

УДК 519.63

doi 10.26089/NumMet.v19r444

ПАРАЛЛЕЛЬНЫЕ ТЕХНОЛОГИИ И СЕТОЧНЫЕ СТРУКТУРЫ ДАННЫХ ДЛЯ РЕШЕНИЯ ТРЕХМЕРНЫХ КРАЕВЫХ ЗАДАЧ В СЛОЖНЫХ ОБЛАСТЯХ НА КВАЗИСТРУКТУРИРОВАННЫХ СЕТКАХ

В. Д. Корнеев¹, В. М. Свешников²

При распараллеливании решения трехмерных краевых задач, особенно в областях со сложной геометрией, важными являются технологии проведения вычислений и структуры данных. От них зависит объем хранимой информации и время решения. В статье предлагаются технологии распараллеливания метода декомпозиции расчетной области на подобласти, сопрягаемые без наложения, на квазиструктурированных сетках. Разработаны параллельные сеточные структуры данных, ориентированные преимущественно на работу со структурированными массивами данных. Приведен иллюстративный пример, показывающий основные положения предлагаемого подхода.

Ключевые слова: краевые задачи, квазиструктурированные сетки, технологии распараллеливания, структуры данных, структурированные массивы.

1. Введение. Предлагаемые ниже разработки излагаются на примере решения трехмерных краевых задач для уравнения Пуассона, которое возникает во многих физических приложениях. Одно из них — расчет потенциала электрического поля, создаваемого многоэлектродными электронно-оптическими системами, в задачах сильноточной электроники. К ним относятся многие практически важные приборы, служащие для получения СВЧ-колебаний, электронно-лучевой сварки, ускорения заряженных частиц и для других практически важных целей [1]. Математическое моделирование таких приборов приводит к решению нелинейной самосогласованной задачи по расчету движения пучков заряженных частиц во внешних, т.е. создаваемых электродами прибора, и собственных, т.е. создаваемых самим пучком, электрических и магнитных полях [2–4]. Одной из подзадач самосогласованной задачи является расчет потенциала электрического поля, который проводится на каждой итерации по нелинейности. В свою очередь, самосогласованная задача решается на каждом шаге по оптимизации конструкции прибора. Для получения адекватной точности необходима сетка, содержащая $10^3 \times 10^3 \times 10^3 = 10^9$ узлов, расчет потенциала на которой занимает часы на персональном компьютере, а решение самосогласованной задачи в сложной расчетной области с мощным пучком заряженных частиц — сутки работы персонального компьютера. В этой связи трехмерное моделирование современных электронно-оптических приборов необходимо проводить на многопроцессорных суперЭВМ. Для этого необходима разработка параллельных алгоритмов, технологий и структур данных, что и обсуждается в настоящей статье.

Решение трехмерных краевых задач в областях со сложной геометрией обычно проводится на неструктурированных адаптивных сетках, для построения которых разработаны сеточные генераторы, например один из наиболее известных NetGen [5]. Недостатком неструктурированных сеток является, во-первых, необходимость хранения большого объема данных, во-вторых, сложность доступа к ним, что увеличивает время работы численных алгоритмов, и, в-третьих, сложность распараллеливания при работе на многопроцессорных суперЭВМ. Наиболее экономичными представляются структурированные равномерные параллелепипедальные сетки, но они требуют введения неоправданно большого количества узлов, которые никак не увеличивают точности расчетов, а лишь ухудшают сходимость численных алгоритмов.

Компромиссным вариантом являются предлагаемые авторами квазиструктурированные сетки, составленные из структурированных параллелепипедальных подсеток. Их суть состоит в том, что расчетная область разбивается структурированной макросеткой на подобласти, в каждой из которых строится своя структурированная подсетка. Плотность узлов в подсетках регулируется на основе физических соображений. Для упомянутой выше самосогласованной задачи в подобластях, через которые проходит пучок,

¹ Институт вычислительной математики и математической геофизики СО РАН, просп. Лаврентьева, д. 6, 630090, Новосибирск; ст. науч. сотр., e-mail: korneev@ssd.sccc.ru

² Институт вычислительной математики и математической геофизики СО РАН, просп. Лаврентьева, д. 6, 630090, Новосибирск; главный науч. сотр., e-mail: victor@lapasrv.sccc.ru

строятся густые подсетки, которые разрежаются к периферии. Отметим, что предлагаемые квазиструктурированные сетки позволяют эффективно решать задачи сильноточной электроники, во-первых, за счет адаптации к пучку и, во-вторых, за счет простой работы с сеточными элементами.

Основным инструментом распараллеливания решения краевых задач является метод декомпозиции расчетной области, изложенный в работах многих авторов (см., например, монографии [6–8] и цитируемую там литературу). Наш подход основан на прямой (непосредственной) аппроксимации уравнения Пуанкаре–Стеклова на границе сопряжения подобластей (интерфейсе) системой линейных алгебраических уравнений (СЛАУ). Такой подход допускает несогласованные подсетки, не требует построения матрицы сеточных уравнений во всей области, а лишь в подобластях, и ориентирован на решение СЛАУ на интерфейсе быстроходящимися итерационными методами в подпространствах Крылова, которые в нашем случае допускают простую реализацию.

