

Параллельный алгоритм моделирования процесса динамической консолидации в грунтовых средах слоистой структуры

Богаенко В.А.

Институт кибернетики им. В.М. Глушкова, просп. Глушкова, 40, Киев, Украина

sevab@ukr.net

Аннотация. Рассматривается задача моделирования процессов динамической консолидации в грунтовых средах со слоистой структурой в двухмерном случае. Использованный конечно-элементный алгоритм решения этой задачи имеет особенности, вытекающие из необходимости учёта условий сопряжения и влияющие на эффективность его распараллеливания. Предлагаются параллельные алгоритмы для кластерных систем решения таких задач. Рассматриваются такие варианты организации вычислений как использование многопоточности на каждом из узлов кластера и ускорение вычислений с помощью графических процессоров (GPU). Приводятся данные относительно ускорения и быстродействия параллельных алгоритмов, полученные при решении модельной задачи на кластерах СКИТ-3 и СКИТ-4.

Ключевые слова

Динамическая консолидация, параллельные алгоритмы, кластеры, многопоточность, GPU, OpenCL.

1 Введение

Рассматривается задача моделирования консолидационных процессов в водонасыщенных грунтах сложной структуры, решение которой необходимо при оценке устойчивости грунтовых сооружений, находящихся под механической нагрузкой, например, дамб.

Процесс описывается начально-краевой задачей для системы уравнений для квазидвухфазных грунтов [1-3], построенной отталкиваясь от принципа эффективных напряжений Терцаги в ситуации идеальной упругопластической среды. Для учёта влияния на поведение грунтового массива неоднородностей его строения, он моделировался в виде нескольких подобластей-слоев, в рамках которых рассматривалась система уравнений с отдельными физическими параметрами. На границах контакта подобластей задавались условия сопряжения, моделирующие физические процессы на этих границах.

Сложность модели и необходимость проводить долгосрочные прогнозы состояния грунтовых массивов вызывают необходимость использования высокопродуктивных вычислительных систем для решения таких задач.

2 Математическая постановка задачи и методы решения

Рассматриваемая система уравнений математической модели динамической консолидации имеет такой вид [2,3]:

$$k_{i1} \frac{\partial^2 u_{ck}^{(i)}}{\partial t^2} + k_{i2} \left(\frac{\partial u_{ck}^{(i)}}{\partial t} - \frac{\partial u_b^{(i)}}{\partial t} \right) - \left(\frac{\partial}{\partial x} \sigma_x^{(i)} + \frac{\partial}{\partial y} \tau_{xy}^{(i)} \right) - (1 - m_i) \frac{\partial}{\partial x} P^{(i)} = f_{i1}(x, y, t), \quad (1)$$

$$k_{i1} \frac{\partial^2 v_{ck}^{(i)}}{\partial t^2} + k_{i2} \left(\frac{\partial v_{ck}^{(i)}}{\partial t} - \frac{\partial v_b^{(i)}}{\partial t} \right) - \left(\frac{\partial}{\partial y} \sigma_y^{(i)} + \frac{\partial}{\partial x} \tau_{xy}^{(i)} \right) - (1 - m_i) \frac{\partial}{\partial y} P^{(i)} = f_{i2}(x, y, t), \quad (2)$$

$$m_i \rho_{i2} \frac{\partial^2 u_b^{(i)}}{\partial t^2} + k_{i2} \left(\frac{\partial u_b^{(i)}}{\partial t} - \frac{\partial u_{ck}^{(i)}}{\partial t} \right) - m_i \frac{\partial}{\partial x} P^{(i)} = f_{i3}(x, y, t), \quad (3)$$

$$m_i \rho_{i2} \frac{\partial^2 v_b^{(i)}}{\partial t^2} + k_{i2} \left(\frac{\partial v_b^{(i)}}{\partial t} - \frac{\partial v_{ck}^{(i)}}{\partial t} \right) - m_i \frac{\partial}{\partial y} P^{(i)} = f_{i4}(x, y, t), \quad (4)$$

