

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ИССЛЕДОВАНИЕ СЛОЖНЫХ УПРАВЛЯЕМЫХ СИСТЕМ

УДК 519.6:539.3

А.Н. Химич, В.А. Декрет, А.В. Попов, А.В. Чистяков

ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ УСТОЙЧИВОСТИ КОМПОЗИТНЫХ МАТЕРИАЛОВ НА КОМПЬЮТЕРАХ ГИБРИДНОЙ АРХИТЕКТУРЫ

Введение

Компьютерное моделирование и численный эксперимент в наше время используются в различных сферах человеческой деятельности. В каждой отрасли народного хозяйства возникают задачи, требующие обработки огромных объемов информации с высокими быстродействием и точностью. Поэтому для их исследования применяются все более мощные вычислительные системы. На сегодня такими компьютерами являются суперкомпьютеры различных параллельных архитектур. Среди них особенно перспективны компьютеры гибридной архитектуры — многоядерные компьютеры MIMD-архитектуры с графическими ускорителями SIMD-архитектуры [1]. В этих компьютерах распараллеливание вычислений на центральных процессорах (CPU, Central Processing Unit) сочетается с вычислениями на графических процессорах (GPU, Graphical Processing Unit), что значительно сокращает время решения задачи. Однако достичь высокой эффективности при решении прикладных задач на таких компьютерах можно лишь, используя алгоритмы, которые учитывают архитектурные и технологические особенности.

MIMD-архитектура, как правило, предполагает наличие вычислительных узлов, каждый из которых может иметь несколько многоядерных процессоров и локальную память, т.е. каждый процессор имеет доступ только к локальной памяти своего узла. Вычислительные узлы объединяются некоторой коммуникационной средой, поэтому доступ к данным, которые размещены в памяти других узлов, выполняется дольше. С помощью системы MPI (Message Passing Interface) вычисления распараллеливаются между ядрами на CPU в виде отдельных процессов [2]. Для эффективного выполнения параллельных вычислений нужна равномерная нагрузка в каждый момент времени всех процессоров, участвующих в решении задачи, в результате чего минимизируется время пребывания процессоров в состоянии ожидания. Кроме того, каждый процессор CPU имеет свою кеш-память, при использовании которой можно значительно улучшить время выполнения задачи.

Графический процессор состоит из нескольких независимых мультипроцессоров, каждый из которых имеет несколько ядер CUDA (Compute Unified Device Architecture). Технология CUDA — программно-аппаратная вычислительная система, позволяющая эффективно распараллеливать однотипные математические операции с большими объемами данных на огромное количество потоков [3]. Это дает возможность значительно ускорить выполнение задачи. Однако CPU и GPU

© А.Н. ХИМИЧ, В.А. ДЕКРЕТ, А.В. ПОПОВ, А.В. ЧИСТЯКОВ, 2018

*Международный научно-технический журнал
«Проблемы управления и информатики», 2018, № 4*

имеют различную память. CUDA предоставляет доступ на GPU к шести видам памяти, каждая из которых имеет свое назначение. При этом связь между CPU и GPU (пересылка информации) осуществляется только через глобальную память (global) графического процессора, которая очень медленная, а вычисления выполняются на разделенной (shared) быстрой памяти, объем которой значительно меньше объема кеш-памяти CPU. Это нужно учитывать как при распределении данных между процессорами, так и при организации вычислений.

Математическое моделирование процессов, связанных с исследованиями устойчивости конструкций, применяется в различных предметных областях. При эксплуатационных нагрузках и технологических процессах изготовления волокнистых и слоистых композитных материалов возникают сжимающие напряжения, которые могут привести к потере устойчивости в структуре композитного материала. Численное решение таких задач сводится к частичной обобщенной алгебраической проблеме собственных значений (АПСЗ) с ленточными положительно-определенными матрицами.

В работе рассматривается численное исследование задачи устойчивости композитных материалов на многоядерном компьютере с графическими процессорами с помощью разработанного параллельного алгоритма метода итераций на подпространстве.

Моделирование задачи устойчивости с применением модели «волокон конечных размеров»

Рассматривается задача устойчивости композитных материалов бесконечно длинных полос (бесконечных в направлении перпендикуляра к рассматриваемой плоскости) конечной ширины, т.е. к слоистым композитам с наполнителем ленточной структуры [4–7].

Композитом называется неоднородный материал, состоящий из двух или более компонентов (фаз). Например, это — различные дисперсионно-упрочненные сплавы, бетоны, металлы, поликристаллические тела и т.д. В механике, ради удобства, одну из фаз называют матрицей (или связующим элементом), а другие — включениями (или армирующими элементами). Композиционные материалы обычно классифицируются по форме включений. Самыми распространенными среди них являются макрочастицы (гранулы), короткие (или разорванные) волокна, непрерывные длинные волокна (нити), а также слои. Матрица обеспечивает монолитность композита, фиксирует форму изделия и взаимное расположение армирующих волокон, распределяет действующие напряжения по объему материала, обеспечивая равномерную нагрузку на волокно и ее перераспределение при разрушении части волокон.

Результаты исследований по теории устойчивости во внутренней структуре (внутренняя неустойчивость) композитов при сжатии вдоль армирующих элементов получены в рамках модели «волокон конечных размеров» (ВКР) с использованием основных соотношений трехмерной линеаризованной теории устойчивости деформируемых тел (ТЛТУДТ) [4–7]. В силу применения модели ВКР в таких задачах возникает неоднородное докритическое состояние, поэтому исследование задач с применением модели ВКР возможно исключительно с применением численных методов.

В соответствии с основными положениями [6], которые применяются для постановки задач ТЛТУДТ композитных материалов в рамках модели ВКР, волокнистый однонаправленный композит в задачах устойчивости представляется двухмерной моделью (плоская деформация) для слоистого композита. Для результатов, полученных при указанном моделировании, применяется терминология, соответствующая теории устойчивости волокнистого однонаправленного композита (обсуждаемые моделирование и терминология применяются только при интерпретации полученных результатов). А результаты определены с использованием соотношений ТЛТУДТ [4, 5] и модели кусочно-однородной среды применительно к композитам, исследование устойчивости которых во внутренней структуре осуществляется в рамках модели ВКР [6, 7].

Постановка задачи

Рассматривается общая постановка задачи определения докритического состояния и задачи устойчивости композитного материала для модели ВКР. Исследование выполняется в рамках плоской деформации в плоскости x_1Ox_2 с применением основных соотношений ТЛТУДТ для малых докритических деформаций, а также модели упругого изотропного тела для армирующих элементов и матрицы (рис. 1).

Применяются лагранжевы координаты, которые в недеформированном состоянии совпадают с декартовыми координатами. На рис. 1 приведена расчетная схема для простейшего случая, когда волокна не взаимодействуют через матрицу между собой как в докритическом состоянии, так и при потере устойчивости. Такая ситуация возникает в случае композитных материалов с достаточно малой концентрацией наполнителя.

