

ГРАНИЧНО-ЭЛЕМЕНТНОЕ МОДЕЛИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ЭЛЕМЕНТОВ С КВАДРАТИЧНЫМ РАСПРЕДЕЛЕНИЕМ УСИЛИЙ И РАСПАРАЛЛЕЛИВАНИЕМ ВЫЧИСЛЕНИЙ

д. ф.-м. н. Журавков М.А., к. ф.-м. н. Круподеров А.В., к. ф.-м. н. Щербаков С.С.

Белорусский государственный университет, Минск

Введение. Как известно любой из существующих численных методов решения задач механики деформируемого твердого тела и в частности задач теории упругости имеет как свои преимущества, так и недостатки. Для метода граничных элементов (МГЭ) одной из наиболее сложных проблем является решение задачи изгиба балки. На основании предыдущих исследований большого количества авторов [1-4] установлено, что при решении задач консольного изгиба с помощью МГЭ на основе схем для граничных элементов с равномерными распределениями усилий не удается добиться хорошего соответствия получаемых решений с аналитическими даже при использовании большого количества элементов. Поэтому, несмотря на простоту реализации МГЭ на основе схем для элементов с равномерными распределениями усилий, требуется использование элементов с распределениями более высокого порядка. Следует отметить, что одним из несомненных преимуществ метода граничных элементов является то, что алгоритмы его реализации хорошо распараллеливаются. В настоящее время интенсивно развивается технология распараллеливания вычислений на основе использования графических процессоров CUDA [5]. Особенностью данной технологии является, что ее использование является намного более экономным, чем, например, использование суперкомпьютеров.

В данной статье описаны результаты реализации процедуры МГЭ на основе использования схемы для элементов с квадратичными распределениями усилий. Представлены результаты распараллеливания вычислений на основе созданных для технологии CUDA алгоритмов. Приведено сравнение полученных решений с аналитическими и сравнение производительности вычислений, выполненных на одном ядре центрального процессора и распараллеленных с использованием технологии CUDA.

Схема метода граничных элементов. В соответствии с общей схемой метода граничных элементов [1-4] исходная граница исследуемого тела разбивается на граничные элементы. При использовании элементов второго порядка элемент представляет собой три узла, в общем случае занимающих произвольное положение в пространстве

Обычно при реализации метода граничных элементов необходимо проведение численного интегрирования для каждого элемента, что требует большого количества машинного времени и уменьшает точность расчета. Поэтому эффективным представляется использование аналитического интегрирования. В этом случае, правда, приходится использовать элементы, узлы которых располагаются на одной прямой. Локальную систему координат будем размещать в центральном узле.

В соответствии с общей схемой непрямого метода граничных элементов требуется найти распределения фиктивных нормальных и касательных усилий, действующих на границе области таким образом, чтобы выполнялись заданные граничные условия. При использовании элементов второго порядка соответствующие компоненты нагрузки на каждом элементе задаются следующим образом:

$$\begin{aligned} p_{\gamma}^n(x) &= a_{\gamma}^n x^2 + b_{\gamma}^n x + c_{\gamma}^n, \\ p_{\gamma}^{\tau}(x) &= a_{\gamma}^{\tau} x^2 + b_{\gamma}^{\tau} x + c_{\gamma}^{\tau}, \end{aligned} \quad (1)$$

где индекс γ соответствует номеру элемента.

Если для составления основной системы уравнений метода граничных элементов за неизвестные принять значения нагрузки в трех узлах, то на каждом элементе нагрузку можно представить в следующем виде:

$$\begin{aligned} p_{\gamma}^n(x) &= p_{\gamma_1}^n q_1(x) + p_{\gamma_2}^n q_2(x) + p_{\gamma_3}^n q_3(x), \\ p_{\gamma}^{\tau}(x) &= p_{\gamma_1}^{\tau} q_1(x) + p_{\gamma_2}^{\tau} q_2(x) + p_{\gamma_3}^{\tau} q_3(x), \end{aligned} \quad (2)$$

где $\gamma_1, \gamma_2, \gamma_3$ – индексы узлов через которые проходит элемент γ а $q_l(x)$ – функции формы, принимающие значения 1 в l -м узле и 0 в остальных узлах.

