

ПАРАЛЛЕЛЬНЫЕ ТЕХНОЛОГИИ РЕШЕНИЯ КРАЕВЫХ ЗАДАЧ НА КВАЗИСТРУКТУРИРОВАННЫХ СЕТКАХ

В.М. Свешников

Введение

На примере смешанной краевой задачи для уравнения Пуассона рассмотрены технологические аспекты распараллеливания решения краевых задач на адаптивных квазиструктурированных сетках в двумерных (плоских или осесимметричных) областях со сложной конфигурацией границы. Квазиструктурированная сетка строится в два этапа: сначала расчетная область покрывается равномерной прямоугольной макросеткой, а затем в каждом макроэлементе строится своя прямоугольная равномерная микросетка, то есть микросетки могут быть несогласованными. Адаптивность результирующей сетки достигается за счет ее модификации вблизи границы и за счет регулировки плотности узлов в подобластях. Фактически при построении макросетки проводится декомпозиция расчетной области на подобласти, каждая из которых является макроэлементом. Метод декомпозиции области служит основным инструментом распараллеливания решения краевых задач на многопроцессорных суперЭВМ [1], но непосредственное его применение с отображением одна подобласть – один процессор в данном случае может привести к значительной разбалансировке загрузки процессоров.

Основным технологическим приемом, который предлагается в настоящей статье, является группировка подобластей в объединения с целью обеспечения приблизительно равной загрузки процессоров при отображении одно объединение – один процессор. Решение в подобластях ищется итерационным методом декомпозиции с сопряжением типа Дирихле – Дирихле [2]. Распараллеливанию подвергается лишь процесс проведения итераций. Все необходимые для него данные: разметка узлов сетки (внешний, внутренний, граничный), расчет коэффициентов сеточных уравнений проводятся на персональном компьютере при помощи пакета прикладных программ ЭРА-DD [3], а затем передаются на суперЭВМ. Формирование этих данных требует значительно меньшего числа операций по сравнению с итерационным процессом. Работа с объединениями привносит в организацию вычислений ряд особенностей, вызванных тем, что объединения могут иметь различное число соседей, с которыми необходимо осуществлять обмен. В работе рассматриваются структура данных и основные технологические этапы проведения расчетов. Отметим, что при описании алгоритма решения задачи даются лишь его основные этапы, необходимые для понимания технологических решений, предлагаемых в настоящей статье. Более подробно алгоритм изложен в работе [2] и цитируемой там литературе.

Постановка задачи и алгоритм ее решения

Пусть требуется решить краевую задачу

$$\Delta u = g_1, \quad (u)_i g_2, \quad \left(\frac{\partial u}{\partial \vec{n}} \right)_i g_3 \quad (1)$$

в замкнутой двумерной (плоской или осесимметричной) области $\bar{G} = G \cup \Gamma$ с границей $\Gamma = \Gamma_1 \cup \Gamma_2$, причем будем предполагать, что Γ_1 аппроксимируется отрезками кривых второго порядка, а Γ_2 состоит из отрезков, параллельных координатным осям. Здесь $u = u(T)$ – искомая функция, $g_1 = g_1(T)$, $g_2 = g_2(T)$, $g_3 = g_3(T)$, – заданные функции ($T = (x, y)$ – текущая точка, где x, y – декартовы или цилиндрические координаты), Δ – оператор Лапласа, \vec{n} – нормаль к Γ_2 . Построим в прямоугольнике \bar{R} ($\bar{G} \subset \bar{R}$) равномерную макросетку Ω_H с шагами намного превышающими максимальный шаг сетки, на которой аппроксимируется задача (1). При этом область G разбивается на подобласти G_k , среди которых будем различать внутренние $G_k^{(1)}$, граничные $G_k^{(2)}$ и внешние подобласти. Замыкание $\bar{G}_k^{(1)}$ внутренних подобластей состоит только из точек G , замыкание граничных $\bar{G}_k^{(2)}$ – из точек G и Γ , а внешние подобласти исключаются из расчетов. Граница сопряжения подобластей (или интерфейс) γ состоит из отрезков координатных линий макросетки Ω_H . Точки пересечения ее координатных линий назовем макроузлами.