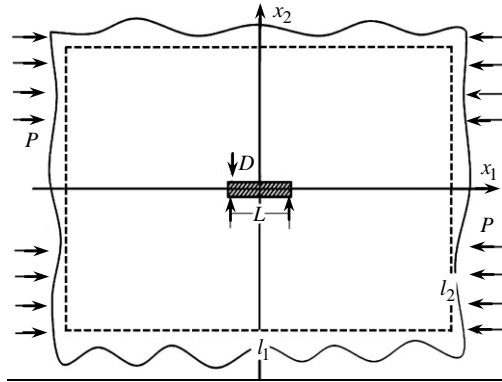


Рис. 1

Рассматривается продольное сечение волокнистых материалов, которое проходит через ось волокон, лежащих в одной плоскости. Таким образом, фактически проводится замена пространственной задачи (для цилиндрических ВКР) плоской (для слоев конечных размеров). Исследование докритического состояния выполняется в рамках классической линейной теории упругости изотропного тела, уравнения равновесия и соотношения упругости которой можно представить в следующем виде:

$$\frac{\partial}{\partial x_i} \sigma_{ij}^0 = 0; \quad \sigma_{ij}^0 = \delta_{ij} \lambda \varepsilon_{nn}^0 + 2\mu \varepsilon_{ij}^0; \quad 2\varepsilon_{ij}^0 = \frac{\partial u_i^0}{\partial x_j} + \frac{\partial u_j^0}{\partial x_i}. \quad (1)$$

Здесь и далее индексом «0» обозначаются величины, относящиеся к начальному (докритическому) состоянию, индексами «a» и «m» — величины, относящиеся к материалу волокна и матрицы соответственно.

Поскольку для матрицы исследование проводится для бесконечной области, напряжения и перемещения в матрице удобно представить в виде суммы компонентов напряжений и перемещений, соответствующих внешней нагрузке, заданной для матрицы «на бесконечности», и компонентов, соответствующих возмущениям напряженно-деформированного состояния, обусловленным наличием волокна конечных размеров.

$$\sigma_{ij}^{0m} = \sigma_{ij}^{\infty} + \sigma_{ij}^{10m}, \quad u_j^{0m} = u_j^{\infty} + u_j^{10m}, \quad (2)$$

где

$$\sigma_{11}^{\infty} = -P, \quad \sigma_{22}^{\infty} = 0, \quad \sigma_{12}^{\infty} = 0, \quad u_1^{\infty} = A_1 x_1, \quad u_2^{\infty} = A_2 x_2. \quad (3)$$

Определение докритического состояния проводится с использованием указанных величин и основных соотношений, которые включают условия непрерывности векторов напряжений и перемещений на границах раздела компонентов композита

$$\sigma_{11}^{\infty} + \sigma_{11}^{10m} = \sigma_{11}^{0a}, \quad \sigma_{12}^{10m} = \sigma_{12}^{0a}, \quad u_j^{\infty} + u_j^{10m} = u_j^{0a}, \quad x_1 = \pm L/2, \quad |x_2| \leq \pm D/2, \quad (4)$$

$$\sigma_{22}^{10m} = \sigma_{22}^{0a}, \quad \sigma_{12}^{10m} = \sigma_{12}^{0a}, \quad u_j^{\infty} + u_j^{10m} = u_j^{0a}, \quad |x_1| \leq \pm L/2, \quad x_2 = \pm D/2, \quad (5)$$

а также условия затухания «на бесконечности»

$$\sigma_{ij}^{10^m} \rightarrow 0, \quad u_j^{10^m} \rightarrow 0, \quad \text{ïðå} \quad \sqrt{x_1^2 + x_2^2} \rightarrow \infty. \quad (6)$$

Исследование задачи устойчивости выполняется с применением статического метода ТЛТУДТ в рамках второго варианта теории малых докритических деформаций, при моделировании наполнителя и матрицы линейно-упругим изотропным телом, что согласовано с постановкой задачи определения докритического напряженно-деформированного состояния.

Таким образом, уравнения устойчивости и составляющие несимметричного тензора напряжений можно представить в следующем виде:

$$\frac{\partial}{\partial x_i} \left(\omega_{ij\alpha\beta} \frac{\partial}{\partial x_\beta} u_\alpha \right) = 0, \quad t_{ij} = \omega_{ij\alpha\beta} \frac{\partial}{\partial x_\beta} u_\alpha, \quad i, j, \alpha, \beta = 1, 2. \quad (7)$$

Для компонентов тензора ω используются следующие выражения:

$$\omega_{ij\alpha\beta} = \delta_{ij}\delta_{\alpha\beta}\lambda + (\delta_{i\beta}\delta_{\alpha j} + \delta_{i\alpha}\delta_{\beta j})\mu + \delta_{\alpha j}\sigma_{i\beta}^0, \quad \sigma_{i\beta}^0 = \delta_{i\beta}\sigma_{\beta\beta}^0, \quad (8)$$

где введены общеизвестные обозначения λ и μ — постоянные Ляме.

При исследовании задачи устойчивости соотношения (7) применяются для матрицы, если записать их относительно величин σ_{ij}^{1m} , ε_{ij}^{1m} , u_j^{1m} , $\omega_{ij\alpha\beta}^{1m}$, λ_m , μ_m , а также — для волокна, если записать их относительно величин σ_{ij}^a , ε_{ij}^a , u_j^a , $\omega_{ij\alpha\beta}^a$, λ_a , μ_a .

Таким образом, в соответствии с (2), (3) и (8) имеют место следующие выражения для матрицы и волокна соответственно:

$$\begin{aligned} \omega_{ij\alpha\beta}^{1m} &= \delta_{ij}\delta_{\alpha\beta}\lambda_m + (\delta_{i\beta}\delta_{\alpha j} + \delta_{i\alpha}\delta_{\beta j})\mu_m + \delta_{\alpha j}\sigma_{i\beta}^{0m}, \\ \sigma_{i\beta}^{0m} &= -\delta_{i\beta}\delta_{\beta 1}P + \sigma_{i\beta}^{10^m}; \end{aligned} \quad (9)$$

$$\omega_{ij\alpha\beta}^a = \delta_{ij}\delta_{\alpha\beta}\lambda_a + (\delta_{i\beta}\delta_{\alpha j} + \delta_{i\alpha}\delta_{\beta j})\mu_a + \delta_{\alpha j}\sigma_{i\beta}^{0a}. \quad (10)$$

Полная формулировка задачи устойчивости включает также условия непрерывности векторов напряжений и перемещений на границах раздела, которые для описанной расчетной схемы можно представить в следующем виде:

$$t_{11}^{1m} = t_{11}^a, \quad t_{12}^{1m} = t_{12}^a, \quad u_1^{1m} = u_1^a, \quad u_2^{1m} = u_2^a, \quad x_1 = \pm L/2, \quad |x_2| \leq \pm D/2, \quad (11)$$

$$t_{22}^{1m} = t_{22}^a, \quad t_{21}^{1m} = t_{21}^a, \quad u_1^{1m} = u_1^a, \quad u_2^{1m} = u_2^a, \quad |x_1| \leq \pm L/2, \quad x_1 = \pm D/2, \quad (12)$$

а также условия затухания «на бесконечности», которые для описанной расчетной схемы имеют вид

$$u_j^{1m} \rightarrow 0, \quad \text{при} \quad \sqrt{x_1^2 + x_2^2} \rightarrow \infty. \quad (13)$$

Заметим, что докритическое состояние в ситуации, представленной на рис. 1, соответствует задаче о концентрации напряжений (при осевом нагружении) возле прямоугольного включения, которое заполнено материалом с отличительными характеристиками. В этом случае докритическое состояние будет неоднородным.

