

Оценки эффективности прямого метода декомпозиции*

В. М. СВЕШНИКОВ

*Институт вычислительной математики
и математической геофизики СО РАН, Новосибирск, Россия*
e-mail: victor@lapasrv.sscs.ru

Estimations of labor costs and volume of saved information for a realization of the algorithm based on a direct decomposition method proposed by the author are given. The results of numerical experiments suggest that the algorithm is robust. Basic properties of the proposed approach are listed in the conclusion.

Введение

В статье [1] был предложен вариант метода декомпозиции без наложения подобластей для решения краевых задач на прямоугольных квазиструктурированных сетках. В настоящей работе оценивается эффективность предложенного подхода, даются результаты численных экспериментов, показывающие его точность, и перечисляются его основные свойства.

1. Постановка задачи и краткое описание алгоритма ее решения

Приведем основные сведения о рассматриваемом алгоритме, которые необходимы для дальнейшего изложения.

Пусть в замкнутой двумерной области $\bar{G} = G \cup \Gamma$ с границей Γ требуется решить краевую задачу

$$\Delta u = g_1, \quad l u|_{\Gamma} = g_2. \quad (1)$$

Здесь $u = u(T)$ — искомая функция; $g_1 = g_1(T)$, $g_2 = g_2(T)$ — заданные функции ($T = (x, y)$ — текущая точка, где x, y — декартовы координаты); Δ — оператор Лапласа; l — оператор граничных условий.

Опишем вокруг расчетной области прямоугольник $R = \{0 \leq x \leq D_x, 0 \leq y \leq D_y\}$, где D_x, D_y — заданы ($\bar{G} \subset R$). Построим в R прямоугольную равномерную макросетку

$$\Omega_H = \left\{ X_I = IH_x, Y_J = JH_y, I = \overline{0, N_x}, J = \overline{0, N_y}, H_x = \frac{D_x}{N_x}, H_y = \frac{D_y}{N_y} \right\},$$

где N_x, N_y — заданные целые числа с шагами $H_x, H_y \gg h$ (h — максимальный шаг сетки, на которой аппроксимируется задача (1)). Тем самым мы фактически проводим

*Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (грант № 08-01-00526).

© Институт вычислительных технологий Сибирского отделения Российской академии наук, 2008.

декомпозицию расчетной области G на непустые подобласти $G_k = G_{I,J} = G \cap R_{I,J} \neq \emptyset$, $k = \overline{1, K}$, где $R_{I,J}$ — открытая прямоугольная подобласть, представляющая собой макроэлемент сетки Ω_H ; K — известное целое число, которое определяется геометрией \bar{G} , (цифрой 0 в данном случае обозначена пустая подобласть). Точки $T_p = T_{I,J} \in G$, $p = \overline{1, P}$, пересечения координатных линий $x = X_I$, $y = Y_J$ макросетки назовем макроузлами. Граница сопряжения подобластей, которую мы обозначим через γ , состоит из отрезков координатных линий макросетки Ω_H . Среди G_k будем различать подобласти двух типов: *внутренние* $G_k^{(1)}$ и *граничные* $G_k^{(2)}$ (рис. 1). Различие между ними состоит в том, что замыкание граничной подобласти содержит отрезки границы Γ , а внутренней — не содержит их. На рис. 1 внутренние подобласти ограничены сплошными линиями, а граничные — штриховыми. Обозначим через K_1, K_2 — соответственно число внутренних и число граничных подобластей ($K_1 + K_2 = K$) и M_1, M_2 — множества номеров (значений индекса k) соответственно внутренних и граничных подобластей. Пусть $\gamma_k \subset \gamma$ — отрезок границы подобласти G_k . Если рассматриваемая подобласть внутренняя ($G_k = G_k^{(1)}$), то вся ее граница совпадает с γ_k , а если — граничная ($G_k = G_k^{(2)}$), то помимо γ_k ее граница включает Γ_k — отрезок границы Γ .

