при этом только симметричную часть элементарных матриц. Предлагаемая стратегия делает упор на равномерное распределение рабочей нагрузки, снижение синхронизации потоков и поддержание достаточной степени детализации для наилучшего использования ресурсов графического процессора. Эффективность предложенной стратегии оценивается путем решения задач упругости и теплопроводности.

В [19, 20] исследуются различные методы решения проблемы конфликтующих обновлений. Рассматриваются безматричные реализации МКЭ высокого порядка для выполнения на графических процессорах. Показано, что на графическом процессоре безматричный подход является предпочтительным методом для элементов второго порядка и выше, обеспечивая как значительно более быстрое выполнение, так и решение значительно более крупных задач.

В статье [21] разработано распараллеливание на графическом процессоре многосеточного итеративного решателя без матриц, предназначенного для средних и высоких степеней полинома, с поддержкой общих криволинейных и адаптивно уточняемых шестигранных сеток. Показано, что предлагаемый подход позволяет избежать возможных условий гонки, связанной с общими вершинами, ребрами и поверхностями.

2. Применение метода конечных элементов для решения динамических задач механики деформируемых сред и конструкций на суперЭВМ

В статье T. Belytschko [22] представлено описание программы решения задач нелинейной динамики конструкций с библиотекой КЭ, которая эффективно использует параллельную обработку информации. В качестве побочного продукта была разработана автоматическая схема определения шагов интегрирования по времени определяющей системы уравнений. Доказана устойчивость алгоритма субциклирования.

Публикации [23–36] посвящены распараллеливанию и применению конечно-элементного решения задач динамики конструкций с явной конечно-разностной схемой интегрирования по времени.

В диссертации S.P. Banihashemi [23] исследуются существующие алгоритмы МКЭ с явной схемой интегрирования по времени, которые затем перерабатываются и реализуются для графических процессоров. Оценивается производительность этих алгоритмов и разрабатывается новый алгоритм пространственной декомпозиции с асинхронным интегрированием (AVISD), который может быть настроен в зависимости от рассматриваемой задачи и производительности компьютера.

В статье A. Bartezzağlhi et al. [24] рассматривается вопрос реализации на графическом процессоре с двойной точностью конечно-элементного решателя с явной схемой интегрирования по времени для задач динамики упругой оболочки при малых деформациях и больших смещениях и поворотах с использованием неструктурированных сеток. Показано, что за счет оптимизации алгоритма и тщательного управления памятью может быть достигнуто ускорение примерно в 5 раз.

Исследование T. Bahcecioğllu, Ö. Kurc [25] направлено на сокращение времени анализа крупномасштабных нелинейных динамических задач с использованием графических процессоров. В реализации использовалась явная версия семейства алгоритмов Ньюмарка. Этот тип алгоритма позволил применять вычисления к каждому КЭ, устраняя необходимость в глобальной сборке матриц. Его реализация осуществлена с использованием языка CUDA.

При использовании неявных конечно-разностных алгоритмов решение задачи сводится, в конечном итоге, к решению системы линейных алгебраических уравнений. При распараллеливании подобные алгоритмы приводят к весьма значительным обменам данными между отдельными узлами системы. Алгоритмы, построенные на основе явных конечно-разностных схем, допускают достаточно простое распараллеливание. Однако основным недостатком явных схем является их условная устойчивость, что снижает эффективность соответствующих алгоритмов. Улучшить описанную ситуацию позволяет переход на алгоритмы, основанные на полуявных конечно-разностных схемах. Полуявность в данном случае означает, что для расчета некоторой искомой величины в любой внутренней точке расчетной области используются значения этой величины в соседних узлах сетки не только на предыдущем временном слое, но и уже вычисленные значения этой величины на текущем.

В статье К.И. Михайленко, С.Ф. Хизбуллиной [26] исследована возможность параллельной реализации различных алгоритмов решения задач механики сплошной среды в зависимости от используемого вида конечно-разностных схем. Показано что алгоритмы, базирующиеся на полуявных численных схемах, которые объединяют некоторые достоинства как явных, так и неявных схем, могут быть эффективно распараллелены для использования на кластерных вычислительных системах. Предлагается техника конвейеризации вычислительного процесса при пространственной декомпозиции расчетной области для построения эффективных параллельных алгоритмов численного решения задач гидродинамики. Показаны методы достижения высокой эффективности параллельного приложения, основанного на конечно-разностной явной или полуявной численной схеме интегрирования по времени. Приводится соотношение, позволяющее оценить минимальные необходимые для эффективного вычислительного процесса размеры расчетной области.

В статье Y. Cai et al. [27] разработана конечно-элементная методика решения задач контактного взаимодействия с помощью графического процессора с использованием специального треугольного оболочечного элемента с гладкими краями (ES-FEM) и явной схемы интегрирования по времени. Особое внимание уделяется разработке подхода к параллельной реализации граничных условий. Предлагаются параллельный алгоритм поиска контактов и метод расчета штрафной функции.

K.T. Danielson, R.R. Namburu в [28] представили программную реализацию МКЭ с явной схемой интегрирования по времени для решения нелинейных нестационарных задач. Программный код написан на FORTRAN 90, для всех межпроцессорных взаимодействий используется библиотека MPI. Обсуждаются вопросы установки, переносимости и эффективности комбинации FORTRAN 90–MPI.