Таким образом, для плоской деформации получаем задачу на собственные значения для системы уравнений в частных производных с переменными коэф-

фициентами, зависящими от двух переменных — x_1 и x_2 . В таком случае получить решение соответствующей задачи на собственные значения с использованием аналитических методов невозможно, что обуславливает необходимость применения численных методов.

При исследовании устойчивости композитного материала под действием сжимающей нагрузки выполняются условия самосопряженности операторов задачи определения начального напряженно-деформированного состояния (1)–(6) и обобщенной задачи на собственные значения, соответствующей задаче трехмерной теории устойчивости (7)–(13). Кроме того, дифференциальные операторы рассмотренных задач являются положительно-определенными при условии отсутствия жесткого смещения. Сходимость вычислительных процессов и выбор методов решения дискретных задач выполняется в зависимости от свойств полученных разностных операторов. Поэтому разностные задачи целесообразно строить так, чтобы операторы сохраняли самосопряженность и положительную определенность дифференциальных операторов.

Построение дискретной модели

Рассмотрим методику численного решения задачи трехмерной линеаризованной теории устойчивости и задачи линейной теории упругости методом конечных разностей с применением концепции базовых схем. Такой подход подробно описан применительно к широкому классу задач механики деформируемого твердого тела в работах [8, 9], а применительно к решению задач трехмерной устойчивости композитных материалов армированных волокнами конечных размеров впервые предложен в работе [10].

В качестве основных математических моделей принимаются уравнения линейной теории упругости кусочно-однородных сред для определения докритического напряженно-деформированного состояния; уравнения трехмерной линеаризованной теории устойчивости при малых деформациях для определения критических параметров. В качестве механической модели применяется модель кусочно-однородной среды. Компоненты композита моделируются линейно-упругими изотропными телами.

Предложенный способ построения дискретных моделей базируется на использовании концепции базовых схем. Базовая схема — это разностная схема, построенная на шаблоне ячейки сетки, которая позволяет выписать в явном виде систему сеточных уравнений на шаблоне ячейки сетки, соответствующую произвольной задаче из рассматриваемого класса задач. Путем суммирования соответствующих значений базовых схем в каждом узле рассматриваемой сеточной области получается разностная задача, соответствующая конкретной задаче из рассматриваемого множества.

Далее рассмотрим методику построения дискретных моделей, при этом ограничимся только случаем плоских задач. Для этого в расчетной области $\bar{\Omega}$ с помощью прямых $x_i = \text{const}$ введем неравномерную по каждому направлению x_i разностную сетку так, чтобы в пределах ячейки сетки материал был однородным

$$\bar{\omega} = \{ x = (x_1^m, x_2^n), \quad m = \overline{0, M}, \quad n = \overline{0, N} \}. \quad (14)$$

Таким образом, сеточная область $\bar{\omega} = \omega \cup \gamma$, где ω — множество внутренних, γ — множество граничных узлов, представляет собой совокупность прямоугольных ячеек сетки. Каждая из ячеек может быть наделена механическими и геометрическими характеристиками компонента композита, который содержится в данной ячейке (рис. 2). На сетке $\bar{\omega}$ заменим интегралы соответствующими суммами по формулам трапеций, а подынтегральные континуальные величины — разностными. Сеточные переменные и функции, определенные в узлах сетки $\bar{\omega}$, будем обозначать так же, как и соответствующие континуальные величины. Выделим ячейку сетки, пронумеруем ее узлы и поставим в соответствие каждому узлу векторный параметр $\xi = (\xi_1, \xi_2) = (\pm 1, \pm 2)$ (рис. 2).

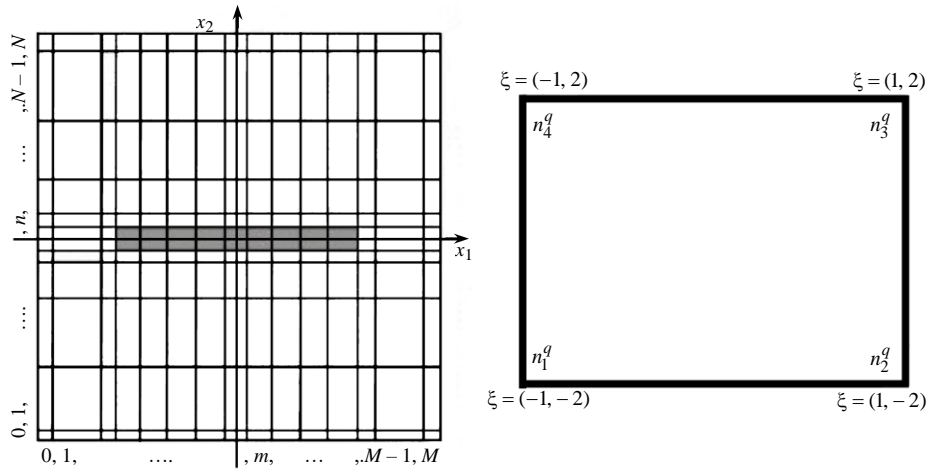


Рис. 2

Очевидно, что к каждому узлу сетки $\mathbf{x} \in \bar{\omega}$ примыкает определенное количество ячеек (от 1 до 4). При этом ячейки могут относиться к различным компонентам композитного материала.

Построение разностной модели

Получение разностной модели дискретной задачи на сетке получаем вариационно-разностным способом с применением концепции базовых схем. Компоненты базовых схем определяются на шаблоне ячейки сетки путем аппроксимации и минимизации функционалов, соответствующих задачам (1)–(13). Следует отметить, что при реализации указанной процедуры применительно к исследованию задачи устойчивости используются вариационные принципы ТЛТУДТ. Путем суммирования значений базовых схем в каждом узле сеточной области получаются разностные задачи, являющиеся дискретными аналогами соответствующих континуальных задач.

Таким образом, операторную форму разностной задачи определения докритического напряженно-деформированного состояния, которая соответствует задаче линейной теории упругости (1)–(6), можно записать в следующем виде:

$$Au = \Phi, \quad \mathbf{x} \in \bar{\omega} \quad \Leftrightarrow \quad A_i u = \Phi_i, \quad \mathbf{x} \in \bar{\omega}, \quad (15)$$

где

$$A_i u = \sum_{\xi \in \mathbf{x}} a_i(\xi) u, \quad \Phi_i = \sum_{\xi \in \mathbf{x}} \varphi_i(\xi), \quad \mathbf{x} \in \bar{\omega}. \quad (16)$$

В (16) знак суммы означает, что в узле $\mathbf{x} \in \bar{\omega}$ суммируются базовые операторы для тех параметров ξ , которые соответствуют узлу \mathbf{x} во всех смежных к выбранному узлу ячейках.