В замкнутых макроэлементах $\bar{R}_k = \bar{R}_{I,J}$ построим равномерные прямоугольные подсетки

$$\bar{\Omega}_{h,k} = \{x_{i_k} = X_I + i_k h_{x,k}, y_{j_k} = Y_J + j_k h_{y,k}, i_k = \overline{0, n_{x,k}}, j_k = \overline{0, n_{y,k}}\}$$

с шагами $h_{x,k} = \frac{X_{I+1} - X_I}{n_{x,k}}$, $h_{y,k} = \frac{Y_{J+1} - Y_J}{n_{y,k}}$, причем будем предполагать без существенного ограничения общности, что $n_{x,k} = 2^{m_{x,k}}$, $n_{y,k} = 2^{m_{y,k}}$, где $m_{x,k}, m_{y,k} > 0$ — целые числа. По аналогии с подобластями будем различать подсетки двух типов — *внутренние* $\bar{\Omega}_{h,k}^{(1)}$ и *граничные* $\bar{\Omega}_{h,k}^{(2)}$.

Решение исходной задачи (1) будем искать на результирующей сетке

$$\bar{\Omega}_h = \bigcup_{k=1}^K \bar{\Omega}_{h,k}. \tag{2}$$

Кроме рассмотренных выше, введем сетку $\omega_h \subset \bar{\Omega}_h$, $\omega_h = \{(x_i, y_i) \in \gamma, (x_i, y_i) \neq T_p, i = \overline{1, n}\}$, где (x_i, y_i) — узлы $\bar{\Omega}_h$, а также подсетки $\omega_{h,k}$, состоящие из узлов ω_h , лежащих на γ_k (n — известно). Подчеркнем что в ω_h , $\omega_{h,k}$ не входят макроузлы T_p .

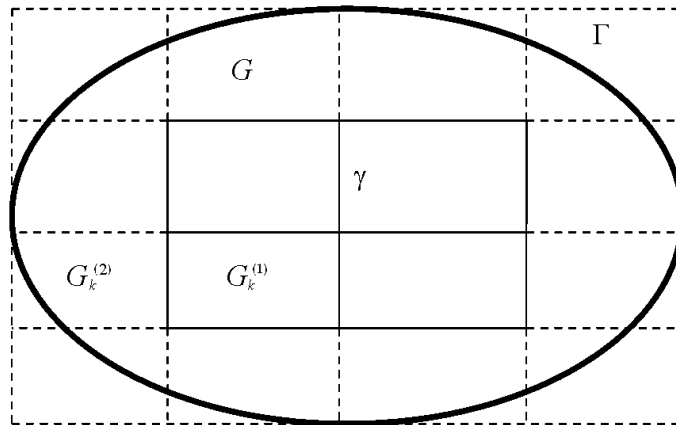


Рис. 1. Декомпозиция расчетной области

Наряду с введенными, будем иметь дело с сеткой Ω_h и подсетками $\Omega_{h,k}$, которые совпадают с сетками $\bar{\Omega}_h$ и $\bar{\Omega}_{h,k}$, но не включают узлы, лежащие на границе сопряжения.

Примем следующие соглашения по нумерации элементов и узлов во введенных сетках. Для макроэлементов и макроузлов в структурированной сетке Ω_H принимается стандартная нумерация: слева направо, снизу вверх. В соответствии с ней нумеруются подобласти G_k (подчеркнем, что внешние подобласти не входят в нумерацию по индексу k). В структурированных подсетках $\Omega_{h,k}$ также вводится стандартная нумерация. Упорядочим узлы сетки ω_h , определенной на γ , следующим образом: сначала пронумеруем слева направо узлы, лежащие на координатных линиях сетки Ω_H , параллельных оси x (x -линиях), причем сами координатные линии будем перебирать снизу вверх. Затем пронумеруем снизу вверх узлы на y -линиях, перебирая последние слева направо.

Алгоритм решения краевой задачи (1) на построенной таким образом квазиструктурированной сетке включает в себя три этапа:

- 1) построение системы линейных алгебраических уравнений (СЛАУ) относительно искомой функции на границе сопряжения подобластей;
- 2) решение данной системы;
- 3) вычисление искомой функции в макроузлах и подобластях.