Для распараллеливания МКЭ на основе декомпозиции расчетной области и передачи сообщений может использоваться один из двух способов разделения, а именно разрез, проведенный через узлы и края или грани элементов (разрез узла), или разрез, проведенный через конечные элементы (разрез элемента). Стоимость последовательной версии конечно-элементного решения задачи динамики (без учета контакта) почти полностью связана с КЭ (оценкой напряженно-деформированного состояния). В прошлом использовалось исключительно разделение узлов. P. Krysl, Z. Bittnar показали [29] , что стратегия разбиения на элементы с разрезом масштабируется и представляет собой жизнеспособную альтернативу традиционному подходу с разрезом по узлам.

В статье B. Nutti, D. Marinković [30] приведен эффективный способ моделирования динамического поведения деформируемых конструкций с помощью МКЭ с применением графических процессоров. Продемонстрировано, как избежать потенциальных узких мест в алгоритме. Чтобы добиться правдоподобного описания деформации при больших локальных вращениях, используется формулировка МКЭ, учитывающая вращение КЭ как жесткого целого.

В статье A.R.M. Rao [31] представлены стратегии параллельных вычислений для реализации кода нелинейного анализа методом конечных элементов с явной схемой интегрирования по времени на компьютерах с распределенной памятью для решения крупномасштабных задач динамики конструкций. Подробно рассматриваются детали реализации как в гомогенных, так и в гетерогенных средах параллельной обработки. Сначала обсуждается реализация программного кода анализа нелинейных задач на однородных системах, а затем этот код переносится на гетерогенные системы. Отмечается, что декомпозиция расчетной области с явной передачей сообщений предпочтительна для параллельной реализации. Реализация передачи сообщений в параллельном алгоритме основана на библиотеках MPI (Message Passing Interface). Представлены аспекты реализации методов декомпозиции перекрывающихся и неперекрывающихся подобластей, динамического распределения задач (DTA) и методов кластеризации для DTA.

В статье V. Rek, I. Němec [32] изложена процедура параллельных вычислений с использованием метода динамической релаксации (DR) на графическом процессоре. Этот метод облегчает рассмотрение множества нелинейностей: статическая задача сводится к нестационарной задаче, для интегрирования которой по времени используется метод центральных разностей.

Основная цель диссертации V. Rek [33] – исследование потенциального использования параллелизма вычислений в области нелинейной динамики. Предлагаемая параллельная модель основана на МКЭ и явной конечно-разностной схеме интегрирования по времени, что, естественно, обеспечивает возможность эффективного распараллеливания. Разработанные подходы исследуются применительно к численному анализу контакта/удара оболочечных конструкций.

Для моделирования формовки листа Y. Cai et al. разработали КЭ с явной схемой интегрирования по времени, основанный на применении графического процессора [34]. Приводятся детали реализации предлагаемого метода.

В статье G. Anandakumar, J. Kim [35] рассматривается поведение функционально градиентных твердых тел при динамической ударной нагрузке в рамках линейной упругости с использованием параллельного алгоритма решения задачи с явной схемой интегрирования по времени. Представлены численные примеры, которые проверяют точность предлагаемого динамического кода КЭ и демонстрируют динамический отклик градиентных материалов. Исследован трехточечный изгиб балки из эпоксидной и стеклянной фаз при низкоскоростном ударе. Достоверность результатов расчетов доказана экспериментальными данными других исследователей.

В статье X. Xu et al. [36] излагается эффективная конечно-элементная модель оценки ударопрочности автомобильных покрытий, в которой объемные элементы и элементы оболочки соответственно используются для дискретизации области ударного контакта и остальной области покрытия. Проводятся три эталонных теста для проверки эффективности и экономичности разработанной вычислительной среды.

В публикациях Y. Cai et al. [37], J.E. Warner et al. [38], J. Allard, H. Courtecuisse, F. Faure [39] представлены реализации МКЭ с неявной схемой интегрирования по времени на графических процессорах.

В статье Y. Cai et al. [37] изложен эффективный метод параллельных вычислений с неявной схемой интегрирования по времени для крупномасштабных задач механики конструкций и твердых тел, основанный на графическом процессоре и вычислительной архитектуре унифицированных устройств (CUDA). Чтобы избежать сборки глобальной матрицы и уменьшить использование глобальной памяти графического процессора, система уравнений решается параллельно с помощью безматричной версии метода сопряженных градиентов. Представлены несколько стратегий эффективного использования памяти графического процессора. Численные примеры иллюстрируют масштабируемость и эффективность предлагаемого параллельного подхода.

Разработанная в NASA масштабируемая реализация МКЭ (Scalable Implementation of Finite Elements by NASA – ScIFEN) – это программный код конечно-элементного анализа, предназначенный для решений задач механики на суперЭВМ. Он поддерживает различные типы конечных элементов, нелинейные модели материалов и граничные условия. В отчете J.E. Warner et al. [38] приведено описание возможностей ScIFEN, включая обзор инструментов и функций, программного обеспечения, а также описание форматов входных и выходных файлов.

В статье J. Allard, H. Courtecuisse, F. Faure [39] отмечается, что существующие варианты МКЭ с явными схемами интегрирования по времени просты и легко распараллеливаются, однако имеют проблемы с устойчивостью и требуют очень малых временных шагов для моделирования жестких материалов. В [39] представлена реализация МКЭ с неявной схемой интегрирования по времени на графических процессорах. Чтобы распараллелить вычисления на GPU, необходимо добиться огромного уровня параллелизма. Во многих случаях вычисления независимы, за исключением того, что они должны распределять свои результаты по набору общих переменных. Такие операции могут быть записаны в виде графа, в котором узлы представляют общие переменные и ребра между ними.