Выражения базовых операторов, полученные на шаблоне ячейки сетки, имеют вид

$$a_i(\xi) u = -H \frac{\sigma_{ji} + \sigma_{ji}^{\xi_j}}{\eta_{\xi_j}}, \quad \varphi_i = -H \left(F_i^0 + \frac{2}{\eta_{\xi_j}} P_{ij}^0 \right), \quad \mathbf{x} \in \gamma,$$

где $H = h_1 h_2$ — площадь ячейки сетки, $h_i = \eta_{\xi_i} \text{sign} \xi_i > 0$.

Разностный оператор задачи (15) сохраняет свойства самосопряженности и положительной определенности соответствующего дифференциального оператора. Таким образом, задача определения докритического напряженно-деформированного состояния сводится к решению сеточных уравнений (15), которые могут быть

представлены в виде системы линейных уравнений с симметричной разреженной матрицей, состоящей, в основном, из нулевых элементов, кроме элементов, размещенных на нескольких диагоналях.

Операторная форма разностной задачи, соответствующая дифференциальной задаче устойчивости (7)–(13), может быть представлена в следующем виде:

$$Au = pBu, \mathbf{x} \in \bar{\omega} \quad A_i u = pB_i u, \mathbf{x} \in \bar{\omega}, \quad (17)$$

$$\text{где } A_i u = \sum_{\xi \in X} a_i(\xi) u, \quad B_i u = \sum_{\xi \in X} b_i(\xi) u, \quad \mathbf{x} \in \bar{\omega}.$$

Выражения базовых операторов, полученные на шаблоне ячейки сетки, имеют вид

$$a_i(\xi) u = -H \frac{\sigma_{ji} + \sigma_{ji}^{\xi_j}}{\eta_{\xi_j}}, \quad b_i(\xi) u = -H \frac{\sigma_{jk}^0 u_{i,\xi_k} + (\sigma_{jk}^0 u_{i,\xi_k})^{\xi_j}}{\eta_{\xi_j}},$$

где $H = h_1 h_2$ — площадь ячейки сетки, $h_i = \eta_{\xi_i} \text{sign} \xi_i > 0$.

Разностные операторы задачи (17) сохраняют свойства самосопряженности и положительной определенности соответствующих дифференциальных операторов. Таким образом, задача устойчивости сводится к решению сеточных уравнений (17), которые могут быть представлены в виде алгебраической обобщенной задачи на собственные значения.

В силу сохранения свойств дифференциальных операторов при описанном способе построения разностных задач, как указывалось выше, матрицы соответствующей обобщенной задачи на собственные значения A и B являются разреженными симметричными положительно-определенными.

Описанный способ построения алгебраических задач существенно упрощает процедуру приближенного решения конкретных задач, так как позволяет использовать общие выражения базовых операторов и базовых матриц для различных задач рассматриваемого класса. Кроме того, описанный способ суммирования является общим для всех рассматриваемых классов задач, что позволяет легко автоматизировать процесс получения алгебраических задач. Для получения численных результатов решения алгебраических задач используются известные в теории разностных схем прямые и итерационные методы.

Таким образом, в рамках модели ВКР с применением основных соотношений трехмерной линейаризированной теории устойчивости деформируемых тел выполнены исследования устойчивости композитных материалов, армированных короткими волокнами различной конфигурации.

На основании изложенной методики проведен анализ численных результатов решения задач устойчивости композитных материалов: для одного волокна, двух волокон, периодических рядов одинаковых волокон, а также для короткого волокна возле свободной граничной поверхности. Исследовано влияние механических и геометрических параметров компонентов композита на величину критической деформации и формы потери устойчивости, а также обнаружены новые механические эффекты. Подробно результаты указанных исследований изложены в [6, 10–14]. Следует отметить, что для получения компьютерных результатов приемлемой точности для каждого из рассмотренных классов задач расчет выполняется на последовательности сеток при их сгущении до уровня, когда величины критических нагрузок практически не зависят от дальнейшего сгущения сеток. В связи с этим возникает необходимость для решения АПСЗ использовать современные высокопроизводительные вычислительные средства.

Численное исследование задачи гибридным алгоритмом метода итераций на подпространстве

Метод итераций на подпространстве применяется для нахождения r минимальных собственных значений и соответствующих им собственных векторов задачи

$$Ax = \lambda Bx \quad (18)$$

с ленточными симметричными матрицами. Этот метод является обобщением метода обратных итераций и заключается в построении для задачи (18) последовательности подпространств E_t ($t = 1, 2, \dots$), которая сходится к подпространству E_∞ , содержащему искомые собственные векторы [15]. На t -й итерации вычисляется ортогональный базис подпространства E_t и, если достигнута требуемая точность приближенного решения, определяются искомые собственные пары.

Таким образом, реализация метода итераций на подпространстве сводится к решению для $t = 1, 2, \dots$ таких подзадач:

— нахождение решения системы линейных алгебраических уравнений (СЛАУ):

$$AX_t = Y_{t-1}; \quad (19)$$

— вычисление проекции матрицы A на подпространство E_t :

$$A_t = X_t^T Y_{t-1} \equiv X_t^T A X_t; \quad (20)$$

— вычисление прямоугольной матрицы:

$$W_t = B X_t; \quad (21)$$

— вычисление проекции матрицы B на подпространство E_t :

$$B_t = X_t^T W_t \equiv X_t^T B X_t; \quad (22)$$

— решение полной проблемы собственных значений для проекций:

$$A_t Z_t = B_t Z_t \Lambda_t; \quad (23)$$

— вычисление следующего приближения:

$$Y_t = W_t Z_t. \quad (24)$$

Если после k итераций выполняются условия окончания итерационного процесса, например $\left| \frac{\lambda_i^{(c)} - \lambda_i^{(c-1)}}{\lambda_i^{(t)}} \right| \leq \varepsilon$, то проводится дополнительная итерация и в качестве приближенных решений задачи (18) положим $\lambda_i^* = \lambda_i^{(c+1)}$ ($i = 1, 2, \dots, r$)

и первые r столбцов матрицы $X^* = X_{c+1} Z_{c+1}$ (имеется в виду, что собственные значения упорядочены по возрастанию $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_r \leq \dots$).

Как известно [15], итерационный процесс сходится линейно, причем скорость сходимости λ_i определяется отношением λ_q / λ_1 , где q — размерность итерируемого подпространства E_t . В последовательных реализациях алгоритма рекомендуется выбирать $q = \min(2r, r+8)$.

Поскольку на каждой итерации выполняется решение СЛАУ (19) с одной и той же матрицей (A), треугольное разложение (например, LL^T) этой матрицы осуществляется один раз до начала итерационного процесса. Тогда на каждой итерации будет решаться задача (19) с факторизованной матрицей.

Реализация гибридного алгоритма метода итераций на подпространстве

Рассматривается гибридный компьютер, архитектура которого состоит из многоядерного MIMD-компьютера (CPU) с топологией коммуникаций между процессорами («гиперкуб») и графических SIMD-процессоров (GPU). При этом каждый из p вычислительных процессов на ядрах CPU с логическими номерами $i = 0, 1, 2, \dots$ взаимодействует с одним процессором GPU с таким же номером. Такая архитектура часто обозначается $p\text{CPU}+p\text{GPU}$.