Запишем СЛАУ первого этапа в виде

$$Av_h + b = 0. \quad (3)$$

Здесь $A = \{a_{i,j}\}$ — квадратная невырожденная матрица, $b = \{b_i\}$, $v_h = \{v_{h,i}\}$ — векторы ($i, j = \bar{1}, n$). Компоненты вектора v_h есть значения искомой функции в узлах сетки ω_h , расчет A , b рассмотрен ниже.

Вычисление элементов b_i , $a_{i,j}$ основано на решении вспомогательных подзадач в подобластях G_k вида

$$\Delta_h^{(k)} \psi_h^{(j,k)} = z_{1,j}^{(k)}, \quad \psi_h^{(j,k)} \Big|_{\gamma_k} = e^{(j,k)}, \quad k \in M_1, \quad (4)$$

на внутренних подсетках $\Omega_{h,k}^{(1)}$ и подзадач

$$\Delta_h^{(k)} \psi_h^{(j,k)} = z_{1,j}^{(k)}, \quad l_h^{(k)} \psi_h^{(j,k)} \Big|_{\Gamma_k} = z_{2,j}^{(k)}, \quad \psi_h^{(j,k)} \Big|_{\gamma_k} = e^{(j,k)}, \quad k \in M_2, \quad (5)$$

на граничных подсетках $\Omega_{h,k}^{(2)}$, где $j = \overline{0, n}$. Здесь операторы $\Delta_h^{(k)}$, $l_h^{(k)}$ — аппроксимации оператора Лапласа и оператора граничных условий на данных подсетках, величины $z_{r,j}^{(k)}$ имеют вид

$$z_{r,j}^{(k)} = \begin{cases} 0, & j > 0, \\ g_{r,h}^{(k)}, & j = 0, \end{cases} \quad r = 1, 2,$$

где $g_{r,h}^{(k)}$ — аппроксимации функций g_r в узлах k -й подсетки, а функции (векторы) $e^{(j,k)}$ определяются как

$$e^{(j,k)} = \begin{cases} 0, & j = 0, \\ e^{(j,k)}(T_i) = \delta_{i,j}, & T_i \in \omega_{h,k}, \quad j > 0, \end{cases}$$

где $\delta_{i,j}$ — символ Кронекера. То есть на каждой подсетке $\Omega_{h,k}$ решаются подзадачи (4) и (5) двух видов: 1) относительно функций $\psi_h^{(0,k)}$ с нулевыми значениями на подсетке $\omega_{h,k}$

и 2) серии подзадач относительно функций $\psi_h^{(j,k)}$, $j > 0$, в каждой из которых данная функция поочередно полагается равной единице в одном из узлов подсетки $\omega_{h,k}$ и нулю — в остальных.

Количество подзадач второго вида значительно сокращается благодаря свойствам введенной квазиструктурированной сетки, в частности, тому, что все макроэлементы $R_{i,j}$ равны между собой. Это сокращение в первую очередь относится к внутренним подобластям. Для них решается серия так называемых базовых подзадач для одной подобласти относительно функций, которые обозначим через $\sigma_h^{(\nu)}$, следующего вида:

$$\Delta_h \sigma_h^{(\nu)} = 0, \quad \sigma_h^{(\nu)}(T_\mu) = \delta_{\nu,\mu}, \quad T_\mu \in \gamma_0, \quad (6)$$

на самой густой подсетке $\Omega_{h,k}^{(1)} = \Omega_{h,0}^{(1)}$. Здесь γ_0 — граница внутренней подобласти $G_0^{(1)} \in R_0$, в которой построена подсетка $\Omega_{h,0}^{(1)}$, а ν, μ — номера узлов подсетки $\omega_{h,0}$ — проекции $\bar{\Omega}_{h,0}^{(1)}$ на γ_0 (без макроузлов).