Распараллеливание решения краевых задач на квазиструктурированных сетках для областей, составленных из параллелепипедов, доведенное до исследования эффективности, рассматривалось в работах авторов [9, 10]. Решение трехмерных краевых задач в областях со сложной геометрией потребовало разработки специальных параллельных технологий и структур данных, так как от этого зависит эффективность распараллеливания. Именно этим вопросам посвящена настоящая статья. Основной принцип разработок состоит в сохранении, по возможности, структурированных массивов информации. Он основан на компромиссном решении между объемом данных и удобством доступа к ним. Впервые данный подход для случая структурированных сеток был рассмотрен в работе [11]. В настоящей работе он развит для квазиструктурированных сеток и параллельных технологий. При невозможности введения структурированных массивов, что имеет место, например, при хранении значений искомой функции в узлах квазиструктурированной сетки, вводятся одномерные вещественные массивы и опорные адреса подсеток в них. Такой прием позволяет работать с подсеткой как со структурированным массивом в отдельной подпрограмме. Отметим, что для двумерных краевых задач квазиструктурированные сетки с локальной модификацией приграничных узлов рассматривались в работе [12], а распараллеливание решения двумерных задач в работе [13].

Раздел 2 настоящей статьи посвящен изложению технологий распараллеливания, раздел 3 — изложению структуры данных, раздел 4 содержит иллюстративный пример, показывающий основные положения нашего подхода. В заключении приводятся выводы по проделанной работе.

2. Технологии распараллеливания. Пусть требуется найти решение $u(r)$ 3D краевой задачи

$$\Delta u(r) = q(r), \quad r \in G, \tag{1}$$

с граничными условиями

$$lu(r) = g(r), \quad r \in \Gamma. \tag{2}$$

Здесь Δ, l — соответственно оператор Лапласа и оператор граничных условий, G — расчетная область, $r = (x, y, z)$ — декартовы координаты, Γ — граница области ($\bar{G} = G \cup \Gamma$). Предполагается, что условия (2) — это условия Дирихле или Неймана, причем последние заданы на плоскостях, параллельных координатным плоскостям.

Опишем вокруг \bar{G} параллелепипед \bar{Q} , такой, что $\bar{G} \in \bar{Q}$, в котором построим равномерную макросетку

$$\bar{\Phi} = \{x_i = iH_x, y_j = jH_y, z_k = kH_z, i = \overline{0, N_x}, j = \overline{0, N_y}, k = \overline{0, N_z}\}.$$

Здесь N_x, N_y, N_z — число интервалов $\bar{\Phi}$ по координатным осям, $H_x = D_x/N_x, H_y = D_y/N_y, H_z = D_z/N_z$, где D_x, D_y, D_z — размеры параллелепипеда \bar{Q} . Будем различать следующие типы подобластей:

- а) внешние при $G_{I,J,K} \cap G = \emptyset$,
- б) граничные при $G_{I,J,K} \cap G \neq \emptyset$ и $G_{I,J,K} \cap \Gamma \neq \emptyset$,
- в) внутренние при $G_{I,J,K} \in G$.

В каждой подобласти построим свою равномерную параллелепипедальную подсетку $\bar{\Omega}_{I,J,K}$ с числом интервалов по координатным осям n_x, n_y, n_z (индексы I, J, K здесь опущены), причем будем предполагать, что эти числа суть 2 в целой степени.

Во внешних подобластях $n_x = n_y = n_z = 0$, т.е. во внешних подобластях подсетки не строятся и они вводятся лишь для поддержки структурированности массивов, описывающих макросетку.

Ниже для сокращения записи узел, имеющий индексы i, j, k , будем обозначать как (i, j, k) . Каждая подсетка $\bar{\Omega}_{I,J,K}$ содержит узлы (i, j, k) разных типов. Назовем узел (i', j', k') соседним к узлу (i, j, k) , если $|i' - i| + |j' - j| + |k' - k| = 1$. Определим следующие типы узлов:

- а) внешний при $(i, j, k) \notin G$,

- б) граничный при $(i, j, k) \in \Gamma$,
 в) приграничный при $(i, j, k) \in G$ и соседний (i', j', k') есть граничный или внешний,
 г) внутренний при $(i, j, k) \in G$ и соседний (i', j', k') есть внутренний или приграничный.

Граничные узлы, в свою очередь, разделяются на два типа:

- 1) узлы, лежащие на отрезках границы с условиями Дирихле,
- 2) узлы, лежащие на отрезках границы с условием Неймана.

Граница γ сопряжения подобластей (интерфейс) состоит из граней f , ребер e и макроузлов m ($\gamma = f \cup e \cup m$).

На интерфейсе вводится сетка $\omega = \omega_f \cup \omega_e \cup \omega_m$, где ω_f — сетка на гранях, ω_e — сетка на ребрах, ω_m — сетка, состоящая из макроузлов макросетки $\bar{\Phi}$.

Введем функцию v , которая является следом u на γ ($v = u|_{\gamma}$). Представим ее в виде

$$v = \tilde{v} + \bar{v}, \quad \text{где } \tilde{v} = u|_f, \quad \bar{v} = u|_{e \cup m}.$$

Определим оператор Пуанкаре–Стеклова S как

$$S\tilde{v} \equiv \left(\frac{\partial u(\tilde{v})}{\partial \mathbf{n}} \right)_f^+ + \left(\frac{\partial u(\tilde{v})}{\partial \mathbf{n}} \right)_f^-. \quad (3)$$

Здесь \mathbf{n} — внутренняя нормаль к f , знаки $+$, $-$ указывают на принадлежность объекта смежным подобластям.

Алгоритм решения задачи (1), (2) методом декомпозиции расчетной области, применяемый нами, основан на работах авторов [9, 10, 14] для случая области, составленной из параллелепипедов. Поэтому здесь будут рассмотрены только необходимые для изложения детали. Суть алгоритма заключается в следующем.