где (для двух подобластей Ω_1 и Ω_2) $k_{i1} = \rho_{iG}(1 - m_i)$, $k_{i2} = \rho_{i2} g \frac{m_i^2}{K_{iD}}$, ρ_{i2} - плотность поровой жидкости, ρ_{iG} — плотность минеральных частиц, m_i — пористости грунта, K_{iD} — коэффициенты фильтрации, μ_i , λ_i — коэффициенты Ламе, g — ускорение свободного падения, M^B — модуль упругости жидкости, $P^{(i)} = \frac{M_b}{m_i} ((1 - m_i) (\frac{\partial u_{ck}^{(i)}}{\partial x} + \frac{\partial v_{ck}^{(i)}}{\partial y}) + m_i (\frac{\partial u_b^{(i)}}{\partial x} + \frac{\partial v_b^{(i)}}{\partial y}))$ - давление (при нулевых начальных условиях), $\vec{\sigma}^{(i)} = (\lambda_i (\frac{\partial u_{ck}^{(i)}}{\partial x} + \frac{\partial v_{ck}^{(i)}}{\partial y}) + 2\mu_i \frac{\partial u_{ck}^{(i)}}{\partial x}, \lambda_i (\frac{\partial u_{ck}^{(i)}}{\partial x} + \frac{\partial v_{ck}^{(i)}}{\partial y}) + 2\mu_i \frac{\partial v_{ck}^{(i)}}{\partial y}, \tau_{xy}^{(i)} = \mu (\frac{\partial u_{ck}^{(i)}}{\partial y} + \frac{\partial v_{ck}^{(i)}}{\partial x}))$ — составляющие тензора напряжений.

Искомые функции — составляющие векторов смещения скелета грунта и жидкости $w_{ck}^{(i)} = (u_{ck}^{(i)}, v_{ck}^{(i)})$, $w_b^{(i)} = (u_b^{(i)}, v_b^{(i)})$ определены на $(x, t) \in \Omega_{iT}$, $\Omega_{iT} = \Omega_i \times (0, T]$, $\Omega = \Omega_1 \cup \Omega_2$ — область моделирования, — её граница.

К уравнениям (1)-(4) ставятся нулевые начальные условия, условия первого и второго рода на внешних границах области моделирования, а также следующие условия сопряжения на границе контакта подобластей:

$$u_{ck}^{(1)} \Big|_{(x,y) \in \gamma^+} = u_{ck}^{(2)} \Big|_{(x,y) \in \gamma^-}, v_{ck}^{(1)} \Big|_{(x,y) \in \gamma^+} = v_{ck}^{(2)} \Big|_{(x,y) \in \gamma^-}, m_1 (u_b^{(1)} - u_{ck}^{(1)}) \Big|_{(x,y) \in \gamma^+} = m_2 (u_b^{(2)} - u_{ck}^{(2)}) \Big|_{(x,y) \in \gamma^-},$$

$$m_1 (v_b^{(1)} - v_{ck}^{(1)}) \Big|_{(x,y) \in \gamma^+} = m_2 (v_b^{(2)} - v_{ck}^{(2)}) \Big|_{(x,y) \in \gamma^-}, \quad (5)$$

$$(\sigma_n^{(1)} + P^{(1)}) \Big|_{(x,y) \in \gamma^+} = (\sigma_n^{(2)} + P^{(2)}) \Big|_{(x,y) \in \gamma^-}, \quad (6)$$

$$\tau_s^{(1)} \Big|_{(x,y) \in \gamma^+} = \tau_s^{(2)} \Big|_{(x,y) \in \gamma^-}, P^{(1)} \Big|_{(x,y) \in \gamma^+} - P^{(2)} \Big|_{(x,y) \in \gamma^-} = k_P m_1 (w_b^{(1)} - w_{ck}^{(1)}) \vec{n} \Big|_{(x,y) \in \gamma^+}.$$

Тут $\sigma_n^{(i)} = (A^{(i)} \cdot \vec{n}) \cdot \vec{n}$ — проекция на нормаль нормальной составляющие тензора напряжений $A^{(i)} = \begin{pmatrix} \sigma_x^{(i)} & \tau_{xy}^{(i)} \\ \tau_{xy}^{(i)} & \sigma_y^{(i)} \end{pmatrix}$, а $\tau_s^{(i)} = (A^{(i)} \cdot \vec{\tau}) \cdot \vec{n}$ — проекция на нормаль касательной составляющей тензора напряжений $A^{(i)}$, $\sigma_n^{(i)} = \sigma_x^{(i)} n_x^2 + \sigma_y^{(i)} n_y^2 + 2\tau_{xy}^{(i)} n_x n_y$, $\tau_s^{(i)} = (\sigma_x^{(i)} - \sigma_y^{(i)}) n_x n_y + \tau_{xy}^{(i)} (n_y^2 - n_x^2)$; $\gamma = \{(x, y) : x = 0\}$ — граница контакта подобластей, $\gamma^+ = \gamma \cap \Omega_1$, $\gamma^- = \gamma \cap \Omega_2$, \vec{n} - нормаль к γ^+ .

Для численного моделирования консолидационных процессов, модель (1)-(6) дискретизировалась методом конечных элементов с использованием в точках, лежащих на границе контакта подобластей специальных базисных функций, удовлетворяющих условиям (5) [2,3]. Результирующая разреженная система линейных алгебраических уравнений (СЛАУ) представлялась, как произведение левой и правой частей «базовой» СЛАУ, сгенерированной без учёта условий сопряжения, на матрицу, описывающую линейное комбинирование строк СЛАУ, соответствующих точкам дискретизации на границе контакта подобластей.