Выражения для q_l можно записать следующим образом:

$$q_1(x) = \frac{x^2}{2a^2} - \frac{x}{2a}, q_2(x) = 1 - \frac{x^2}{a^2}, q_3(x) = \frac{x^2}{2a^2} + \frac{x}{2a}, \quad (3)$$

где a – полудлина элемента.

Таким образом, требуя выполнения граничных условий в узлах граничноэлементной сетки, получаем следующую систему линейных уравнений относительно неизвестных значений фиктивных нагрузок в узлах:

$$\begin{aligned} \sum_{i=1}^{N_n} (Asnn_{ij} p_i^n + Assn_{ij} p_i^\tau) &= P_{2j}, \quad \sum_{i=1}^{N_n} (Aunn_{ij} p_i^n + Auns_{ij} p_i^\tau) = U_{2j}, \\ \sum_{i=1}^{N_n} (Asns_{ij} p_i^n + Asss_{ij} p_i^\tau) &= P_{1j}, \quad \sum_{i=1}^{N_n} (Auns_{ij} p_i^\tau + Auss_{ij} p_i^\tau) = U_{1j}, \end{aligned} \quad (4)$$

где $Asnn_{ij}, Asns_{ij}, Assn_{ij}, Asss_{ij}$ и $Aunn_{ij}, Auns_{ij}, Auns_{ij}, Auss_{ij}$ – коэффициенты влияния, соответственно, усилий и перемещений от нагрузки в i -м узле на соответствующее значение в j -м узле. Данные коэффициенты могут быть вычислены по следующим формулам:

$$\begin{aligned} Asnn_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha, \beta} a_{2\alpha}^{ij} a_{\beta 2}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha\beta}^{(n,\sigma)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \\ Asns_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha, \beta} a_{2\alpha}^{ij} a_{\beta 1}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha\beta}^{(n,\sigma)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \\ Assn_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha, \beta} a_{2\alpha}^{ij} a_{\beta 2}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha\beta}^{(\tau,\sigma)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \end{aligned} \quad (5)$$

$$\begin{aligned} Asss_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha, \beta} a_{2\alpha}^{ij} a_{\beta 1}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha\beta}^{(\tau,\sigma)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \\ Aunn_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha} a_{2\alpha}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha}^{(n,u)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \\ Asnu_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha} a_{1\alpha}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha}^{(n,u)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \\ Assn_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha} a_{2\alpha}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha}^{(\tau,u)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \\ Asss_{ij} &= \sum_{\gamma, i \in E_\gamma} \sum_{\alpha, \beta} a_{2\alpha}^{ij} a_{\beta 1}^{ij} \int_{-a_\gamma}^{a_\gamma} G_{\alpha}^{(\tau,u)}(\bar{x}_{ij} - \xi, \bar{y}_{ij}) q_{N_i}(\xi) d\xi, \end{aligned} \quad (6)$$

где $\bar{x}_{ij}, \bar{y}_{ij}$ – координаты j -го узла в системе координат, связанной с γ -м элементом; $G_{\alpha\beta}^{(n,\sigma)}, G_{\alpha}^{(n,u)}$ – функции влияния соответственно для тензора напряжений и вектора перемещений фундаментальных решений при действии нормальной нагрузки [1-4]; $G_{\alpha\beta}^{(\tau,\sigma)}, G_{\alpha}^{(\tau,u)}$ – функции влияния соответственно для тензора напряжений и вектора перемещений фундаментальных решений при действии касательной нагрузки [1-4]; N_i – номер i -го узла в соответствующем

элементе, $a_{\alpha\beta}^{\gamma j}$ – компоненты матрицы перехода от системы координат связанной с γ -м элементом к системе координат, связанной с i -м узлом.

Реализация МГЭ была произведена на языке программирования C++ в программной среде Visual Studio 2010. При этом процедуры расчета коэффициентов влияния граничных элементов друг на друга, решения системы линейных алгебраических уравнений, а также расчета компонент напряженно-деформированного состояния были распараллелены с помощью технологии CUDA.

Результаты расчетов. Рассмотрим сначала результаты расчетов на основе приведенной схемы МГЭ применительно к одной из проблемных задач МГЭ, задаче консольного изгиба балки. Схема задачи приведена на рисунке 1.