С появлением передовых архитектур гетерогенных вычислений на основе ускорителей GPU разработчикам крупномасштабных производственных программ пришлось переосмыслить свои численные алгоритмы и включить новые модели программирования и стратегии управления памятью, чтобы эффективно использовать новейшие суперкомпьютеры. A. Vargas et al. в [40] обсуждают стратегию решения этих проблем, повышения производительности и достижения переносимости с помощью мультифизического программного кода нового поколения MARBL, разрабатываемого в Ливерморской национальной лаборатории им. Э. Лоуренса.

Конечно-элементный анализ сложных конструкций часто требует уточнения сетки в некоторой локальной области. Для сокращения времени вычислений Z. Ma et al. [41] предлагают метод асинхронных пошаговых параллельных вычислений, использующий декомпозицию с перекрывающимися узлами для разделения расчетной области на разные подобласти. Перекрывающиеся узлы между подобластями составляют область связи. Каждая подобласть выбирает временной шаг на основе характеристик сетки. Для согласования границы асинхронного шага используется метод подцикла. Модель подобласти, информация о границах и результаты расчетов хранятся в параллельных файлах, что означает, что весь процесс конечно-элементного анализа реализуется параллельно. Общее ускорение алгоритма связано с соотношением шагов между подобластями, количеством подобластей и балансом нагрузки.

Гибридный метод конечных и дискретных элементов (FEM–DEM) повышает эффективность моделирования взаимодействия конструкций и твердых частиц. Однако требуемые затраты вычислительных ресурсов ограничивают его применение к сложным инженерным задачам.

В статье Z. Zheng et al. [42] предлагается использовать графический процессор на основе FEM–DEM, включая обнаружение контакта, расчет силы и обновление информации. FEM–DEM на основе графического процессора применен для решения задачи взаимодействия пневматической шины с гранулированной средой. Результаты расчета показывают, что может быть достигнуто ускорение более чем в 15 раз. Результаты моделирования хорошо согласуются с результатами эксперимента в отношении общего тягового усилия и сопротивления движению.

Чтобы сократить время решения задач динамики, T. Li, Q. Wang, X. Jin [43] предложили для FEM–DEM эффективный алгоритм асинхронного пошагового интегрирования по времени. Декомпозиция делит расчетную область на подобласть FEM и подобласть DEM, в каждой из которых выбирается соответствующий временной шаг на основе характеристик элементов. На границах используются метод перекрытия узлов и частиц и метод подциклов для решения проблемы передачи и сопоставления граничных данных. На тестовых расчетах показано, что предлагаемый алгоритм обеспечивает необходимую точность расчета, существенно сокращает время компьютерного моделирования и повышает эффективность гибридного метода FEM–DEM.

Большую часть времени выполнения программ в многопроцессорных вычислительных системах занимают циклы. Их распараллеливание приводит к повышению быстродействия вычислительной системы и к более эффективному использованию ресурсов. В статье П.Ю. Ткачева с соавторами [44] представлены математическая модель и метод распараллеливания циклических участков последовательных программ, основанные на определении возможности параллельного выполнения итераций цикла, поиске и распараллеливании линейных участков внутри циклов с неизвестным количеством итераций, позволяющие сократить временные затраты на распараллеливание циклических участков последовательных программ за счет применения бинарных операций. Предложен алгоритм распараллеливания циклических участков последовательных программ, отличающийся использованием операций булевой алгебры в бинарных матрицах. Отмечается, что для повышения скоростных характеристик вычислительных систем недостаточно выполнять распараллеливание только на уровне команд, необходимо выявлять параллелизм и на уровне данных.

Заключение

Наличие публикаций, содержащих описание верифицированных методов усовершенствования конечно-элементных схем решения задач механики деформируемого твердого тела, свидетельствует об актуальности рассматриваемой проблемы, а представленные в них результаты демонстрирует значительный прогресс в этой области.

Среди рассмотренных методов повышения эффективности МКЭ можно отметить:

– внедрение графических процессоров (GPU) в качестве ускорителей при распараллеливании решения статических и динамических задач в области анализа деформирования конструкций методом конечных элементов;

– ускорение решения задачи за счет использования симметричного характера матрицы жесткости;

– использование итерационных методов с предобусловливанием для решения разреженных СЛАУ большой размерности (в частности, итерационных методов подпространств Крылова с предобусловливанием) при конечно-элементном исследовании статического деформирования конструкций;

– применение безматричного подхода к конечно-элементным вычислениям, который позволяет избежать сборки элементарных матриц и обеспечивает как значительно более быстрое выполнение, так и решение значительно более крупных задач;

– разработку алгоритма пространственной декомпозиции с асинхронным интегрированием по времени, который может быть настроен в зависимости от рассматриваемой задачи и производительности компьютера;

– применение техники конвейеризации вычислительного процесса при пространственной декомпозиции расчетной области для построения эффективных параллельных алгоритмов численного решения задач динамики;

– использование параллельных вычислений и метода динамической релаксации на графическом процессоре для анализа нелинейностей путем сведения статической задачи к нестационарной задаче, решение которой основано на методе центральных разностей;

– развитие гибридного метода конечных элементов – дискретных элементов (FEM–DEM) для повышения эффективности моделирования взаимодействия конструкций и твердых частиц.

Опубликованные в [4–43] результаты свидетельствуют об эффективности метода конечных элементов и позволяют рекомендовать его для численного исследования на суперЭВМ напряженно-деформированного состояния конструкций в физически и геометрически нелинейных задачах.

Список литературы

1. Воеводин В.В., Воеводин Вл.В. *Параллельные вычисления*. СПб.: БХВ-Петербург, 2002. 608 с.