В целом, нахождение решения задачи на каждом следующем шаге по времени состоит из двух этапов - нахождение коэффициентов СЛАУ и её решение. Поскольку изначальная система дифференциальных уравнений - нелинейная, коэффициенты СЛАУ изменяются на каждом шаге по времени и требуют пересчёта. Это, а также сложность системы (1)-(4), приводит к тому, что время, тратящееся на нахождение коэффициентов СЛАУ сопоставимо со временем, которое тратится на её решение, что обуславливает требование к параллельным алгоритмам распараллеливать как решение СЛАУ, так и нахождение её коэффициентов.

3 Параллельные алгоритмы

На кластерных системах, распараллеливание нахождения коэффициентов СЛАУ не требует обменов, и его эффективность зависит только от равномерности загрузки процессоров. При решении разреженной СЛАУ итерационными методами обмены требуются, и их объем зависит от распределения матрицы СЛАУ по системе. В связи с этим, а также с тем, что при решении рассматриваемой задачи методом конечных элементов, результирующая СЛАУ заполнена существенно неоднородно, возникает оптимизационная задача выбора оптимального по критерию быстродействия распределения данных.

Разработанный параллельный алгоритм учитывает эти особенности следующим образом:

— для решения СЛАУ применяется параллельный алгоритм BiCGStab [2] с блочно-строчным распределением матрицы, хранимой в сжато-строчном (CSR) формате, который модифицируется с учётом представления матрицы СЛАУ в виде произведения «базовой» матрицы и матрицы, отвечающей за условия сопряжения, в том числе добавлением операций обмена данными;

— для минимизации обменов данными используется перенумерование строк и столбцов матрицы по алгоритму Катхила-МакКи, а для оптимизации суммарной вычислительной эффективности используется эвристический алгоритм, базирующийся на теоретических оценках времени работы алгоритмов с коэффициентами, вычисляющимися для конкретной системы в процессе работы.

Вычисления, проводящиеся каждым процессом, могут быть дополнительно распараллелены используя многопоточность, или графический процессор (GPU). Во втором случае, особенности вычислений на GPU и решаемой задачи были учтены следующим образом:

— матрица СЛАУ для вычислений на GPU конвертируется в блочный сжато-строчный (BCSR) формат, использование которого улучшает характеристики алгоритмов решения разреженных СЛАУ в случае большого коэффициента заполненности [3], что наблюдается при решении рассматриваемой задачи;

— при вычислении коэффициентов СЛАУ, каждый блок потоков обрабатывает один конечный элемент, а каждый поток – одну из вершин этого элемента; граф связности элементов раскрашивается, и вычисления проводятся последовательно по каждому цвету, что необходимо для обеспечения независимости вычислений и дает возможность использовать быструю локальную память для доступа к данным, общим для каждого конечного элемента.

4 Тестирование алгоритмов

Тестирование разработанного алгоритмического и программного обеспечения проводилось на модельном примере на области, дискретизированной конечными элементами с 18236 вершинами сетки. Размер матрицы СЛАУ составил 510608×510608 с количеством ненулевых элементов, равным 7416976. Тестировались три варианта организации вычислений: однопоточный режим (в рамках каждого процесса работает лишь один поток), многопоточный режим (четыре потока на процесс) и режим с использованием GPU. Время решения задачи на одном шаге и ускорение алгоритмов представлены на рис.1 и 2 соответственно.

Разработанные алгоритмы тестировались на кластерах СКИТ-3 и СКИТ-4 Института кибернетики НАН Украины, для вычислений на GPU использовалась технология OpenCL, а в качестве библиотеки обмена сообщениями – OpenMPI.