В подобластях \bar{G}_k построим равномерные прямоугольные подсетки $\bar{\Omega}_{h,k}$. Будем предполагать без существенного ограничения общности, что числа интервалов подсеток есть целые степени 2. В целях адаптации в граничных подобластях проведем локальную модификацию подсеток, состоящую в сдвиге приграничных узлов, отстоящих от границы на расстояние, меньшее половины шага подсетки, в ближайшие точки пересечения координатных линий с границей. Объединение всех подсеток образует результирующую

квазиструктурированную сетку $\bar{\Omega}_h$. Краевая задача (1) методом конечных объемов [4] сводится к решению приближенных подзадач на подсетках $\bar{\Omega}_{h,k}$, причем на отрезках границы сопряжения $\gamma_k \subset \gamma$ ставится условие Дирихле для искомых функций $u_h^{(k)}$

$$\left(u_h^{(k)} \right)_i v_h^{(k)}$$

В свою очередь, каждая подзадача дает систему в общем случае девятиточечных линейных алгебраических уравнений, переходящих в пятиточечные уравнения в узлах, не примыкающих к Γ , которых подавляющее большинство в $\bar{\Omega}_h$.

Для нахождения $v_h = \bigcup_k v_h^{(k)}$ введем на границе сопряжения сетку ω_h , состоящую из узлов подсеток $\bar{\Omega}_{h,k}$, лежащих на γ , и не содержащую макроузлов. Пусть N – число узлов ω_h . В работе [2] показано, что функцию v_h можно найти из решения системы линейных алгебраических уравнений вида

$$A v_h + b = 0, \quad (2)$$

где A – квадратная матрица порядка N , а $b = (b_i, i = 1, \dots, N)$ – вектор. Для вычисления компонент вектора b решаются вспомогательные краевые задачи в подобластях с граничными условиями (1) и нулевыми значениями искомых функций $\psi_h^{(k)}$ на границе сопряжения. Тогда

$$b_i^{(k)} = \left(D_h^{(k)} \psi_h^{(k)} \right)_i,$$

где $b = \bigcup_k b^{(k)}$, $b_k = (b_i^{(k)})$, а операторы $D_h^{(k)}$ определяются как

$$\left(D_h^{(k)} \psi_h^{(k)} \right)_i = \left(d_h^{(k,+)} \psi_h^{(k,+)} \right)_i + \left(d_h^{(k,-)} \psi_h^{(k,-)} \right)_i. \quad (3)$$

Здесь $d_h^{(k,\pm)}$ – операторы, аппроксимирующие внутренние нормальные производные по разные стороны от границы сопряжения.

Элементы матрицы A явно не вычисляются. Решение системы (2) проводится методом сопряженных градиентов по формулам [4]

$$r^{(0)} = b - A v_h^{(0)}, \quad p^{(0)} = r^{(0)}, \quad (4)$$

$$\alpha \dot{r}_n = (r^{(n)}, r^{(n)}) / (A p^{(n)}, p^{(n)}), \quad (5)$$

$$v_h^{(n+1)} = v_h^{(n)} + \dot{r}_n p^{(n)}, \quad \dot{r}_n$$

$$r^{(n+1)} = r^{(n)} - \dot{r}_n A p^{(n)}, \quad p^{(n+1)} = p^{(n)} + \beta_n p^{(n)}, \quad (6)$$

$$\beta_n = \frac{(r^{(n+1)}, r^{(n+1)})}{(r^{(n)}, r^{(n)})}, \quad n = 0, 1, \dots \quad (7)$$

Для вычисления $A w_h$, где под w_h понимаются векторы $v_h^{(0)}$, $p^{(n)}$, фигурирующие в приведенных выше формулах, решаются вспомогательные краевые задачи в подобластях с граничными условиями (1) и значениями w_h искомых функций $\varphi_h^{(k)}$ на границе сопряжения. Тогда искомое произведение равно

$$A w_h = D_h \varphi_h - b,$$

где $D_h \varphi_h = \bigcup_k D_h^{(k)} \varphi_h^{(k)}$.

После того, как найдены значения v_h в узлах сетки ω_h , вычисляются значения $v_h^{(p)}$ искомой функции в макроузлах из соотношения

$$\Delta_h^{(p)} v_h^{(p)} = g_h^{(p)}.$$

Здесь $g_h^{(p)}$ значения правой части (1) в макроузлах, $\Delta_h^{(p)}$ – аппроксимация оператора Лапласа на сеточном шаблоне, включающем узлы сетки ω_h и один из макроузлов.