2. Гергель В.П., Линёв А.В. Проблемы и перспективы достижения экзафлопного уровня производительности суперкомпьютерных систем. *Вестник Нижегородского университета им. Н.И. Лобачевского*. 2012. №3(1). С. 189–198.

3. Антонов А.С., Афанасьев И.В., Воеводин Вл.В. Высокопроизводительные вычислительные платформы: текущий статус и тенденции развития. *Вычислительные методы и программирование*. 2021. Т. 22. Вып. 2. С. 138–181. DOI: 10.26089/NumMet.v22r210.

4. Georgescu S., Chow P., Okuda H. GPU acceleration for FEM-based structural analysis. *Archives of Computational Methods in Engineering*. 2013. Vol. 20. No 2. P. 111–121. DOI: 10.1007/s11831-013-9082-8.

5. Guney M.E. High-performance direct solution of finite element problems on multi-core processors. *D. Sci.* (*School of Civil and Environmental Engineering*) *Dissertation*. Atlanta, Georgia, USA: Georgia Institute of Technology, 2010. 324 p.

6. Sanfui S., Sharma D. Exploiting symmetry in elemental computation and assembly stage of GPU-accelerated. *The 10th International Conference on Computational Methods* (*ICCM2019*). Singapore, 9–13 July 2019. Singapore. 2019. P. 641–651. https://www.iitg.ac.in/dsharma/papers/Conference/2019_symmetry_assembly.pdf.

7. Поляков С.В., Карамзин Ю.Н., Косолапов О.А., Кудряшова Т.А., Суков С.А. Гибридная суперкомпьютерная платформа и разработка приложений для решения задач механики сплошной среды сеточными методами. *Известия ЮФУ. Технические науки*. 2012. №6 (131). С. 105–115.

8. Mueller-Roemer J.S., Stork A. GPU-based polynomial finite element matrix assembly for simplex meshes. *Computer Graphics Forum*. 2018. Vol. 37. Iss. 7. P. 443–454. https://doi.org/10.1111/cgf.13581.

9. Губайдуллин Д.А., Никифоров А.И., Садовников Р.В. Библиотека GPU_SPARSE для численного решения задач механики сплошных сред на гибридной вычислительной системе. *Вестник Нижегородского университета им. Н.И. Лобачевского*. 2011. №2(1). С. 190–196.

10. Губайдуллин Д.А., Никифоров А.И., Садовников Р.В. Об особенностях использования архитектуры гетерогенного кластера для решения задач механики сплошных сред. *Вычислительные методы и программирование*. 2011. Т. 12. Вып. 4. С. 450–460.

11. Fialko S.Y., Zeglen F. Preconditioned conjugate gradient method for solution of large finite element problems on CPU and GPU. *Journal of Telecommunications and Information Technology*. 2016. No 2. P. 26–33.

12. Кадыров И.Р., Копысов С.П., Новиков А.К. Разделение триангулированной многосвязной области на подобласти без ветвления внутренних границ. *Ученые записки Казанского университета. Серия: Физико-математические науки*. 2018. Т. 160. Кн. 3. С. 544–560.

13. Cecka C., Lew A.J., Darve E. Assembly of finite element methods on graphics processors *International Journal for Numerical Methods in Engineering*. 2010. Vol. 85. No 5. P. 640–669. https://doi.org/10.1002/nme.2989.

14. Cecka C., Lew A., Darve E. Introduction to assembly of finite element methods on graphics processors. *IOP Conference Series: Materials Science and Engineering*. 2010. Vol. 10. Article No 012009. https://iopscience.iop.org/article/10.1088/1757-899X/10/1/012009/pdf.

15. Wong J., Kuhl E., Darve E. A new sparse matrix vector multiplication graphics processing unit algorithm designed for finite element problems. *International Journal for Numerical Methods in Engineering*. 2015. Vol. 102. Iss. 12. P. 1784–1814. https://doi.org/10.1002/nme.4865.

16. Zegard T., Paulino G.H. Toward GPU accelerated topology optimization on unstructured meshes. *Structural and Multidisciplinary Optimization*. 2013. Vol. 48. No 3. P. 473–485. https://doi.org/10.1007/s00158-013-0920-y.

17. Климонов И.А., Корнеев В.Д., Свешников В.М. Ускорение параллельных алгоритмов решения трехмерных краевых задач на квазиструктурированных сетках. *Вычислительные методы и программирование*. 2018. Т. 19. Вып. 2. С. 121–129. https://doi.org/10.26089/NumMet.v19r211.

18. Kiran U., Gautam S.S., Sharma D. GPU-based matrix-free finite element solver exploiting symmetry of elemental matrices. *Computing*. 2020. Vol. 102. No 9. P. 1941–1965. https://www.iitg.ac.in/dsharma/papers/journal/2020_EBE_SYM_FEA.pdf.

19. Ljungkvist K. Matrix-free finite-element operator application on graphics processing units. In: *European Conference on Parallel Processing*. Cham: Springer, 2014. P. 450–461. DOI: 10.1007/978-3-319-14313-2_38.

20. Ljungkvist K. Finite element computations on multicore and graphics processors. *D. Sci. Dissertation*. Uppsala, Sweden: Acta Universitatis Upsaliensis, 2017. 64 p. https://uu.diva-portal.org/smash/get/diva2:1088894/FULLTEXT01.pdf.