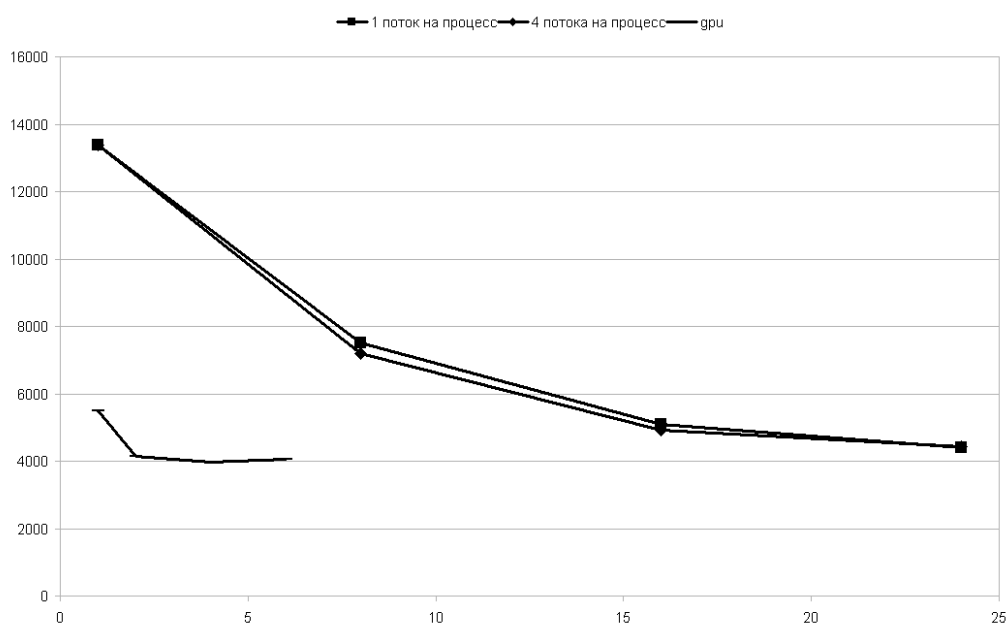


Рис. 1. Время работы алгоритмов

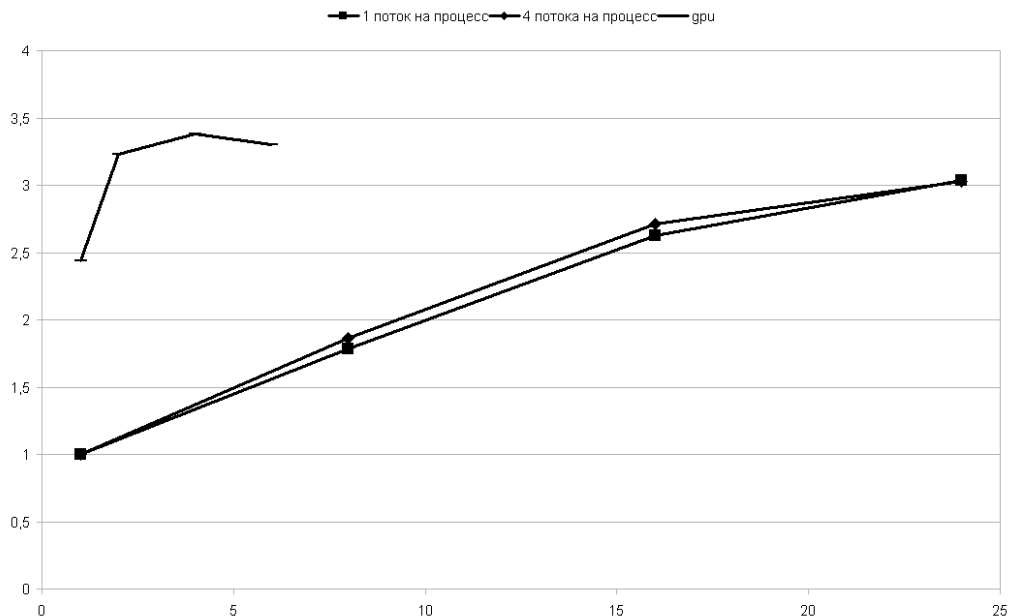


Рис. 2. Ускорение алгоритмов

5 Заключение

Полученные результаты тестирования показывают, что использование многопоточной схемы вычислений не приносит существенного ускорения, тогда как вычисления на GPU дают большее ускорение работы, чем алгоритмы не использующие GPU. Масштабируемость всех разработанных алгоритмов невелика, что объясняется большим количеством обменов при решении СЛАУ, возникающих из-за особенностей учёта условий сопряжения в использованном варианте метода конечных элементов. Следует также отметить существенное влияние на производительность алгоритмов оптимизации загрузки процессоров, в направлении усовершенствования которых планируется проводить дальнейшие исследования.

Список литературы

- [1] Зарецкий Ю.К., Ломбардо В.Н. Статика и динамика грунтовых плотин. – Москва: Энергоатомиздат, 1983. – 256 с.
- [2] Скопецкий В. В., Марченко О. А., Самойленко Т. А. Приближенное решение нелинейной системы уравнений для двухфазных сред // Кибернетика и системный анализ. – 2008. – №4. – С. 73-88.
- [3] Скопецкий В.В., Марченко О.А., Самойленко Т.А. Построение дискретного приближенного решения нелинейной системы динамики двухфазных сред // Кибернетика и системный анализ. – 2009. – № 4. – С.69- 80.
- [4] Yousef Saad Iterative methods for sparse linear systems, 2 edition. - Society for Industrial and Applied Mathematics, 2003. – 528 p.
- [5] Abhijeet Gaikwad, Ioane Muni Toke GPU based Sparse Grid Technique for Solving Multidimensional Options Pricing PDEs // Supercomputing'09, Proceedings of the 2nd Workshop on High-Performance Computational Finance - SC09-WHPCF09, ACM Press, 2009, Article No. 6