Исходную краевую задачу переформулируем следующим образом: в замкнутой области \bar{G} требуется найти решение уравнения Пуанкаре–Стеклова

$$S\tilde{v}(r) = 0, \quad r \in f, \quad (4)$$

совместно с решением краевой задачи

$$\Delta u(r) = q(r), \quad r \in G \setminus f, \quad (5)$$

$$lu(r) = g(r), \quad r \in \Gamma, \quad (6)$$

$$u(r) = v(r), \quad r \in \gamma. \quad (7)$$

Задача (5)–(7) разбивается на подзадачи на ребрах и в макроузлах

$$\Delta u(\bar{v}(r)) = q(r), \quad r \in e \cup m, \quad lu(\bar{v}(r)) = g(r), \quad r \in \Gamma, \quad (8)$$

подзадачи во внутренних подобластях

$$\Delta u(r) = q(r), \quad u(r) = v(r), \quad r \in \gamma, \quad (9)$$

и подзадачи в граничных подобластях

$$\Delta u(r) = q(r), \quad u(r) = v(r), \quad r \in \gamma, \quad lu(r) = g, \quad r \in \Gamma. \quad (10)$$

Данные задачи аппроксимируются на квазиструктурированной сетке. Система сеточных уравнений строится и решается во внутренних, приграничных и граничных узлах 2-го типа. В связи с этим назовем совокупность таких узлов счетными узлами.

Во внутренних узлах внутренних и граничных подобластей исходное уравнение (1) аппроксимируется при помощи семиточечной схемы вида

$$\begin{aligned} & \frac{1}{h_x^2} (u_{i-1,j,k} + u_{i+1,j,k} - 2u_{i,j,k}) + \\ & + \frac{1}{h_y^2} (u_{i,j-1,k} + u_{i,j+1,k} - 2u_{i,j,k}) + \frac{1}{h_z^2} (u_{i,j,k-1} + u_{i,j,k+1} - 2u_{i,j,k}) = q_{i,j,k}. \end{aligned} \quad (11)$$

Здесь h_x, h_y, h_z шаги подсетки по координатным осям, а переменные с индексами — соответствующие приближенные величины.

В граничных узлах 1-го типа хранятся значения функции g — правой части условий (2), а граничные узлы 2-го типа входят в аппроксимации исходного уравнения и граничных условий, т.е. в них пересчитываются значения искомой функции.

В приграничных узлах проводится специальная аппроксимация, учитывающая криволинейность границы и неравномерность сеточных шаблонов. В данную аппроксимацию входят внутренние узлы, и таким образом приграничные уравнения связываются с уравнениями (11). Полученная система уравнений решается итерационными методами. Отметим, что во внутренних подобластях используется метод Писмана–Рэчфорда, который допускает параллельную реализацию на графических ускорителях [10].

Во внешних узлах расчеты не проводятся. Они необходимы лишь для поддержки структурированности массивов.

Значения функции \bar{v} на ребрах и в макроузлах находятся из аппроксимации исходной краевой задачи в узлах сетки $\omega_e \cup \omega_m$, которая строится с привлечением ближайших узлов сетки ω_f после проведения итерационного процесса по подобластям, поэтому функция \tilde{v} предполагается известной. СЛАУ, полученная в результате аппроксимации, решается при помощи двухуровневого итерационного процесса, который состоит в следующем. Предполагая известными значения функции в макроузлах ω_m , находим значения на ребрах ω_e при помощи прогонок, которые могут выполняться параллельно. Затем из аппроксимации в макроузлах ω_m , построенной с привлечением ближайших узлов сетки ω_e , пересчитываются значения в макроузлах, и так далее продолжается до сходимости данного итерационного процесса. Решение уравнения Пуанкаре–Стеклова проводится по следующему алгоритму.

Заменим в (3) производные разностными соотношениями. Тогда оператор S заменится его приближенным аналогом S_h , определяемым по формуле

$$S_h \tilde{v}_{f,h} \equiv (d_h u_h)^+ + (d_h u_h)^-, \tag{12}$$

где d_h — оператор численного дифференцирования, $u_h, v_{f,h}$ — приближенные значения функций, $u_h = u_h(v_{f,h})$. Так как S_h — линейный оператор, действующий в конечномерном пространстве, то его можно представить в матрично-векторном виде

$$S_h \tilde{v}_{f,h} \equiv A \tilde{v}_{f,h} + b, \tag{13}$$

где A — квадратная матрица, $a, b, v_{f,h}$ — векторы. Потребуем, чтобы разность приближенных производных в узлах сетки обращалась в нуль, что, как следует из (13), дает систему линейных алгебраических уравнений

$$A \tilde{v}_{f,h} + b = 0. \tag{14}$$

Из (12), (13) следует, что вектор b можно вычислить, положив $v_{f,h} = 0$. Вычислять элементы матрицы A нет необходимости, так как для решения уравнения (14) будем применять итерационные методы в подпространствах Крылова [15], которые в общем виде можно записать как

$$\tilde{v}_{f,h}^{k+1} = \Lambda(\tilde{v}_{f,h}^k, A p^k), \tag{15}$$

где $k = 0, 1, \dots$ — номер итерации, Λ — функция, определяющая конкретный метод, например метод сопряженных градиентов, метод сопряженных невязок или метод обобщенных минимальных невязок (GMRES: Generalized Minimal Residual Method), p — вспомогательный вектор, определяемый выбором конкретного метода решения. Итерационные методы (15) не требуют знания элементов матрицы A , а требуют знания лишь результата действия $A p$ матрицы на вектор. В нашем подходе, как это следует из (12) и (13), произведение матрицы на вектор можно вычислить просто как разность приближенных производных по разные стороны интерфейса.

Решение исходной краевой задачи проводится методом итераций по подобластям, состоящим из следующих этапов.