Распределение данных между процессорными устройствами. Для решения задачи (18) на гибридном компьютере матрица A разбивается на квадратные блоки $A_{l,j}$ порядка s . Элементы (главной диагонали и нижнего или верхнего треугольника, в зависимости от используемых алгоритмов) ненулевых блоков ленточной симметричной матрицы распределяются между CPU в соответствии с одномерной блочно-циклической схемой. Согласно этой схеме блок $A_{l,j}$ хранится в CPU с логическим номером $(l+j) \bmod p$ (результат операции $k \bmod j$ — остаток от деления k на j , $-1 \leq l \leq p-2$ — сдвиг, обычно $l = -1$). Таким же образом распределяются блоки нижней треугольной матрицы L или верхней треугольной матрицы L^T (строками блоков матрицы порядка s [16]).

Такая же блочно-циклическая схема распределения используется для элементов матрицы B и прямоугольных матриц итерируемых векторов X_t, Y_t, W_t . При этом достаточно распределять и хранить только ненулевые элементы матрицы B в такой последовательности: поддиагональные, диагональные и наддиагональные. Это значительно упрощает алгоритм умножения такой матрицы на прямоугольную матрицу, несущественно увеличивая общий объем данных.

В памяти GPU хранятся копии тех блоков (в соответствии с распределением между CPU) матриц A, B, X_t, Y_t, W_t , которые используются для выполнения операций над ними.

Планирование вычислений на гибридном компьютере. На основе проведенного анализа метода итераций на подпространстве и в целях эффективного использования архитектурных и технологических особенностей гибридного компьютера этапы решения задачи (18) разделим на четыре подзадачи (разной вычислительной сложности) [17].

1. Формирование распределенной между процессами матрицы Y_0 начальных итерируемых векторов, таких, чтобы матрица B_1 из (22) была положительно-определенной. Поскольку подматрицы E_t итерированных векторов распределены между процессами на CPU блоками строк, эту операцию можно выполнить на каждом из них без обменов, используя, например, предложенный в [15] параллельный алгоритм.

2. LL^T -факторизация ленточной симметричной положительно-определенной матрицы A (на CPU и GPU) с помощью, например, гибридного блочного алгоритма [18].

3. Выполнение итерационного процесса (19)–(24), в котором для каждого $t = 1, 2, \dots$ вычисления распределены между процессорными устройствами следующим образом.

3.1. Решение СЛАУ (19) с помощью полученного ранее LL^T -разложения матрицы A , т.е. решаются гибридным алгоритмом две системы с треугольными матрицами $LV = Y_{t-1}$ и $L^T X_t = V$ [18].

3.2. Вычисление прямоугольной матрицы $W_t = BX_t$ по формуле (21), т.е. произведения прямоугольной матрицы на ленточную матрицу, которое выполняется в таком порядке:

— пересылка в каждый процесс от процессов, где они постоянно расположены, элементов (строк) прямоугольной матрицы, которые необходимы для вычисления очередных kp строк матрицы W_t ;

— вычисление частных сумм для элементов очередных kp строк матрицы W_t выполняется в каждом процессе с помощью соответствующего ему процессора GPU;

— мультиобмен очередных kp строк матрицы W_t , т.е. мультисборка строк из частных сумм элементов, которую можно совместить с распределением их между процессами в соответствии схемы хранения.

3.3. Вычисления (20) и (22) произведений прямоугольных матриц для формирования проекций матриц A и B на подпространство выполняются GPU, а сборка

матриц проекций каждым процессом — CPU. Причем вычисление произведения прямоугольных матриц осуществляется асинхронно с другими вычислительными операциями или обменами.

3.4. Решение полной обобщенной АПСЗ (23) методом Якоби, учитывая сравнительно небольшой порядок матриц проекций, выполняется каждым процессом на CPU без обменов данными между вычислительными устройствами [16].

3.5. Проверка условий окончания итерационного процесса выполняется каждым процессом на CPU.

3.6. Вычисление (24) новой матрицы итерированных векторов Y_t (или матрицы приближенных собственных векторов X^*) выполняется на GPU в соответствии с распределением данных (вычисляется подматрица матрицы Y_t или X^*), причем нет необходимости в обмене данными между процессорными устройствами.

4. Исследование достоверности результатов — вычисление оценок погрешности полученного решения по технологии, предложенной в [16]. Для вычисления оценок относительной погрешности вычисленных приближений к собственным значениям выполняются операции, аналогичные операциям, выполняемым в итерационном процессе. Для реализации этих операций используются те же гибридные алгоритмы, что и в итерационном процессе.

Результатом работы гибридного алгоритма метода итераций на подпространстве являются

— вычисленные значения собственных значений λ_i ; расположены в каждом процессе в порядке возрастания;

— матрица собственных векторов X , соответствующих вычисленным собственным значениям (элементы этой матрицы будут расположены в процессах соответственно одномерной блочно-циклической схемы, которая совпадает со схемой распределения элементов матриц A и B).

Эффективность гибридного алгоритма

Исследование эффективности предложенного гибридного алгоритма базируется на методологиях из [16, 17]. Эффективность параллельных алгоритмов оценивается коэффициентами ускорения $S_p = T_1/T_p$ и эффективности $E_p = S_p/p$. Здесь (для гибридной архитектуры) T_1 — время решения задачи на архитектуре 1 CPU + 1 GPU, T_p — время решения этой же задачи на архитектуре p CPU + p GPU.

Время решения задачи разработанным алгоритмом вычисляется следующими формулами:

$$\begin{aligned} T_1 &= O_1 t_C + O_{1G} t_G / n_o + O_o t_o + O_c t_c + O_{cG} t_{cG}, \\ T_p &= O_p t_C + O_{pG} t_G / n_o + O_o t_o + O_o t_o + O_o t_o + O_c t_c + O_{cG} t_{cG}, \end{aligned} \quad (25)$$

где t_C , t_G — среднее время выполнения на CPU и GPU соответственно одной операции с плавающей точкой; n_o — количество арифметических операций, которое может быть выполнено одновременно на GPU; t_o , t_oG — время, необходимое для обмена одним машинным словом между MPI-процессами на CPU или между CPU и GPU соответственно; O_o , O_oG — объем (количество машинных слов) обменов, которые выполняются одним CPU и GPU соответственно; t_c , t_cG — время, необходимое для синхронизации двух MPI-процессов или одного MPI-процесса и его GPU соответственно; O_c , O_cG — количество таких синхронизаций, которые выполняются одним MPI-процессом.