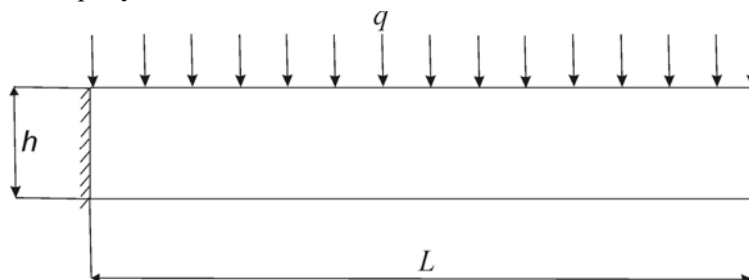


Рисунок 1 – Схема задачи о нагружении консольной балки равномерной нагрузкой

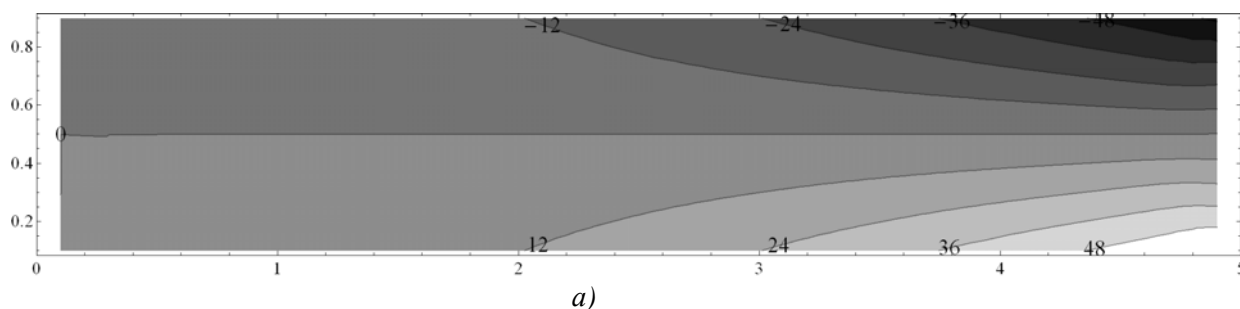
Расчеты проводились для следующих значений параметров модели: $L = 5$ м, $h = 1$ м, $q = 1$ Н/м², $E = 1$ МПа, с использованием элементов с как квадратичным, так и с равномерным распределением нагрузки. На рисунках 2-3 приведены некоторые результаты расчета в виде как собственно распределений напряжений и перемещений, так и распределений погрешностей расчета (в том числе средних, см, таблицу 1) по отношению к результатам, полученным исходя из балочной теории, вычисляемых по следующей формуле:

$$errK_i = \frac{K_i - K_i^{an}}{\max\left(\left|K_i^{an}\right|\right)_i}, \quad (7)$$

где K_i – расчетная величина компоненты в i -ой точке; K_i^{an} – величина компоненты, рассчитанная на основании балочной теории; $\max\left(\left|K_i^{an}\right|\right)_i$ – максимум модуля компоненты, рассчитанный на основе балочной теории. При расчетах балка разбивалась элементами с одинаковой длиной, равной 1/32 м.

Таблица 1– Данные по средней погрешности при равномерном нагружении и $L = 5$ м

Тип элементов	Средние погрешности по компонентам		
	$\Delta\sigma_{xx}$	$\Delta\sigma_{xy}$	Δu_y
Квадратичные элементы	0.031313	0.0683885	0.0314232
Равномерные элементы	0.0850076	0.092198	0.281024