21. Kronbichler M., Ljungkvist K. Multigrid for matrix-free high-order finite element computations on graphics processors. *ACM Transactions on Parallel Computing* (*TOPC*). 2019. Vol. 6. No 1. P. 1–32. https://doi.org/10.1145/3322813.

22. Belytschko T. Parallel processors and nonlinear structural dynamics algorithms and software. *Final Technical Report* (*NASA-CR-179889*). Evanston, IL, USA: Northwestern university, 1986. 35 p.

23. Banihashemi S.P. Parallel explicit FEM algorithms using GPU's. *D. Sci. Dissertation.* Atlanta, Georgia, USA: Institute of Technology, 2015. 152 p.

24. Bartezzaghi A., Cremonesi M., Parolini N., Perego U. An explicit dynamics GPU structural solver for thin shell finite elements. *Computers & Structures.* 2015. Vol. 154. P. 29–40. https://doi.org/10.1016/j.compstruc.2015.03.005.

25. Bahcecioğlu T., Kurc Ö. Nonlinear dynamic finite element analysis with GPU. *Conference ICCCBE 14.* 2012. 9 p.

26. Михайленко К.И., Хизбуллина С.Ф. Об одном эффективном конвейерном параллельном алгоритме для решения задач механики сплошной среды. *Труды Института механики им. Р.Р. Мавлютова УНЦ РАН*. 2006. Т. 4. С. 90–102. DOI: 10.21662/uim2006.1.009.

27. Cai Y., Cui X-Y., Li G., Liu W. A parallel finite element procedure for contact-impact problems using edge-based smooth triangular element and GPU. *Computer Physics Communications*. 2018. Vol. 225. P. 47–58. DOI:10.1016/j.cpc.2017.12.006.

28. Danielson K.T., Namburu R.R. Nonlinear dynamic finite element analysis on parallel computers using FORTRAN 90 and MPI. *Advances in Engineering Software*. 1998. Vol. 29. Iss. 3–6. P. 179–186. https://doi.org/10.1016/S0965-9978(98)00019-2.

29. Krysl P., Bittnar Z. Parallel explicit finite element solid dynamics with domain decomposition and message passing: dual partitioning scalability. *Computers & Structures*. 2001. Vol. 79. Iss. 3. P. 345–360. DOI: 10.1016/S0045-7949(00)00130-9.

30. Nutti B., Marinković D. An approach to efficient FEM simulations on graphics processing units using CUDA. *Facta Universitatis. Series: Mechanical Engineering.* 2014. Vol. 12. No 1. P. 15–25.

31. Rao A.R.M. Explicit nonlinear dynamic finite element analysis on homogeneous/heterogeneous parallel computing environment. *Advances in Engineering Software.* 2006. Vol. 37. Iss. 11. P. 701–720. https://doi.org/10.1016/j.advengsoft.2006.04.003.

32. Rek V., Němec I. Parallel computing procedure for dynamic relaxation method on GPU using NVIDIA's CUDA. *Applied Mechanics and Materials*. 2016. Vol. 821. P. 331–337. https://doi.org/10.4028/www.scientific.net/AMM.821.331.

33. Rek V. The exploitation of parallelization to numerical solutions regarding problems in nonlinear dynamicsk. *D. Sci. Dissertation*. Brno, Česko: Brno University of Technology, 2018. 51 p. https://dspace.vutbr.cz/bitstream/handle/11012/137861/thesis-1.pdf?sequence=21.

34. Cai Y., Li G., Wang H., Zheng G., Lin S. Development of parallel explicit finite element sheet forming simulation system based on GPU architecture. *Advances in Engineering Software*. 2012. Vol. 45. Iss. 1. P. 370–379. https://doi.org/10.1016/j.advengsoft.2011.10.014.

35. Anandakumar G., Kim J. Three-dimensional finite element analysis for nonhomogeneous materials using parallel explicit algorithm. In: *Computational Models in Engineering*. Ed. K. Volkov. 2018. P. 17. DOI: 10.5772/intechopen.82410.

36. Xu X., Zou C., Zang M., Chen S. Development of a GPU parallel computational framework for impact debonding of coating-substrate interfaces. *Thin-Walled Structures*. 2022. Vol. 175. Article No 109270. https://doi.org/10.1016/j.tws.2022.109270.

37. Cai Y., Li G., Wang H. A parallel node-based solution scheme for implicit finite element method using GPU. *Procedia Engineering*. 2013. Vol. 61. P. 318–324. https://doi.org/10.1016/j.proeng.2013.08.022.

38. Warner J.E., Bomarito G.F., Heber G., Hochhalter J.D. Scalable implementation of finite elements by NASA _ Implicit (ScIFEi). *Report № NF1676 L-23876. NASA/TM-2016-219180*. 2016. 15 p.

39. Allard J., Courtecuisse H., Faure F. Implicit FEM solver on GPU for interactive deformation simulation. *GPU Computing Gems Jade Edition*. 2012. P. 281–294. https://doi.org/10.1016/B978-0-12-385963-1.00021-6.

40. Vargas A., Stitt T.M., Weiss K., Tomov V.Z., Camier J.S., Kolev T., Rieben R.N. Matrix-free approaches for GPU acceleration of a high-order finite element hydrodynamics application using MFEM, Umpire, and RAJA. *The International Journal of High Performance Computing Applications*. 2022. No 36. Iss. 44. P. 492–509. DOI: 10.1177/10943420221100262.

41. Ma Z., Lou Y., Li J., Jin X. An explicit asynchronous step parallel computing method for finite element analysis on multi-core clusters. *Engineering with Computers*. 2020. Vol. 36. No 2. P. 443–453. DOI: 10.1007/s00366-019-00704-5.