При исследовании и решении частичной обобщенной АПСЗ с симметричными матрицами (18) часть этапов (формирование матрицы начальных итерирован-

ных векторов, LL^T -разложение матрицы A , вычисление оценок приближенного решения) имеет фиксированное количество арифметических операций, а для итерационного процесса (19)–(24) количество арифметических операций пропорционально количеству выполненных итераций. Число итераций, необходимое для нахождения r минимальных собственных значений, как правило, определяется величиной $O(r)$, причем оно тем меньше, чем больше размер q итерированного подпространства. Таким образом, $T_p = T_p^{(F)} + c_I T_p^{(II)}$, а коэффициент предложенного алгоритма можно представить в виде

$$S_p = \frac{T_1}{T_p} = \frac{T_p^{(F)}}{T_p} S_p^{(F)} + \frac{c_I T_p^{(II)}}{T_p} S_p^{(II)}, \quad (26)$$

где c_I — количество итераций, $T_p^{(F)}$, $T_p^{(II)}$ — соответственно время решения подзадачи с фиксированным количеством арифметических операций и время выполнения одной итерации на архитектуре p CPU и p GPU, $S_p^{(F)}$, $S_p^{(II)}$ — коэффициенты ускорения алгоритмов соответствующих подзадач.

Далее введем следующие обозначения: $T_p^{(LLT)}$ — время выполнения LL^T -разложения матрицы A ; $T_p^{(SS)}(k)$ — время, необходимое для решения СЛАУ вида (19) с k правыми частями (используется ранее вычисленное LL^T -разложение матрицы системы); $T_p^{(Fo)}$, $T_p^{(Io)}(k)$ — соответственно времена выполнения остальных операций при решении подзадач с фиксированным количеством арифметических операций и одной итерации. Тогда $T_p^{(F)}$, $T_p^{(II)}$ можно представить в виде

$$T_p^{(F)} = T_p^{(LLT)} + T_p^{(SS)}(q) + T_p^{(Fo)}, \quad T_p^{(II)} = T_p^{(SS)}(q) + T_p^{(Io)}(q). \quad (27)$$

Заметим, что здесь и далее формулы для всех $T_1^{(*)}$ можно получить, подставив в формулы для $T_p^{(*)}$ значения $p = 1$. Для всех этих времен справедливы формулы (25) устойчивости слоисто-композитного материала при сжатии поверхностной нагрузкой. Тогда коэффициент ускорения гибридного алгоритма метода итераций на подпространстве для архитектуры p CPU + p GPU, базирясь на (26), можно вычислить по формуле

$$S_p = \frac{T_p^{(F)}}{T_p} \left(\alpha_p^{(LLT)} S_p^{(LLT)} + \alpha_p^{(SS)}(q) S_p^{(SS)}(q) + \alpha_p^{(Fo)} S_p^{(Fo)} \right) + \frac{c_I T_p^{(II)}}{T_p} \left(\beta_p^{(SS)}(q) S_p^{(SS)}(q) + \beta_p^{(Io)}(q) S_p^{(Io)}(q) \right), \quad (28)$$

где $\alpha_p^{(*)} = T_p^{(*)} / T_p^{(F)}$, $\beta_p^{(*)} = T_p^{(*)} / T_p^{(II)}$, $S_p^{(*)}$ — коэффициент ускорения решения соответствующей подзадачи (выполнения операций), а верхние индексы и параметры совпадают с соответствующими индексами и параметрами в (28) для времен $T_p^{(*)}$.

Поскольку рассматривается алгоритм для ленточных матриц, введем следующие обозначения: m_A, m_B — полуширина ленты матрицы A и B соответственно, η_B — среднее количество ненулевых элементов в одной строке матрицы B , $t_C^{(E)}(d) = t_s + dt_o$ и $t_G^{(E)}(d) = t_{sG} + dt_{oG}$ — времена обмена массивом с d двойных слов между двумя CPU и CPU + GPU соответственно.

Лемма 1. При использовании p CPU и p GPU для решения СЛАУ с ленточной симметричной положительно-определенной матрицей A размера $n \gg q > r$, $m_A > sp$ гибридным алгоритмом LL^T -разложения на компьютерной конфигурации «гиперкуб» справедливы такие оценки временных характеристик:

$$T_p^{(LLT)} \approx \frac{n}{p} \left(\frac{s^2}{3} pt_C + \max \left\{ m_A^2 \frac{t_G}{n_o}, t_C^{(E)}(sm_A) \frac{p}{s} \log_2 p \right\} + \right. \\ \left. + 2psm_A \frac{t_G}{n_o} + t_G^{(E)}(sm_A) \frac{p}{s} \right), \quad (29)$$

$$T_p^{(SS)}(k) = \frac{n}{p} \left(kt_C p \log_2 p + (4m_A + 2ps)k \frac{t_G}{n_o} + \right. \\ \left. + (t_C^{(E)}(sk) \log_2 p + 2t_G^{(E)}(sk)) \frac{2p}{s} \right). \quad (30)$$

Из (29), (30), а также, учитывая, что $t_G^{(m)}(s) = \frac{m_A^2 s}{p \log_2 p n_o} \frac{t_G}{n_o}$, получим оценки коэффициентов ускорения гибридного алгоритма LL^T -разложения ленточной симметричной матрицы:

$$S_p^{(LLT)} \approx p \left(1 - \frac{2s(p-1)}{m_A + 2ps} \right), \quad t_G^{(m)}(s) \geq t_C^{(E)}(sm_A), \\ S_p^{(LLT)} \approx p \left(\frac{2ps}{m_A + 2s} + \frac{p \log_2 p}{m_A s (m_A + 2s)} \frac{n_o t_C^{(E)}(sm_A)}{t_G} \right)^{-1}, \quad (31) \\ t_G^{(m)}(s) < t_C^{(E)}(sm_A), \\ S_p^{(SS)}(k) \approx p \left(1 + \frac{(p-1)s}{2m_A + s} + \frac{p \log_2 p}{ks(2m_A + s)} \frac{n_o t_C^{(E)}(ks)}{t_G} \right)^{-1}.$$

Справедливо следующее утверждение.

Лемма 2. Для ленточной симметричной матрицы A при условии $n \gg q > r$, $m_A \geq m_B$ и $1 \leq \eta_B \ll m_B$, для времени выполнения с фиксированным количеством арифметических операций ($T_p^{(Fo)}$) и времени выполнения одной итерации ($T_p^{(Io)}(k)$) справедливы оценки

$$T_p^{(Fo)} = \frac{n}{p} \left(\frac{n + 2(n+2)p \log_2 p}{n} qt_C + (2\eta_B + 6)q \frac{t_G}{n_o} \right) + \\ + \left(2t_C^{(E)}(sq) \log_2 p + 2t_G^{(E)}(sq) \right) \frac{n}{s}, \quad (32)$$

$$T_p^{(Io)}(k) = \frac{n}{p} \left(\frac{(n+2k) \log_2 p + O(k^2)}{n} p k t_C + (2\eta_B + 4k)k \frac{t_G}{n_o} \right) + \\ + \left(t_C^{(E)}(sk) \log_2 p + 2t_G^{(E)}(sk) \right) \frac{n}{s}. \quad (33)$$

Поскольку в (32) значение $T_p^{(Fo)}$ существенно меньше значений $T_p^{(LLT)}$ и $T_p^{(SS)}(q)$ из (27), им можно пренебречь, положив в (28) $\alpha_p^{(Fo)} \cong 0$.