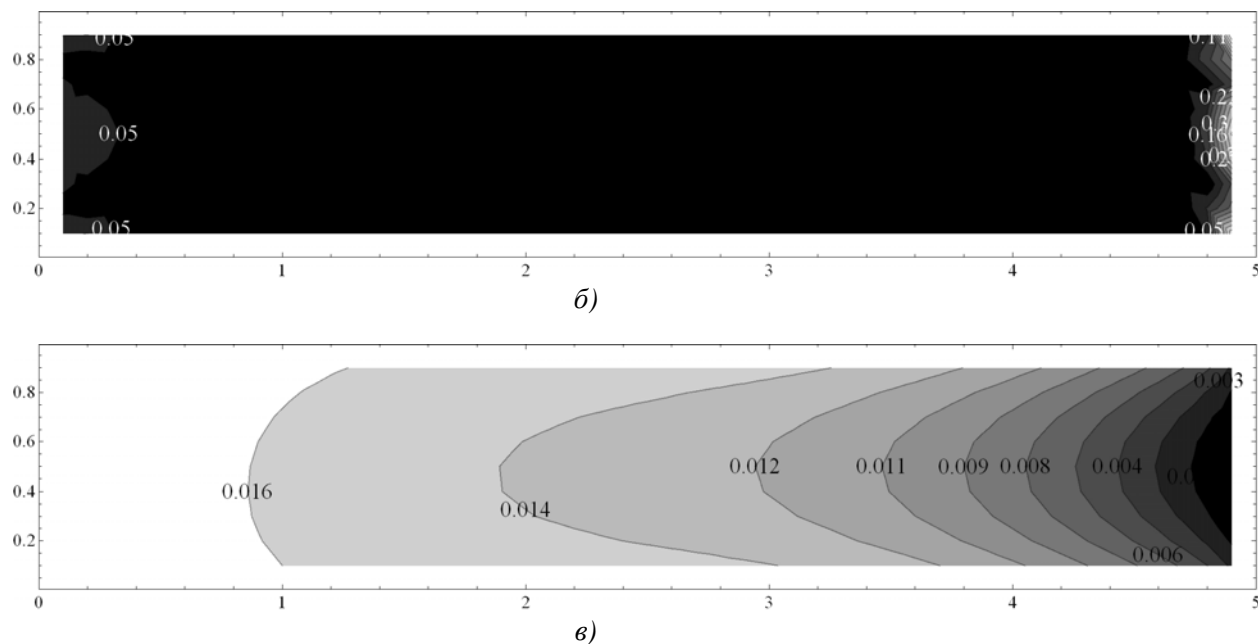


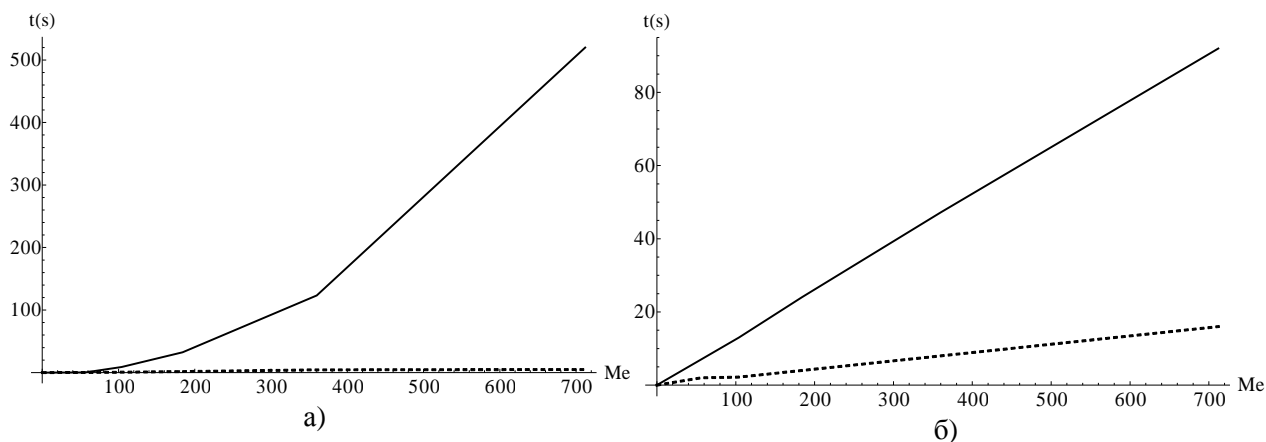
Рисунок 3 – Относительные погрешности компонент НДС по сравнению с решением по балочной теории а) σ_{xx} б) σ_{xy} в) u_2

Анализ рисунков 2 и таблицы 1 показывает, что при использовании элементов с квадратичным распределением усилий погрешность вычислений прогиба значительно ниже (максимальная примерно 4%, средняя 3%) чем при использовании элементов с равномерным распределением усилий (максимальная погрешность примерно 70%, средняя 28%).

На рисунке 4 показаны графики, показывающие эффективность распараллеливания на графические процессоры для различных процедур: сборка матрицы взаимовлияния элементов, вычисление напряжений и перемещений в расчетных точках, решение системы линейных алгебраических уравнений (СЛАУ). Пунктирными линиями показаны данные, соответствующие распараллеленным процедурам, а сплошными – нераспараллеленным.