42. Zheng Z., Zang M., Chen S., Zeng H. A GPU-based DEM-FEM computational framework for tire-sand interaction simulations. *Computers & Structures*. 2018. Vol. 209. P. 74–92. https://doi.org/10.1016/j.compstruc.2018.08.011.

43. Li T., Wang Q., Jin X. A staggered asynchronous step integration algorithm for hybrid finite-element and discrete-element modeling. *International Journal of Computational Methods*. 2022. Vol. 19. No 2. Article No 2150064. https://doi.org/10.1142/S021987622150064X.

44. Ткачев П.Ю., Борзов Д.Б., Чернецкая И.Е. Метод и алгоритм поиска линейных участков внутри циклов с последующим распараллеливанием. *Известия Юго-Западного государственного университета*. 2015. №5 (62). С. 16–20.

References

1. Voevodin V.V., Voevodin Vl.V. *Parallelnye vychisleniya* [*Parallel Computing*]. Saint Petersburg. BKhV-Petesrburg Publ. 2002. 608 p. (In Russian).

2. Gergel V.P., Linev A.V. Problemy i perspektivy dostizheniya ekzaflopnogo urovnya proizvoditelnosti superkompyuternykh sistem [Exaflop performance of supercomputers: challenges and trends]. *Vestnik Nizhegorodskogo universiteta imeni N.I. Lobachevskogo* [*Vestnik of Lobachevsky University of Nizhny Novgorod*]. 2012. No 3(1). P. 189–198 (In Russian).

3. Antonov A.S., Afanasiev I.V., Voevodin Vl.V. Vysokoproizvoditelnye vychislitelnye platformy: tekushchiy status i tendentsii razvitiya [High-performance computing platforms:current status and development trends]. *Vychislitelnye metody i programmirovanie* [*Numerical Methods and Programming*]. 2021. Vol. 22. Iss. 2. P. 138–181 (In Russian).

4. Georgescu S., Chow P., Okuda H. GPU acceleration for FEM-based structural analysis. *Arch. Comput. Methods Eng*. 2013. Vol. 20. No 2. P. 111–121. DOI: 10.1007/s11831-013-9082-8.

5. Guney M.E. High-performance direct solution of finite element problems on multi-core processors. *D. Sci.* (*School of Civil and Environmental Engineering*) *Dissertation*. Atlanta, Georgia, USA. Georgia Institute of Technology. 2010. 324 p.

6. Sanfui S., Sharma D. Exploiting symmetry in elemental computation and assembly stage of GPU-accelerated. *The 10th International Conference on Computational Methods* (*ICCM2019*). Singapore, 9–13 July 2019. Singapore. 2019. P. 641–651. https://www.iitg.ac.in/dsharma/papers/Conference/2019_symmetry_assembly.pdf

7. Polyakov S.V., Karamzin Yu.N., Kosolapov O.A., Kudryashova T.A., Soukov S.A. Gibridnaya superkompyuternaya platforma i razrabotka prilozheniy dlya resheniya zadach mekhaniki sploshnoy sredy setochnymi metodami [Hibrid supercomputer platform and applications programming for the solution of continuous mechanics problems by grid methods]. *Izvestiya Yuzhnogo federalnogo universiteta. Tekhnicheskie nauki* [*Izvestiya SFedU. Engineering Sciences*]. 2012. No 6 (131). P. 105–115 (In Russian).

8. Mueller-Roemer J.S., Stork A. GPU-based polynomial finite element matrix assembly for simplex meshes. *Comput. Graph. Forum*. 2018. Vol. 37. Iss. 7. P. 443–454. https://doi.org/10.1111/cgf.13581.

9. Gubaydullin D.A., Nikiforov A.I., Sadovnikov R.V. Biblioteka GPU_SPARSE dlya chislennogo resheniya zadach mekhaniki sploshnykh sred na gibridnoy vychislitelnoy sisteme [GPU_SPARSE library for the numerical solution of continuum mechanics problems on a hybrid computing system]. *Vestnik Nizhegorodskogo universiteta imeni N.I. Lobachevskogo* [*Vestnik of Lobachevsky University of Nizhny Novgorod*]. 2011. No 2(1). P. 190–196 (In Russian).

10. Gubaydullin D.A., Nikiforov A.I., Sadovnikov R.V. Ob osobennostyakh ispolzovaniya arkhitektury geterogennogo klastera dlya resheniya zadach mekhaniki sploshnykh sred [On peculiarities of using heterogeneous cluster architecture for solving continuum mechanics problems]. *Vychislitelnye metody i programmirovanie* [*Numerical Methods and Programming*]. 2011. Vol. 12. Iss. 4. P. 450–460 (In Russian).

11. Fialko S.Y., Zeglen F. Preconditioned conjugate gradient method for solution of large finite element problems on CPU and GPU. *J. Telecommun. Inf. Technol*. 2016. No 2. P. 26–33.

12. Kadyrov I.R., Kopysov S.P., Novikov A.K. Razdelenie triangulirovannoy mnogosvyaznoy oblasti na podoblasti bez vetvleniya vnutrennikh granits [Partitioning of triangulated multiply connected domain into subdomains without branching of inner boundaries]. *Uchenye zapiski Kazanskogo universiteta. Seriya: Fiziko-matematicheskie nauki* [*Proceedings of Kazan University. Physics and Mathematics Series*]. 2018. Vol. 160. Book 3. P. 544–560 (In Russian).