Структура данных и параллельные технологии проведения расчетов

Геометрическая и функциональная информация о расчетной области, граничных условиях и квазиструктурированной сетке задаются при помощи специального графического интерфейса, входящего в состав пакета прикладных программ ЭРА-DD [3], ориентированного на работу на персональном компьютере. Вычислительными модулями данного пакета проводится построение квазиструктурированной сетки, ее локальная модификация и расчет коэффициентов сеточных уравнений. В целях удобства работы с

квазиструктурированной сеткой для каждой k -й внутренней или граничной подобласти хранятся все узлы структурированной подсетки $\bar{\Omega}_{h,k}$, включая внешние узлы. Среди узлов, принадлежащих расчетной области \bar{G} , различаются внутренние узлы, координаты которых не хранятся, так как вычисляются по простым правилам, следующим из структуры $\bar{\Omega}_{h,k}$, и граничные или модифицированные узлы, координаты которых хранятся в специальном массиве. С точки зрения хранения коэффициентов сеточных уравнений различаются стандартные узлы, которые имеют набор коэффициентов, единый для всей подсетки, и нестандартные, набор коэффициентов которых может различаться. К последним относятся внутренние узлы, хотя бы один из соседей которых является модифицированным. Массивы информации, полученные в результате этих расчетов, пересылаются многопроцессорный компьютер.

Распараллеливанию подлежит итерационный процесс по подобластям, который занимает подавляющую часть времени решения всей задачи. Важным звеном в технологической цепи решений по достижению эффективности распараллеливания является отображение сеточных данных на вычислительную сеть. Обычно принятый способ отображения: одна подобласть – один процессор в данном случае не эффективен, так как подобласти могут содержать различное число узлов, в которых вычисляются значения искомой функции (в дальнейшем – счетных узлов), что приводит к разбалансировке загрузки процессоров. Поэтому подобласти группируются в объединения $U_l, l=1, L$, где L – известное число, с целью обеспечения приблизительно равной загрузки процессоров. Для этого в каждое объединение $U_l = \cup_k \bar{\Omega}_{h,k}^{(l)}$ включаются такие подсетки $\bar{\Omega}_{h,k}^{(l)}$, которые давали бы в сумме $N_l = \sum_k N_l^{(k)}$ число счетных узлов $N_l \approx N_U$, где N_U – заданная величина, а $N_l^{(k)}$ – число счетных узлов в подобластях. Алгоритм группировки подобластей для идеальной балансировки должен строиться на решении задачи линейного программирования. При этом может оказаться, что в объединение включаются не только соседние подобласти, но и подобласти, разделенные подобластями из других объединений. Последнее обстоятельство может привести, как будет видно из дальнейшего, к значительному увеличению объема самых медленных операций, а именно операций пересылки в системах с разделенной памятью. Обмены не происходят между соседними подобластями одного объединения. В связи с этим был принят следующий алгоритм построения объединений. Определяется число N_U , которое может быть равно максимальному числу узлов в подобласти или вычисляться как $N_U = N_h / L$, где N_h число счетных узлов во всей квазиструктурированной сетке $\bar{\Omega}_h$. Для каждой свободной, то есть не включенной ни в одно объединение, подобласти просматриваются свободные соседи. Количество узлов просмотренных подсеток суммируется. Если после просмотра k -й подсетки в l -м объединении окажется $N_l > N_U$, то процесс группировки заканчивается, причем при $N_l - N_U > 0.5 N_l^{(k)}$ последняя подсетка не включается в данное объединение.

На нулевом процессоре, который выбран в качестве управляющего, проводится прием информации с персонального компьютера и ее предварительная обработка. На этом же процессоре формируются объединения. В результате на каждый процессор, включая 0, рассылается информация следующих типов: 1) общая для всех объединений, 2) специальная для конкретного объединения.

К первому типу относятся следующие массивы (принята фортрановская форма записи): NNSG(nx,ny,2) – числа интервалов подсеток по x, y (nx, ny – числа интервалов макросетки); ITSG(nx,ny) – типы подобластей (внешняя, внутренняя, граничная); COLM() – координаты модифицированных узлов: для каждого узла хранится x, y, ig . причем последнее число – номер границы, на которой лежит рассматриваемый узел, COE1() – коэффициенты сеточных уравнений для узлов, лежащих вблизи границы расчетной области, COE2(nx,ny,3) – стандартные коэффициенты сеточных уравнений в подобластях (для каждого узла – 3 коэффициента), NPIJ(nx,ny) – номера процессоров, отвечающих за каждую подобласть; ISIU(nsu) – числа подобластей в объединениях (nsu – число объединений), NSIU() – номера (I,J) подобластей в объединениях, IASU(nsu) – опорные адреса начал подмассивов для каждого объединения в массиве номеров NSIU().

Ко второму типу информации относятся следующие массивы: XYSK() – координаты вершин прямоугольных подобластей в данном объединении, ITNK() – типы узлов подсеток, входящих в объединение, RUNK() – значения искомой функции, RFNK() – значения правой части, IPRS(npr) – номера процессоров-соседей для чтения (npr – их число). IPWS(npw) – номера процессоров-соседей для записи (npw – их число).