1. Задается начальное приближение $\tilde{v}_f^{(0)}$ на гранях f .
2. Из задачи (8) находятся значения функции $\bar{v}_{e,m}^{(k)}$ ($k = 0, 1, \dots$ — номер итерации) на ребрах и в макроузлах.
3. Полагая $v = v^{(k)} = \tilde{v}_f^{(k)} + \bar{v}_{e,m}^{(k)}$, находятся значения искомой функции на k -м приближении в подобластях.

4. Рассчитываются производные, входящие в выражение (3).
5. Делается очередной $(k + 1)$ -й шаг по решению уравнения Пуанкаре–Стеклова (4) и находятся значения функции $\tilde{v}_f^{(k+1)}$ на гранях.
6. Если сходимость итерационного процесса достигнута, то находится окончательное решение на ребрах, в макроузлах и в подобластях, если же это не так, то повторяются шаги 2–5.

На каждой итерации по подобластям решаются краевые подзадачи в подобластях на сетках $\bar{\Omega}_{I,J,K}$, что составляет наибольшую часть времени решения всей задачи, поэтому подзадачи решаются параллельно. Обычно применяемое отображение “одна подобласть–один процессор” неэффективно, так как подсетки могут содержать различное количество счетных узлов, что приведет к разбалансировке загрузки процессоров (под процессором мы понимаем вычислительное ядро суперкомпьютера). Поэтому подсетки группируются в объединения, содержащие приблизительно одинаковое число узлов. Устанавливается отображение “одно объединение–один процессор”.

Передача информации между подобластями одного объединения обходится без межпроцессорных обменов, а между подобластями различных объединений требуется их проведение. Инициализация и осуществление межпроцессорных обменов являются самыми медленными операциями процесса расчета. В связи с этим в основу применяемой нами технологии MPI-распараллеливания кладутся асинхронные операции, позволяющие хотя бы частично проводить обмены на фоне выполнения необходимых арифметических и логических операций. Для проведения расчетов по приведенным выше алгоритмам необходима следующая информация:

- 1) решение краевых подзадач (9) во внутренних подобластях — вещественный массив значений искомой функции в узлах подсеток и числа h_x, h_y, h_z для проведения вычислений по формуле (11);
- 2) решение краевых подзадач (10) в граничных подобластях — вещественный массив значений искомой функции и целый массив типов узлов в узлах подсеток, числа h_x, h_y, h_z для проведения вычислений по формуле (11) во внутренних узлах и вещественный массив коэффициентов сеточных уравнений в приграничных узлах и в граничных узлах 2-го типа;
- 3) решение краевых подзадач (8) на ребрах и в макроузлах — вещественный массив значений искомой функции и целый массив типов узлов, вещественный массив коэффициентов сеточных уравнений в узлах подсеток на ребрах и в макроузлах;
- 4) решение уравнения Пуанкаре–Стеклова — вещественный массив значений искомой функции, вещественный массив внутренних нормальных производных и целый массив типов узлов в узлах подсеток на гранях интерфейса.

Кроме того, вводятся вспомогательные массивы для удобства проведения расчетов. Более подробное изложение хранимых массивов, а также их распределение по процессорам приведено ниже.

3. Сеточная структура данных. При формировании структуры данных мы будем стремиться сохранить структурированность массивов. Причины этому следующие. Во-первых, неструктурированные массивы, несмотря на кажущуюся экономичность, значительно усложняют работу с сеточной информацией. Так, например, в часто возникающей при реализации численных алгоритмов задаче выбора ближайших к заданной точке сеточных узлов или элементов требуется трудоемкий анализ информации, в то время как в структурированных массивах эта задача решается при помощи простых соотношений. Во-вторых, упрощается выборка значений в текущем узле путем просто указаний его индекса. В-третьих, упрощается разработка (написание и отладка) программ, реализующих численные алгоритмы.

Там, где невозможно введение структурированных массивов, например при хранении значений искомой функции на квазиструктурированной сетке, мы поступаем следующим образом. Вводим дополнительный целый структурированный массив-указатель опорных адресов подмассивов в подсетках. Расчеты на подсетках выполняются отдельной подпрограммой, на вход которой в качестве параметра поступает адрес рассматриваемой подсетки и информация о ней (число узлов по каждой координате), что дает возможность в данной подпрограмме работать с подсеткой как со структурированным массивом.

Сеточная структура данных состоит из следующих разделов:

- 1) макросетка и подсетки в подобластях,
- 2) объединения подсеток,

3) сетка на интерфейсе, которая включает в себя подсетки на гранях, подсетки на ребрах и макроузлы.

Пользователем на основе физических представлений о конкретной решаемой задаче задается макро-сетка $\bar{\Phi}$ и подсетки $\bar{\Omega}_{I,J,K}$, что составляет основу квазиструктурированной сетки. Подсетки на ребрах и гранях интерфейса строятся с учетом заданной информации. Они состоят из совпадающих узлов подсеток на границах смежных подобластей (напомним, что число интервалов в подсетках есть 2 в целой степени). Определение типов подобластей, типов узлов в подсетках и построение вспомогательных массивов проводится в программе на основе заданной информации о квазиструктурированной сетке и информации о геометрии расчетной области.

Информация о макросетке рассылается на все процессоры. Информация о подсетках подвергается распараллеливанию так, что на p -м процессоре хранится только та информация, которая входит в p -е объединение. Кроме того, аналогичным образом распараллеливается информация о сетке на интерфейсе, т.е. на p -м процессоре хранится только информация о подсетках на гранях и ребрах интерфейса данного объединения.

При проведении итераций по подобластям необходимо вычислять суммы производных, входящих в выражение (3). Для этого помимо производных в текущей подобласти необходимы значения производных в смежных подобластях, что порождает операции пересылок между процессорами. Кроме того, пересылки необходимы при аппроксимации исходной краевой задачи в узлах сетки $\omega_e \cup \omega_m$ для нахождения значений искомой функции на ребрах и в макроузлах.