13. Cecka C., Lew A.J., Darve E. Assembly of finite element methods on graphics processors *International Journal for Numerical Methods in Engineering*. 2010. Vol. 85. No 5. P. 640–669. https://doi.org/10.1002/nme.2989.

14. Cecka C., Lew A., Darve E. Introduction to assembly of finite element methods on graphics processors. *IOP Conference Series: Materials Science and Engineering*. 2010. Vol. 10. Article No 012009. https://iopscience.iop.org/article/10.1088/1757-899X/10/1/012009/pdf.

15. Wong J., Kuhl E., Darve E. A new sparse matrix vector multiplication graphics processing unit algorithm designed for finite element problems. *International Journal for Numerical Methods in Engineering*. 2015. Vol. 102. Iss. 12. P. 1784–1814. https://doi.org/10.1002/nme.4865.

16. Zegard T., Paulino G.H. Toward GPU accelerated topology optimization on unstructured meshes. *Structural and Multidisciplinary Optimization*. 2013. Vol. 48. No 3. P. 473–485. https://doi.org/10.1007/s00158-013-0920-y.

17. Klimonov I.A., Korneev V.D., Sveshnikov V.M. Uskorenie parallelnykh algoritmov resheniya trekhmernykh kraevykh zadach na kvazistrukturirovannykh setkakh [Acceleration of parallel algorithms for solving three-dimensional boundaryvalue problems on quasi-structured grids]. *Vychislitelnye metody i programmirovanie* [*Numerical Methods and Programming*]. 2018. Vol. 19. Iss. 2. P. 121–129. https://doi.org/10.26089/NumMet.v19r211 (In Russian).

18. Kiran U., Gautam S.S., Sharma D. GPU-based matrix-free finite element solver exploiting symmetry of elemental matrices. *Computing*. 2020. Vol. 102. No 9. P. 1941–1965. https://www.iitg.ac.in/dsharma/papers/journal/2020_EBE_SYM_FEA.pdf.

19. Ljungkvist K. Matrix-free finite-element operator application on graphics processing units. In: *European Conference on Parallel Processing*. Cham. Springer. 2014. P. 450–461. DOI: 10.1007/978-3-319-14313-2_38.

20. Ljungkvist K. Finite element computations on multicore and graphics processors. *D. Sci. Dissertation.* Uppsala, Sweden. Acta Universitatis Upsaliensis. 2017. 64 p. https://uu.diva-portal.org/smash/get/diva2:1088894/FULLTEXT01.pdf.

21. Kronbichler M., Ljungkvist K. Multigrid for matrix-free high-order finite element computations on graphics processors. *ACM Transactions on Parallel Computing* (*TOPC*). 2019. Vol. 6. No 1. P. 1–32. https://doi.org/10.1145/3322813.

22. Belytschko T. Parallel processors and nonlinear structural dynamics algorithms and software. *Final Technical Report* (*NASA-CR-179889*). Evanston, IL, USA. Northwestern university. 1986. 35 p.

23. Banihashemi S.P. Parallel explicit FEM algorithms using GPU's. *D. Sci. Dissertation.* Atlanta, Georgia, USA. Institute of Technology. 2015. 152 p.

24. Bartezzaghi A., Cremonesi M., Parolini N., Perego U. An explicit dynamics GPU structural solver for thin shell finite elements. *Comp. Struct*. 2015. Vol. 154. P. 29–40. https://doi.org/10.1016/j.compstruc.2015.03.005.

25. Bahcecioğlu T., Kurc Ö. Nonlinear dynamic finite element analysis with GPU. *Conference ICCCBE 14.* 2012. 9 p.

26. Mikhaylenko K.I., Khizbullina S.F. Ob odnom effektivnom konveyernom parallelnom algoritme dlya resheniya zadach mekhaniki sploshnoy sredy [About one efficient pipeline parallel algorithm for solving problems of continuum mechanics]. *Trudy Instituta mekhaniki im. R.R. Mavlyutova Ufimskogo nauchnogo tsentra RAN* [*Proceedings of the Mavlyutov Institute of Mechanics*]. 2006. Vol. 4. P. 90–102 (In Russian).

27. Cai Y., Cui X-Y., Li G., Liu W. A parallel finite element procedure for contact-impact problems using edge-based smooth triangular element and GPU. *Comput. Phys. Commun*. 2018. Vol. 225. P. 47–58. DOI: 10.1016/j.cpc.2017.12.006.

28. Danielson K.T., Namburu R.R. Nonlinear dynamic finite element analysis on parallel computers using FORTRAN 90 and MPI. *Adv. Eng. Softw*. 1998. Vol. 29. Iss. 3–6. P. 179–186. https://doi.org/10.1016/S0965-9978(98)00019-2.

29. Krysl P., Bittnar Z. Parallel explicit finite element solid dynamics with domain decomposition and message passing: dual partitioning scalability. *Comput. Struct*. 2001. Vol. 79. Iss. 3. P. 345–360. DOI: 10.1016/S0045-7949(00)00130-9.

30. Nutti B., Marinković D. An approach to efficient FEM simulations on graphics processing units using CUDA. *Facta Universitatis. Series: Mechanical Engineering*. 2014. Vol. 12. No 1. P. 15–25.

31. Rao A.R.M. Explicit nonlinear dynamic finite element analysis on homogeneous/heterogeneous parallel computing environment. *Adv. Eng. Softw*. 2006. Vol. 37. Iss. 11. P. 701–720. https://doi.org/10.1016/j.advengsoft.2006.04.003.