Влияние $T_p^{(Io)}(g)$ зависит от соотношений m_A, η_B и q : если q существенно больше η_B , то в (33) можно пренебречь слагаемым, содержащим η_B ; если $m_A \gg \eta_B$ и $m_A \gg q$, то в (27) можно пренебречь $T_p^{(Io)}(g)$. Учитывая временные характеристики из (33), получим следующую оценку:

$$S_p^{(Io)}(k) \approx p \left(1 + \frac{p \log_2 p}{2k(\eta_B + 2k)} \frac{n_o}{t_G} (kt_C + t_C^{(E)}(sk)) \right)^{-1}. \quad (34)$$

Справедлива следующая теорема.

Теорема. Для гибридного алгоритма метода итерации на подпространстве решения частичной АПСЗ с ленточной матрицей, при условии $n \gg q$, $m_A \geq m_B$, $m_A \gg q$, $m_A > sp$ и $\eta_B \ll m_B$, справедливы такие оценки коэффициентов ускорения:

$$S_p \approx \frac{\gamma_p}{\gamma_p + c_I} (\alpha_p^{(LLT)} S_p^{(LLT)} + \alpha_p^{(SS)}(q) S_p^{(SS)}(q)) + \frac{c_I}{\gamma_p + c_I} \times \\ \times (\beta_p^{(SS)}(q) S_p^{(SS)}(q) + \beta_p^{(Io)}(q) S_p^{(Io)}(q)), \quad (35)$$

где $\gamma_p = (T_p^{(LLT)} + T_p^{(SS)}(q)) / T_p^{(II)}$, а коэффициенты $\alpha_p^{(*)} = T_p^{(*)} / T_p^{(F)}$ и $\beta_p^{(*)} = T_p^{(*)} / T_p^{(II)}$ вычисляются формулами (29), (30), (32), (33), причем в полученных выражениях составляющие малого порядка по отношению к другим отбрасываются; коэффициенты ускорения $S_p^{(*)}$ соответствующей подзадачи представлены формулами (31), (34).

Анализ оценок (28)–(35) свидетельствует, что эффективность предложенного гибридного алгоритма в значительной мере определяется эффективностью алгоритма LL^T -разложения, например, гибридного блочного алгоритма, а также эффективностью алгоритмов последующего решения СЛАУ с нижней и верхней треугольными матрицами. Следует заметить, если накладные расходы на пересылку данных между вычислительными устройствами будут меньше времени вычислений, то эти алгоритмы за счет асинхронности будут иметь коэффициенты эффективности, близкие к единице.

На эффективность гибридного алгоритма решения частичной обобщенной АПСЗ методом итераций на подпространстве существенное влияние имеет количество c_I выполненных итераций для достижения требуемой точности. Обычно c_I при постоянном r тем меньше, чем больше размерность итерируемого подпространства q . Как следует из оценок (30), (33), объем вычислений пропорционален q . Однако в ряде случаев при относительно малых значениях q возникает недостаточная загруженность GPU (т.е. фактически уменьшается значение n_o) при увеличении количества обменов, что приводит к снижению реальной эффективности алгоритма. В таких случаях увеличение размерности итерируемого подпространства (пока не будет достигнуто максимальное значение n_o) позволяет сократить время решения задачи за счет сокращения количества итераций.

Результаты численного исследования задачи устойчивости на гибридных компьютерах

Решение рассматриваемой задачи устойчивости композитных материалов при использовании гибридного алгоритма метода итераций на подпространстве для решения частичной обобщенной АПСЗ ленточных матриц получено на компьютерах различных параллельных архитектур серии Инпарк, которые созданы в Институте кибернетики им. В.М. Глушкова НАН Украины совместно с Государственным предприятием «Электронмаш».

Выходные данные математических моделей задач, как правило, приближенно заданные, т.е. заданы с некоторой погрешностью. Кроме того, компьютерная модель, которую в конечном итоге и приходится решать, всегда имеет приближенные данные относительно исходной задачи (из-за наследственной погрешности в исходных данных, погрешности дискретизации и погрешности ввода числовых данных в компьютер). Поэтому в рассматриваемом гибридном алгоритме в машинной среде исследуются математические свойства компьютерной задачи (определенность или вырожденность матрицы АПСЗ) и определяется их соответствие применяемому алгоритму. Также проводится анализ точности полученных решений.

Ниже приведен фрагмент компьютерного протокола решения задачи с различными исходными данными на персональном суперкомпьютере гибридной архитектуры Инпарком_pg [19] (один вычислительный узел, два четырехъядерных процессора Xeon 5606, 2 GPU Tesla K40) при использовании двух CPU и двух GPU: порядок матриц A и B — 12282, полуширина ленты матрицы A — 6212, полуширина ленты матрицы B — 71.

Рассмотрим фрагмент протокола решения задачи на Инпарком_pg.

```
Problem solving: total time = 10.0033e+00
Results: SOLUTION WAS CALCULATED by 20 iterations
FIRST 3 EIGENVALUES
Eigenvalues (calculated) Estimates of Errors
1.248665556779830e-01 1.893e-06
1.248734577580778e-01 8.894e-07
1.248765574182433e-01 2.837e-06
```

В протоколе представлены время решения задачи, количество итераций, а также три вычисленных минимальных собственных значения. Время решения задачи гибридным алгоритмом метода итераций на подпространстве на Инпарком_pg составляет 10,0033 с.

Задача решалась также аналогичным алгоритмом из пакета MATLAB [20], получено решение за 340 с. Таким образом, решение задачи гибридным алгоритмом получено в 30 раз быстрее, чем с использованием MATLAB. Проводился сравнительный анализ решения задачи гибридным алгоритмом и последовательным, а также параллельным алгоритмом на MIMD-компьютере Инпарком-256. Для гибридного алгоритма при использовании одного GPU получено ускорение от 6,5 до 7,5 раз, а при использовании двух GPU — от 9 до 11 раз по сравнению с последовательной версией алгоритма. Результаты решения параллельным и гибридным алгоритмами показали, что при использовании восьми процессов на CPU получено ускорение в 4 раза по сравнению с последовательной версией алгоритма, а при использовании графических ускорителей — в 6 и 9 раз для одного и двух GPU соответственно.

На основании вычисленных собственных значений можно определить величины критических параметров устойчивости слоисто-композитного материала при сжатии поверхностной нагрузкой.

Заключение

В наше время круг задач, требующих для своего решения применения мощных вычислительных ресурсов, все больше расширяется. В последние годы многоядерные компьютеры с графическими процессорами стали широко использоваться для решения задач больших размеров. Эти компьютеры сравнительно недорогие и могут обеспечить значительное ускорение вычислений. Однако при решении задачи на гибридном компьютере ко времени фактического решения задачи добавляется время выполнения дополнительных операций, необходимых для обмена информацией между CPU и GPU, т.е. накладные расходы, которые могут быть очень существенные. Это связано с большими архитектурными отличиями CPU и графических процессоров. Таким образом, возникает необходимость в анализе и планировании вычислений на CPU и GPU, а также в выполнении однотипных вычислений с большими объемами данных на GPU.