На текущем процессоре выполняются следующие вычислительные работы:

1. Вычисление искомой функции в подобластях. Отметим, что подобласти в объединении перебираются в том же порядке, в котором они включались в объединение, поэтому хранить специальный массив опорных адресов для подмассивов, входящих в RUNK() и RFNK() не требуется. По поводу формирования значений граничных условий в подобластях см. ниже подпункт 3.
2. Расчет внутренних нормальных производных на сторонах подобластей, входящих в интерфейс. При этом вводятся массивы: RDUK() – значений производных, IADK() – опорных адресов начал подмассивов на сторонах. Принято правило, согласно которому в формировании интерфейса на текущем процессоре

участвуют правая и верхняя стороны подобластей. По окончании расчетов массивы $RDUK(\)$, $IADK(\)$ пересылаются на процессоры-соседи для записи.

3. Вычисление сумм производных (3) на смежных сторонах подобластей. Напомним, что на смежных сторонах могут быть определены подсетки, содержащие различное число узлов, а в интерфейс входят все узлы густой подсетки. Производные, необходимые для вычисления сумм в «оборванных» узлах, вычисляются путем линейной интерполяции (экстраполяции) по ближайшим узлам. Перед проведением вычислений с каждого процессора-соседа читаются массивы производных $RDUK(\)$ и опорных адресов $IADK(\)$ соответственно в массивы $RDUS(\)$ и $IADS(\)$.
4. Реализация очередного шага итерационного процесса по подобластям. При этом формируются массивы $RINK(\)$, $PINK(\)$, $APIK(\)$, $UINK(\)$, предназначенные соответственно для хранения векторов $r^{(n)}$, $p^{(n)}$, Aw_h , $v^{(n)}$. На каждом процессоре вычисляются частичные суммы, входящие в скалярные произведения $(Ap^{(n)}, p^{(n)})$, $(r^{(n)}, r^{(n)})$. Их окончательное вычисление проводится на управляющем процессоре, который рассылает значения α_n , β_n на все процессоры, после чего проводятся вычисления по формулам (4)-(7). При этом предусмотрена специальная процедура рассылки величин, определенных в узлах интерфейса в подсетки на сторонах подобластей, связанная с пересылкой информации. Суть ее работы заключается в следующем. Подмассивы массива $VINK(\)$, содержащего значения функций на интерфейсе, рассылаются в подмассивы массива $RUNK(\)$ для правой/верхней сторон рассматриваемой подобласти, а также для левой/нижней сторон соседних правых/верхних подобластей того же объединения. Если же данные соседи лежат в других объединениях (на других процессорах), то рассылки не производится. Если соседние левая/нижняя подобласти лежат в других объединениях то, наоборот, проводится рассылка значений в подмассивы массива $RUNK(\)$ для левой/нижней сторон текущей подобласти из массивов $VINK(\)$ этих объединений, предварительно считанных с соответствующих процессоров, и не проводится рассылки в противном случае.

В заключении отметим, что введенные квазиструктурированные сетки могут привести к существенной экономии вычислительных ресурсов (времени и требуемой памяти) при решении сложных практических задач с сильной неоднородностью и разномасштабностью по сравнению с расчетами на структурированных прямоугольных сетках. К тому же, решение линейной стационарной задачи, пример которой рассмотрен выше, может требоваться многократно при решении нелинейных и/или нестационарных задач, которые, в свою очередь, многократно решаются в задачах оптимизации. Поэтому можно сказать, что некоторое усложнение алгоритмов и технологии распараллеливания, рассмотренные в настоящей статье, оправданы.

Работа выполнена при финансовой поддержке РФФИ (проекты № 08-01-00526-а, № 10-01-90009-Бел_а) и СО РАН (проекты ИП26, ИП 59)

ЛИТЕРАТУРА:

1. A. Quarteroni., A Valli. Domain Decomposition Methods for Partial Differential Equations // Oxford: Clarendon Press. 1999.
2. В.М. Свешников "Построение прямых и итерационных методов декомпозиции" // Сибирский журнал индустр. матем. Т12, №3. 2009. С.99-109.
3. Д.О. Беляев, А.Н. Козырев, В.М. Свешников "Пакет прикладных программ ЭРА-DD для решения двумерных краевых задач на квазиструктурированных сетках" // Вестник НГУ. Серия: Информационные технологии. Т.8, вып. 1. 2010. С. 3-11
4. В.П. Ильин Методы конечных разностей и конечных объемов для эллиптических уравнений. // Новосибирск: ИВМиМГ (ВЦ) СО РАН. 2001.