Ниже приведено описание массивов, составляющих структуру данных. При этом применяется фор-трановская форма записи. Например, запись

$$AA(N_x, N_y, N_z)$$

означает

$$DIMENSION AA(N_x, N_y, N_z),$$

а указание имени массива с индексами в операторе присваивания

$$a = AA(I, J, K)$$

означает выборку элемента массива.

3.1. Макросетка и подсетки. Массивы информации о макросетке и подсетках формируются до проведения итерационного процесса по подобластям. Они рассылаются на все процессоры, участвующие в вычислениях. Перечислим целые структурированные массивы.

1. $T_{\Phi}(N_x, N_y, N_z)$ — типы подобластей: внешняя, граничная, внутренняя.
2. $P_{\Phi}(N_x, N_y, N_z)$ — номера процессоров, отвечающих за подобласти.
3. $N_{\Omega}(3, N_x, N_y, N_z)$ — число интервалов подсеток по координатным осям. Здесь для подсетки $\bar{\Omega}_{I,J,K}$ число интервалов по координатным осям выбирается как $n_x = N_{\Omega}(1, I, J, K)$, $n_y = N_{\Omega}(2, I, J, K)$, $n_z = N_{\Omega}(3, I, J, K)$. Кроме того, вводится одномерный вещественный массив коэффициентов сеточных уравнений в приграничных узлах и в граничных узлах 2-го типа.
4. $C_M(N_c)$, где N_c — длина массива. Доступ к элементам данного массива осуществляется через опорные адреса, указанные в массиве типов узлов (см. ниже).

3.2. Объединения. Информация об объединениях подобластей (подсеток) хранится в одномерных целых и вещественных массивах.

1. $M_S(N_P)$ — число подобластей в объединениях.
2. $N_S(3N_xN_yN_z)$ — номера (I, J, K) подобластей в объединениях.
3. $A_S(N_P)$ — опорные адреса в массиве N_S . Массивы, перечисленные в подпунктах 1–3, рассылаются на все процессоры. Это связано с тем, что при расчете одного объединения необходима информация о подобластях из соседних объединений. Далее перечислим массивы информации об одном p -м объединении, хранящиеся на одном p -м процессоре.
4. $U_S(N_M)$ — значения искомой функции в узлах подсеток, входящих в объединение (N_M — суммарное число узлов в подсетках). Подсетки располагаются в порядке следования их номеров в массиве N_S .

5. $T_S(N_M)$ — типы узлов подсеток, входящих в объединение. Заметим, что признаком приграничного узла является наличие его опорного адреса в массиве коэффициентов C_M .
6. $A_S(M_s(p))$ — опорные адреса в массивах U_S, T_S .

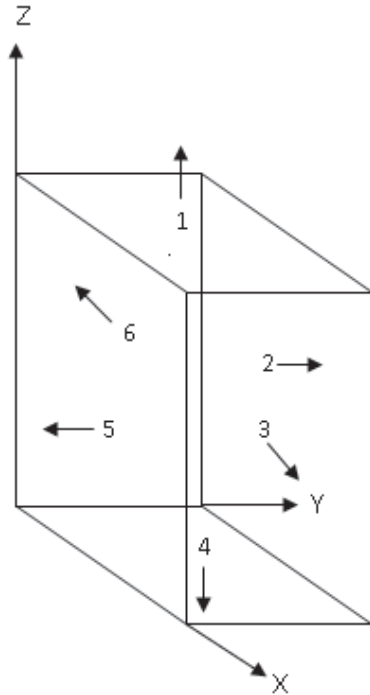


Рис. 1. Нумерация граней

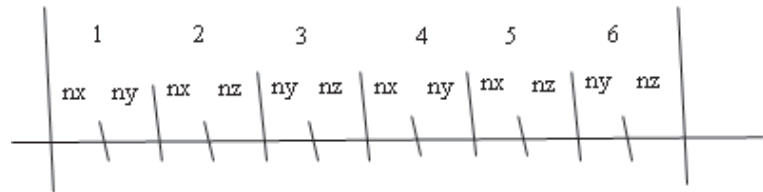


Рис. 2. Размеры подсеток на гранях

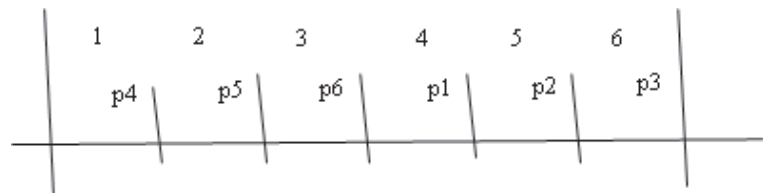


Рис. 3. Номера смежных процессоров

3.3. Интерфейс. Интерфейс состоит из граней, ребер и макроузлов, на которых строятся соответственно сетки ω_f, ω_e и ω_m .

3.3.1. Грани. Грани интерфейса (имеется в виду сетка ω_f) хранятся только для подобластей, входящих в объединение на одном процессоре, в порядке следования подобластей в рассматриваемом объединении. Нумерация граней одной подобласти показана на рис. 1. Подсетки на гранях, входящие в сетку ω_f , могут иметь разное число узлов, поэтому доступ к информации в них осуществляется через опорные адреса. Для хранения информации о гранях интерфейса p -го объединения отводятся следующие массивы.