В последние годы компания Intel предложила новое решение повышения производительности компьютеров — многоядерные процессоры Intel Xeon Phi первого поколения, которые используются как мощные сопроцессоры, а также процессоры Intel Xeon Phi второго поколения, которые могут использоваться и в качестве основных процессоров [21]. Новый процессор Intel Xeon Phi второго поколения стал первым загрузочным хост-процессором компании Intel, в котором впервые интегрированы память и технологии коммутации. Поэтому, в отличие от GPU-ускорителей, его функциональные возможности не ограничены коммуникационными задержками. Таким образом, процессоры Intel Xeon Phi обеспечивают еще большую эффективность и масштабируемость, способны работать в еще более разнообразных конфигурациях, чем параллельные компьютеры с использованием графических ускорителей. В последнем рейтинге Top500 параллельные компьютеры с процессорами Intel Xeon Phi второго поколения, кроме параллельных компьютеров с графическими ускорителями, занимают первые 10 позиций [1].

В Институте кибернетики совместно с «Электронмаш» создан параллельный персональный суперкомпьютер Инпарком_xp с хост-процессором Intel Xeon Phi с такими техническими характеристиками: процессор Intel Xeon Phi 7210 (64 ядра (256 потоков)), 1,3 ГГц (1,5 ГГц Турбо); оперативная память 192 Гбайт DDR4; SSD 240 Гбайт SATA3.

Реализация рассматриваемого алгоритма с учетом архитектурных особенностей этого компьютера показала высокую эффективность — время счета для приведенной выше задачи составило десять секунд [22].

О.М. Хімич, В.А. Декрет, О.В. Попов, О.В. Чистяков

ЧИСЕЛЬНЕ ДОСЛІДЖЕННЯ СТІЙКОСТІ КОМПОЗИТНИХ МАТЕРІАЛІВ НА КОМП'ЮТЕРАХ ГІБРИДНОЇ АРХІТЕКТУРИ

Розглянуто проблеми чисельного дослідження стійкості композитних матеріалів при стисненні вздовж армуючих елементів з використанням багатоядерних комп'ютерів з графічними процесорами. Описано задачу тривимірної теорії стійкості композитів із застосуванням моделі «волокон кінцевих розмірів» та математичні методи її розв'язування. Запропоновано гібридний алгоритм розв'язування часткової узагальненої проблеми власних значень для стрічкових матриць.

A.N. Khimich, V.A. Dekret, A.V. Popov, A.V. Chistyakov

NUMERICAL STUDY OF THE STABILITY OF COMPOSITE MATERIALS ON COMPUTERS OF HYBRID ARCHITECTURE

The problem of numerical investigation of the stability of composite materials when compressing along the reinforced elements using multicore computers with graphic processors is considered. The problem of the three-dimensional theory of stability of composites with the application of the «finite-size fiber» model and mathematical methods for its solution are presented. A hybrid algorithm for solving a particular generalized eigenvalue problem for tape matrices is proposed.

1. <https://www.top500.org/Lists/2017/11/>.
2. Немнюгин С.А., Стесик О.Л. Параллельное программирование для многопроцессорных вычислительных систем. — СПб.: БХВ-Петербург. — 2002. — 400 с.
3. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. — М.: ДМК Пресс, 2010. — 232 с.
4. Гузь А.Н. Основы трехмерной теории устойчивости деформируемых тел. — К.: Вища школа, 1986. — 512 с.
5. Guz A.N. Fundamentals of the Three-dimensional theory of stability of deformable bodies. — Berlin; Heidelberg; New York: Springer, 1999. — 555 p.
6. Гузь А.Н., Декрет В.А. Модель коротких волокон в теории устойчивости композитов. — Saarbruecken: Lambert Acad. Publ., 2015. — 315 с.
7. Гузь А.Н. Механика разрушения композитных материалов при сжатии. — К.: Наук. думка, 1990. — 630 с.
8. Kokhanenko Yu.V. Numerical study of three-dimensional stability problems for laminated and ribbon-reinforced composites // Int. Appl. Mech. — 2001. — **37**, N 3. — P. 317–345.
9. Гузь А.Н., Коханенко Ю.В. Численное решение задач трехмерной теории устойчивости упругих тел // Прикл. механика. — 2004. — **40**, № 11. — С. 117–126.
10. Guz A.N., Dekret V.A., Kokhanenko Yu.V. Solution of plane problems of the three-dimensional problems stability of a ribbon-reinforced composite // Int. Appl. Mech. — 2000. — **36**, N 10. — P. 1317–1328.
11. Guz A.N., Dekret V.A. Interaction of two parallel short fibers in the matrix at loss of stability // Computer Modeling in Engineering & Sciences. — 2006. — **13**, N 3. — P. 165–170.
12. Guz A.N., Dekret V.A. On two models in the three-dimensional theory of stability of composites // Int. Appl. Mech. — 2008. — **44**, N 8. — P. 839–854.
13. Guz A.N., Dekret V.A. Stability loss in nanotube reinforced composites // Computer Modeling in Engineering & Sciences. — 2009. — **42**, N 1. — P. 69–80.
14. Guz A.N., Dekret V.A. Stability problem of composite material reinforced by periodical row of short fibers // Ibid. — 2009. — **42**, N 3. — P. 177–186.
15. Парлетт Б. Симметричная проблема собственных значений. Численные методы. — М.: Мир, 1983. — 384 с.
16. Параллельные алгоритмы решения задач вычислительной математики / А.Н. Химич, И.Н. Молчанов, А.В. Попов, Т.В. Чистякова, М.Ф. Яковлев — К.: Наук. думка, 2008. — 247 с.
17. Химич А.Н., Попов А.В., Чистяков А.В. Гибридные алгоритмы решения алгебраической проблемы собственных значений с разреженными матрицами // Кибернетика и системный анализ. — 2017. — **53**, № 6. — С. 132–146.
18. Хіміч О.М., Баранов А.Ю. Гібридний алгоритм розв'язування лінійних систем зі стрічковими матрицями прямими методами // Компьютерная математика. — 2013. — **2**. — С. 80–87.
19. Інтелектуальний персональний суперкомп'ютер для розв'язування науково-технічних задач / О.М. Хіміч, І.М. Молчанов, В.І. Мова, О.О. Ніколайчук, О.В. Попов, Т.В. Чистякова, М.Ф. Яковлев, В.Г. Тульчинський, Р.А. Ющенко // Наука і інновації. — 2016. — **12**, № 4. — С. 17–31.
20. <https://www.mathworks.com/products/matlab-online.html>.
21. <https://www.intel.com/content/www/us/en/products/processors/xeon-phi>.
22. Чистяков А.В. Про деякі особливості розв'язання алгебраїчної проблеми власних значень на паралельних комп'ютерах з процесорами Intel Xeon Phi // Матеріали Міжнародної наукової конференції «Сучасна інформатика: проблеми, досягнення та перспективи розвитку», присвяченої 60-річчю заснування Інституту кібернетики імені В.М. Глушкова НАН України. — Київ, 13–15 грудня 2017. — С. 166–167.

Получено 18.04.2018