Для расчетов использовался процессор Intel Core i7 2400 ГГц и видеокарта GeForce 660.

Анализ рисунков 4 показывает, что время параллельного расчета сокращается для больших граничноэлементных сеток в 10-100 раз по сравнению с расчетом на одном ядре. Особенно явно это проявляется для наиболее продолжительной процедуры сборки матрицы, для которой время вычислений сокращается более чем в сто раз. Данные результаты свидетельствуют о высокой эффективности распараллеливания гранично-элементных вычислений.



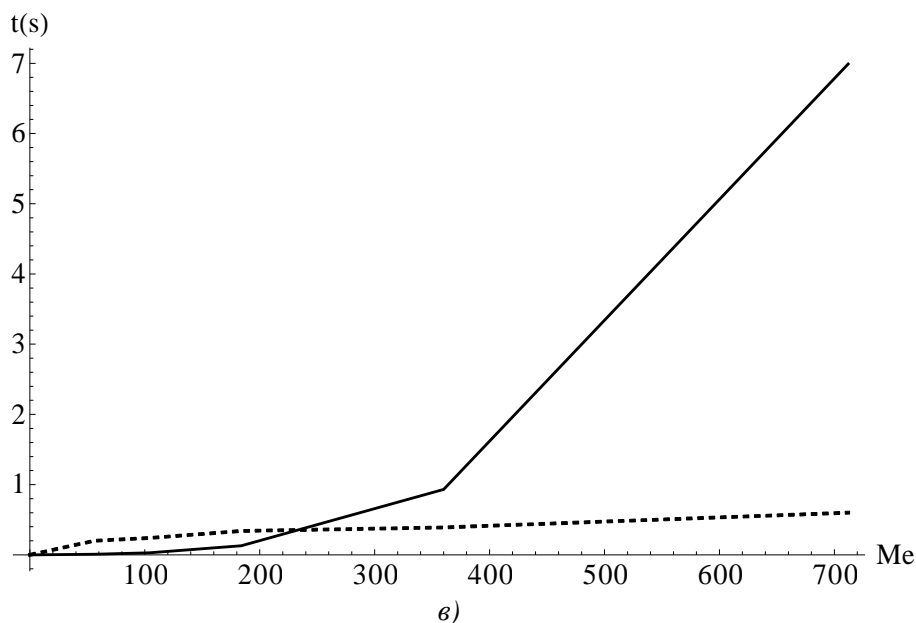


Рисунок 4 – Время а) сборки матрицы взаимовлияний, б) вычисления НДС в расчетных точках, в) решения СЛАУ в зависимости от числа граничных элементов Me

Заключение. Приведена реализация метода граничных элементов на основе схемы с квадратичным распределением поверхностных усилий. Полученные оценки погрешности результатов численного моделирования в сравнении с балочной теорией свидетельствуют о его хорошем соответствии аналитическому расчету. Приведены результаты распараллеливания вычислений на основе технологии CUDA. Показано значительное сокращение (до 100 раз) времени численного моделирования, что свидетельствует об эффективности предложенных расчетных схем.

РЕЗЮМЕ

В работе представлена реализация алгоритмов для метода граничных элементов с использованием элементов с квадратичным распределением нагрузки. Представлены результаты распараллеливания алгоритмов на графические процессоры с использованием технологии CUDA. Приведены примеры расчетов. Показана высокая эффективность распараллеливания вычислений.

ЛИТЕРАТУРА

1. Бенерджи, П. Метод граничных элементов в прикладных науках / П. Бенерджт, Р. Баттерфилд – М:Мир, 1984. – 494 с.
2. Брэббия, К. Метод граничных элементов/ К. Брэббия [и др.] – М:Мир, 1987. – 524 с.
3. Крауч, С Методы граничных элементов в механике твердого тела / С. Крауч, А. Старфилд – М:Мир, 1987. – 328 с.
4. Beer, G. Boundary element method with programming / G. Beer, I. Smith, Duenser C. – Springer, 2008. – 494 p.
5. http://www.nvidia.ru/object/cuda_home_new.html

SUMMARY

Realization of the boundary element method based on using the elements with quadric distribution of tractions is presented in the paper. The results of parallelizing of algorithms using CUDA technology are given. Examples of calculations are presented. High efficiency of calculations parallelizing is shown.

Поступила в редакцию 16.10.2013