1. $U_f(N_f)$ — значения искомой функции в узлах подсеток.
2. $T_f(N_f)$ — типы узлов в подсетках.
3. $A_f(6M_s(p))$ — опорные адреса в массивах U_f, T_f . Здесь N_f — длина массива. Данная информация хранится для каждой из 6 граней подобласти. Поясним: опорные адреса хранятся группами по 6 чисел для каждой подобласти в порядке следования подобластей. Например, выборка значений функции для подсетки 3-й грани i -й подобласти осуществляется как $U_f(A_f(6(i-1) + 3))$.
4. $R_f(12M_s(p))$ — число интервалов в подсетках на гранях. Структура массива R_f для одной подобласти показана на рис. 2 и аналогична структуре массива A_f (числа вверху означают номера граней). Например, выборка размеров подсетки 3-й грани i -й подобласти осуществляется как $n_x = R_f(12(i-1) + 5), n_y = R_f(12(i-1) + 6)$.

Рассматриваемая грань f_t может быть внешней при $f_t \cap \bar{G} = 0$ или граничной при $f_t \cap G = 0$ и $f_t \cap \Gamma \neq 0$. И в том и другом случае f_t не входит в интерфейс. Тогда в массиве опорных адресов A_f в соответствующем месте устанавливается специальный признак, например, -1 .

Для реализации пересылок значениями в узлах смежных подсеток, лежащих в различных объединениях, хранится массив Q_f целых чисел (тегов), однозначно идентифицирующих грани. Его структура аналогична структуре массива A_f .

5. $Q_f(6M_s(p))$ — теги граней. Кроме того, для каждой грани вводится массив P_f , в котором хранятся номера процессоров, отвечающих за смежные для данной грани подсетки.
6. $P_f(6M_s(p))$ — номера процессоров в смежных подсетках. Его структура аналогична структуре массива A_f и показана на рис. 3. Здесь p_i ($i = \overline{1,6}$) — номер процессора, отвечающего за смежную для рассматриваемой грани подобласть.

3.3.2. Ребра. Ребра интерфейса (имеется в виду сетка ω_e) хранятся только для подобластей, входящих в объединение на одном процессоре, в порядке следования подобластей в рассматриваемом объединении. Нумерация ребер одной подобласти показана на рис. 4.

Сетки ω_e составляют подсетки на ребрах. Для хранения информации о ребрах интерфейса p -го объединения отводятся следующие массивы: $U_e(N_e)$, $T_e(N_e)$, $A_e(12M_s(p))$, $R_e(12M_s(p))$, смысл и назначение которых аналогичны соответствующим массивам для граней.

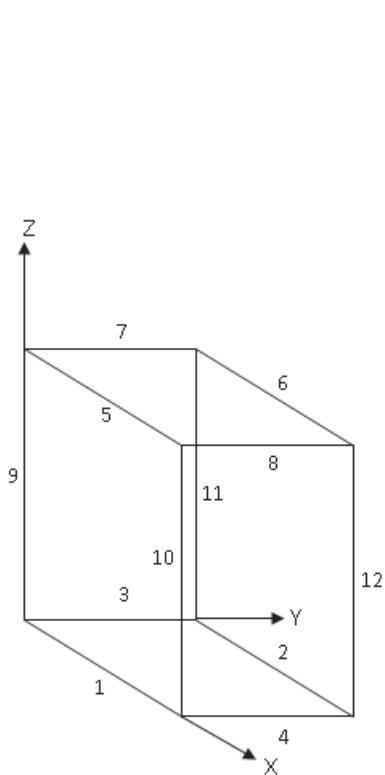


Рис. 4. Нумерация ребер подобласти

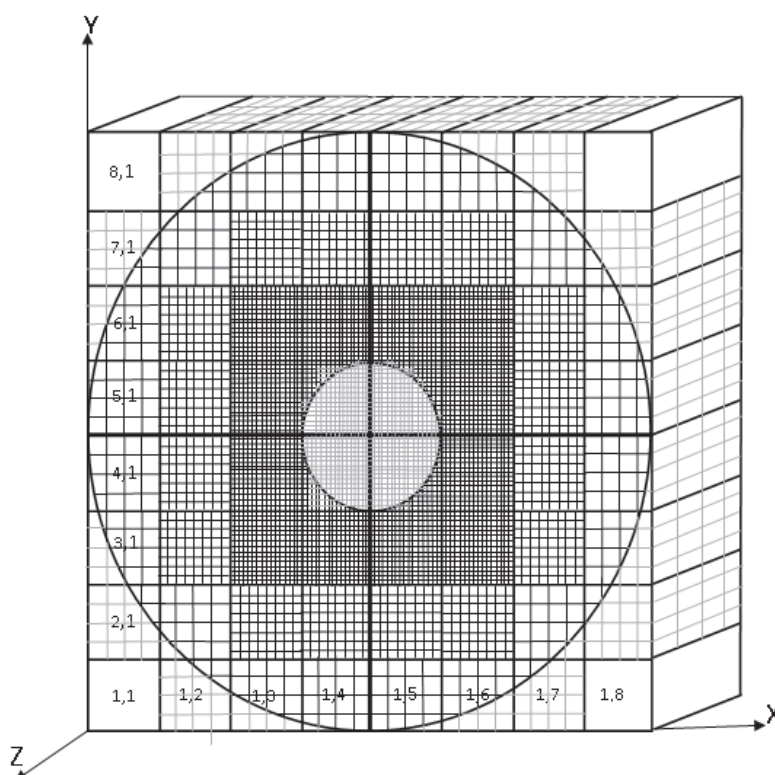


Рис. 5. Расчетная область и квазиструктурированная сетка

3.3.3. Макроузлы. Макроузлы интерфейса составляют сетку ω_m , которая совпадает с макросеткой $\bar{\Phi}$. На p -м процессоре хранится информация о макроузлах подобластей, входящих в p -е объединение. Макроузлы одной подобласти нумеруются цифрами от 1 до 8.