32. Rek V., Němec I. Parallel computing procedure for dynamic relaxation method on GPU using NVIDIA's CUDA. *Appl. Mech. Mater*. 2016. Vol. 821. P. 331–337. https://doi.org/10.4028/www.scientific.net/AMM.821.331.

33. Rek V. The exploitation of parallelization to numerical solutions regarding problems in nonlinear dynamicsk. *D. Sci. Dissertation*. Brno, Česko. Brno University of Technology. 2018. 51 p. https://dspace.vutbr.cz/bitstream/handle/11012/137861/thesis-1.pdf?sequence=21.

34. Cai Y., Li G., Wang H., Zheng G., Lin S. Development of parallel explicit finite element sheet forming simulation system based on GPU architecture. *Adv. Eng. Softw*. 2012. Vol. 45. Iss. 1. P. 370–379. https://doi.org/10.1016/j.advengsoft.2011.10.014.

35. Anandakumar G., Kim J. Three-dimensional finite element analysis for nonhomogeneous materials using parallel explicit algorithm. In: *Computational Models in Engineering*. Ed. K. Volkov. 2018. P. 17. DOI: 10.5772/intechopen.82410.

36. Xu X., Zou C., Zang M., Chen S. Development of a GPU parallel computational framework for impact debonding of coating-substrate interfaces. *Thin Wall. Struct.* 2022. Vol. 175. Article No 109270. https://doi.org/10.1016/j.tws.2022.109270.

37. Cai Y., Li G., Wang H. A parallel node-based solution scheme for implicit finite element method using GPU. *Procedia Engineering*. 2013. Vol. 61. P. 318–324. https://doi.org/10.1016/j.proeng.2013.08.022.

38. Warner J.E., Bomarito G.F., Heber G., Hochhalter J.D. Scalable implementation of finite elements by NASA _ Implicit (ScIFEi). *Report № NF1676 L-23876. NASA/TM-2016-21918*0. 2016. 15 p.

39. Allard J., Courtecuisse H., Faure F. Implicit FEM solver on GPU for interactive deformation simulation. *GPU Computing Gems Jade Edition*. 2012. P. 281–294. https://doi.org/10.1016/B978-0-12-385963-1.00021-6.

40. Vargas A., Stitt T.M., Weiss K., Tomov V.Z., Camier J.S., Kolev T., Rieben R.N. Matrix-free approaches for GPU acceleration of a high-order finite element hydrodynamics application using MFEM, Umpire, and RAJA. *The International Journal of High Performance Computing Applications*. 2022. No 36. Iss. 44. P. 492–509. DOI: 10.1177/10943420221100262.

41. Ma Z., Lou Y., Li J., Jin X. An explicit asynchronous step parallel computing method for finite element analysis on multi-core clusters. *Eng. Comput.* 2020. Vol. 36. No 2. P. 443–453. DOI: 10.1007/s00366-019-00704-5.

42. Zheng Z., Zang M., Chen S., Zeng H. A GPU-based DEM-FEM computational framework for tire-sand interaction simulations. *Comput. Struct*. 2018. Vol. 209. P. 74–92. https://doi.org/10.1016/j.compstruc.2018.08.011.

43. Li T., Wang Q., Jin X. A staggered asynchronous step integration algorithm for hybrid finite-element and discrete-element modeling. *Int. J. Comput. Methods*. 2022. Vol. 19. No 2. Article No 2150064. https://doi.org/10.1142/S021987622150064X.

44. Tkachev P.Yu., Borzov D.B., Chernetckaja I.E. Metod i algoritm poiska lineynykh uchastkov vnutri tsiklov s posleduyushchim rasparallelivaniem [Method and algorythm of searching linear sections in the cycles with next step parallelization]. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta* [*Proceedings of the Southwest State University*]. 2015. No 5 (62). P. 16–20 (In Russian).

REVIEW OF LITERATURE ON PARALLELING FINITE ELEMENT SCHEMES FOR SOLVING NONLINEAR STRUCTURE DEFORMATION PROBLEMS ON SUPER COMPUTERS*

Kibets A.I., Kalinina Yu.A.

National Research Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod, Russian Federation

kibec@mech.unn.ru

Received by the Editor 2023/06/27

This article provides a review of publications devoted to the features of using parallel computing systems for solving problems in continuum mechanics using the finite element method. These publications discuss the conditions for the successful application of algorithms and software developed for the efficient operation of parallel computing systems. Methods for accelerating the solution of a problem by taking into account its specifics are studied and estimates of their effectiveness are given. The issues of application are considered: iterative methods with preconditioning for solving sparse systems of linear algebraic equations with an irregular structure, both symmetric and asymmetric; finite element approximations on unstructured and quasi-structured computational grids; pipelineization of the computational process during spatial decomposition of the computational domain; a parallel computing method with an explicit and implicit time integration scheme for large-scale problems in the mechanics of structures and solids; finite element modeling of contact-impact problems using a graphics processor; matrix-free approach to finite element calculations intended for graphic processors; an explicit method of asynchronous step-by-step

* This study was conducted within the scientific program of the National Center for Physics and Mathematics, stage #2 “Mathematical Modeling on Exa-scale and Zetta-scale supercomputers. Stage 2023–2025”.

parallel calculations to reduce computation time when refining the mesh in a certain local area; discrete and finite element method and graphic processor for solid particle interaction modeling, including contact detection, force calculation and information updating; methods for parallelizing cyclic sections of sequential programs, based on determining the possibility of parallel execution of loop iterations, searching and parallelizing linear sections within loops with an unknown number of iterations.

Keywords: nonlinear, non-stationary, deformation, finite element method, parallel computing systems, supercomputer.