Доступ к макроузлам обеспечивается через следующие структурированные массивы.

1. $U_m(M_s(p), 8)$ — значения функции в макроузлах.
2. $T_m(M_s(p), 8)$ — типы макроузлов: внешний, внутренний, граничный.

Для каждого макроузла до проведения итераций рассчитывается и хранится набор коэффициентов сеточного уравнения, построенного на семиточечном шаблоне вида (11), но с неравными шагами [16]. Для хранения коэффициентов отводится структурированный массив.

3. $C_m(M_s(p), 8, 7)$ — коэффициенты в макроузлах. где $M_s(p)$ — число подобластей в объединении, число 8 — это число макроузлов в подобласти, число 7 — это число коэффициентов в сеточном уравнении.

Выборка, например, 3-го коэффициента c_t для 5-го макроузла 7-й подобласти осуществляется как $c_t = C_m(7, 5, 3)$.

4. Иллюстративный пример. Рассмотрим расчетную область G с криволинейной границей Γ , изображенную на рис. 5. Эта область заключена между двумя соосными цилиндрами с диаметрами d и

D ($D > d$) и плоскостями $Z = -L, 0$. Опишем вокруг G параллелепипед

$$\bar{Q} = \{0 \leq X \leq D, 0 \leq Y \leq D, -L \leq Z \leq 0\}.$$

Построим макросетку с числом интервалов $8 \times 8 \times 1$. Поясним: отрицательные координаты по оси Z и $N_Z = 1$ выбраны для наглядности изображения. В связи с этим третий индекс при нумерации подобластей и подсеток будем опускать. Подобласти нумеруются естественным порядком слева направо и снизу вверх. Подобласти с номерами $(1, 1), (1, 8), (8, 1), (8, 8)$ — внешние, подобласти с номерами $\{(1, J), J = \overline{2, 7}\}, \{(I, J), I = \overline{2, 7}, J = 1, 8\}, \{(8, J), J = \overline{2, 7}\}, (2, 1), (2, 8), (7, 1), (7, 8)$ — граничные, а остальные подобласти — внутренние.

Подсетки в подобластях построены так, что они сгущаются к центру. Это отвечает требованиям точности решения задачи, так как при $d \rightarrow 0$ возникает особенность и при конечных, но малых d необходима более подробная сетка. На рис. 5 подсетки имеют размеры $16 \times 16, 8 \times 8, 4 \times 4$. Внешние узлы граничных подсеток и координатные линии изображены бледным цветом. Во внешних подобластях подсетки не строятся.

Подсетки группируются в 4 объединения, содержащие одинаковое число внешних узлов. На рис. 5 граница объединений показана жирными линиями. Например, левое-нижнее объединение включает подсетки с номерами $(1, 2), (1, 3), (1, 4), (2, 2), (2, 3), (2, 4), (3, 1), (3, 2), (3, 3), (3, 4), (4, 1), (4, 2), (4, 3), (4, 4)$. Распараллеливание проводится на 4 процессорах, каждый из которых отвечает за одно объединение.

5. Заключение. Изложены основные технологии и структуры данных для распараллеливания решения трехмерных краевых задач в сложных областях на квазиструктурированных сетках. Предложен принцип хранения данных, поддерживающий структурированные массивы. Разработаны параллельные структуры данных для хранения объединений подсеток, создаваемых с целью балансировки загрузки процессоров. Предложены способы распараллеливания сеток на интерфейсе, предполагающие хранение минимального объема информации, необходимой для работы текущего процессора. Рассмотрен иллюстративный пример, наглядно показывающий основные детали разработанного подхода. Разработанные технологии и структуры данных будут использованы при построении программного комплекса для решения трехмерных задач расчета интенсивных пучков заряженных частиц в сложных областях.

Работа поддержана грантом РНФ (проект 14-11-00485п) и грантом РФФИ (проект 16-01-00168).

СПИСОК ЛИТЕРАТУРЫ

1. *Алямовский И.В.* Электронные пучки и электронные пушки. М.: Советское радио, 1966.
2. *Ильин В.П.* Численные методы решения задач электрофизики. М.: Наука, 1985.
3. *Молоковский С.И., Сушков А.Д.* Интенсивные электронные и ионные пучки. М.: Энергоатомиздат, 1991.
4. *Сыровой В.А.* Введение в теорию интенсивных пучков заряженных частиц. М.: Энергоатомиздат, 2004.
5. NETGEN mesh generator. <https://sourceforge.net/projects/netgen-mesher/>.
6. *Василевский Ю.В., Ольшанский М.А.* Краткий курс по многосеточным методам и методам декомпозиции области. М.: Изд-во Моск. ун-та, 2007.
7. *Quarteroni A., Valli A.* Domain decomposition methods for partial differential equations. Oxford: Clarendon Press, 1999.
8. *Dolean V., Jolivet P., Nataf F.* An introduction to domain decomposition methods: algorithms, theory and parallel implementation. Philadelphia: SIAM Press, 2015.
9. *Корнеев В.Д., Свешников В.М.* Параллельные алгоритмы и технологии декомпозиции расчетной области для решения трехмерных краевых задач на квазиструктурированных сетках // Сиб. журн. вычисл. матем. 2016. **19**, № 2. 183–194.
10. *Климонов И.А., Корнеев В.Д., Свешников В.М.* Технологии распараллеливания решения трехмерных краевых задач на квазиструктурированных сетках в гибридной вычислительной среде CPU+GPU // Вычислительные методы и программирование: новые вычислительные технологии. 2016. **17**. 65–71.
11. *Ильин В.П., Голубцов Б.И.* Автоматизация решения краевых задач для уравнения Пуассона. Новосибирск: Наука, 1969.
12. *Козырев А.Н., Свешников В.М.* О построении двумерных локально-модифицированных квазиструктурированных сеток и решении на них краевых задач в областях с криволинейной границей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2017. **6**, № 2. 5–21.
13. *Свешников В.М., Рыбдылов Б.Д.* О распараллеливании решения краевых задач на квазиструктурированных сетках // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2013. **2**, № 3. 63–72.
14. *Свешников В.М.* Построение прямых и итерационных методов декомпозиции // Сибирский журнал промышленной математики. 2009. **12**, № 3. 99–109.
15. *Ильин В.П.* Методы и технологии конечных элементов. Новосибирск: ИВМиМГ, 2007.

16. Ильин В.П. Методы конечных разностей и конечных объемов для эллиптических уравнений. Новосибирск: ИВМиМГ, 2001.

Поступила в редакцию
5.07.2018

Parallelization Technologies and Grid Data Structures for Solving Three-Dimensional Boundary Value Problems in Complex Domains on Quasistructured Grids

V. D. Korneev¹ and V. M. Sveshnikov²

¹ *Institute of Computational Mathematics and Mathematical Geophysics, Siberian Branch of Russian Academy of Sciences; prospekt Lavrentyeva 6, Novosibirsk, 630090, Russia; Ph.D., Associate Professor, e-mail: korneev@ssd.sgcc.ru*

² *Institute of Computational Mathematics and Mathematical Geophysics, Siberian Branch of Russian Academy of Sciences; prospekt Lavrentyeva 6, Novosibirsk, 630090, Russia; Dr. Sci., Professor, Principal Scientist, e-mail: victor@lapasrv.sgcc.ru*

Received July 5, 2018

Abstract: When parallelizing the solution of three-dimensional boundary value problems, especially in domains with complex geometry, the computational technologies and data structures are important. The amount of stored information and the computational time depend on them. In this paper we propose the technologies for parallelizing the method of decomposition of the computational domain into subdomains conjugated without overlapping on a quasistructured grid. Parallel grid data structures oriented mainly to work with structured data arrays are developed. An illustrative example clarifying the fundamentals of the proposed approach is discussed.

Keywords: boundary value problems, quasistructured grids, parallelization technologies, data structures, structured arrays.

References

1. I. V. Alyamovskii, *Electron Beams and Electron Guns* (Sov. Radio, Moscow, 1966) [in Russian].
2. V. P. Il'in, *Numerical Methods for Solving Electrophysical Problems* (Nauka, Moscow, 1985) [in Russian].
3. S. I. Molokovskii and A. D. Sushkov, *Intense Electron and Ion Beams* (Energoatomizdat, Moscow, 1991) [in Russian].
4. V. A. Syrovoi, *Introduction to the Theory of Intense Beams of Charged Particles* (Energoatomizdat, Moscow, 2004) [in Russian].
5. NETGEN mesh generator. <https://sourceforge.net/projects/netgen-mesher>. Cited November 20, 2018.
6. Yu. V. Vasilevskii and M. A. Ol'shanskii, *A Short Course on Multigrid and Domain Decomposition Methods* (Mosk. Gos. Univ., Moscow, 2007) [in Russian].
7. A. Quarteroni and A. Valli, *Domain Decomposition Methods for Partial Differential Equations* (Clarendon Press, Oxford, 1999).
8. V. Dolean, P. Jolivet, and F. Nataf, *An Introduction to Domain Decomposition Methods: Algorithms, Theory, and Parallel Implementation* (SIAM Press, Philadelphia, 2015).
9. V. D. Korneev and V. M. Sveshnikov, "Parallel Algorithms and Domain Decomposition Techniques for Solving Three-Dimensional Boundary Value Problems on Quasi-Structured Grids," *Sib. Zh. Vych. Mat.* **19** (2), 183–194 (2016) [Numer. Anal. Appl. **9** (2), 141–149 (2016)].
10. I. A. Klimonov, V. D. Korneev, and V. M. Sveshnikov, "Parallelization Technologies for Solving Three-Dimensional Boundary Value Problems on Quasi-Structured Grids Using the CPU+GPU Hybrid Computing Environment," *Vychisl. Metody Programm.* **17**, 65–71 (2016).
11. V. P. Il'in and B. I. Golubtsov, *Automation of Solving Boundary Value Problems for Poisson's Equation* (Novosibirsk, Nauka 1969) [in Russian].
12. A. N. Kozyrev and V. M. Sveshnikov, "On the Construction of Two-Dimensional Local-Modified Quasistructured Grids and Solving on Them Two-Dimensional Boundary Value Problems in the Domains with Curvilinear Boundary," *Vestn. Yuzhn. Ural. Gos. Univ. Ser. Vychisl. Mat. Inf.* **6** (2), 5–21 (2017).

13. V. M. Sveshnikov and B. D. Rybdylov, "About Parallelization of Solving of Boundary Value Problems on Quasistructured Grids," *Vestn. Yuzhn. Ural. Gos. Univ. Ser. Vychisl. Mat. Inf.* **2** (3), 63–72 (2013).
14. V. M. Sveshnikov, "Construction of Direct and Iterative Decomposition Methods," *Sib. Zh. Ind. Mat.* **12** (3), 99–109 (2009) [*J. Appl. Ind. Math.* **4** (3), 431–440 (2010)].
15. V. P. Il'in, *Methods and Technologies of Finite Elements* (Inst. Comput. Math. Math. Geophys., Novosibirsk, 2007) [in Russian].
16. V. P. Il'in, *Finite Difference and Finite Volume Methods for Elliptic Equations* (Inst. Comput. Math. Math. Geophys., Novosibirsk, 2001) [in Russian].