Если в k -й внутренней подобласти подсетка $\Omega_{h,k}^{(1)}$ совпадает с подсеткой $\Omega_{h,0}^{(1)}$, то $\psi_h^{(\nu,k)} = \sigma_h^{(\nu)}$. Если же это не так и подсетка $\Omega_{h,k}^{(1)}$ более редкая по сравнению с $\Omega_{h,0}^{(1)}$, то требуемые функции вычисляются согласно следующему принципу. Для простоты его изложения предположим, что в исходной задаче (1) $g_1 = 0$, что не вносит ограничений на проводимые рассуждения. Тогда решение задачи (1) в рассматриваемой подобласти на густой подсетке $\Omega_{h,0}^{(1)}$ можно записать в виде суммы $\sum_{\nu \in \omega_{h,0}} \sigma_h^{(\nu)} v_{h,\nu}$, а на редкой подсетке $\Omega_{h,k}^{(1)}$ — как $\sum_{\nu' \in \omega_{h,\nu'}} \psi_h^{(\nu',k)} v_{h,\nu'}$, где номера ν' относятся к редкой подсетке $\omega_{h,\nu'}$ (по построению подсеток они совпадают с подмножеством номеров густой подсетки $\omega_{h,0}$). Исключим из первой суммы величины $v_{h,\nu''}$ в несовпадающих узлах ν'' при помощи интерполяции, построенной по совпадающим узлам ν' . Тогда, сравнивая получившееся выражение со второй суммой, найдем величины $\psi_h^{(\nu',k)}$ в совпадающих узлах ν' .

Из соображений симметрии базовые подзадачи (6) решаются лишь для узлов, лежащих на половинах двух смежных сторон подобласти R_0 .

Вспомогательные подзадачи решаются, вообще говоря, для всех граничных подобластей $G_k^{(2)}$, относительно функций $\psi_h^{(j,k)}$. В этом случае имеются исключения, приводящие к уменьшению количества подзадач, которые даны ниже, в разд. 2.

В узлах сетки ω_h вычисляются суммы разностных аналогов $f_{h,i}^{(j)}$ нормальных производных от вспомогательных решений по формуле

$$f_{h,i}^{(j)} = \left(d_h^{(+)} \psi_h^{(j,+)} \right)_i + \left(d_h^{(-)} \psi_h^{(j,-)} \right)_i, \quad i = \overline{1, n}, \quad j = \overline{0, n}, \quad (7)$$

где под нормалью понимается внутренняя нормаль η_\pm к границе сопряжения (рис. 2); $d_h^{(+)}, d_h^{(-)}$ — линейные операторы, определяющие конкретные схемы; индексы “+” “−” обозначают операторы и функции по разные стороны от границы сопряжения. Вычисленные величины выражаются через компоненты b_i вектора b и элементы $a_{i,j}$ матрицы A как

$$f_{h,i}^{(j)} = \begin{cases} b_i, & j = 0, \\ a_{i,j}, & j > 0, \end{cases} \quad i = \overline{1, n}. \quad (8)$$

Диагональный элемент $a_{i,i}$ соответствует i -му узлу сетки ω_h . Пусть он лежит на отрезке 2 границы сопряжения двух внутренних подобластей $G_+^{(1)}, G_-^{(1)}$ (рис. 2, а). Каждая

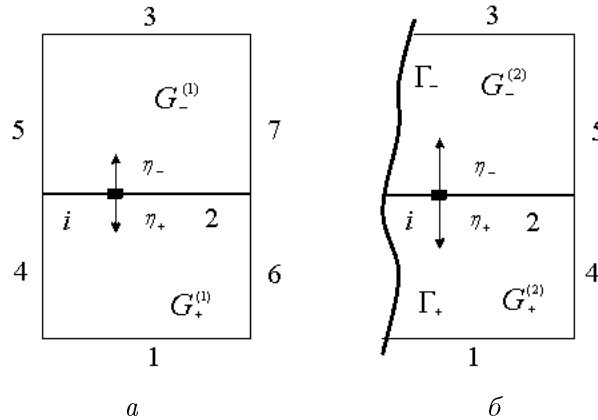


Рис. 2. К расчету элементов матрицы A : a — внутренние области; b — граничные подобласти

строка матрицы A , в том числе и i -я строка, содержит n элементов, соответствующих всем узлам сетки ω_h ; отличными от нуля в i -й строке будут только элементы, соответствующие узлам, лежащим на отрезках 1–7 границ подобластей $G_+^{(1)}$, $G_-^{(1)}$. Аналогичная ситуация имеет место и для граничных подобластей $G_+^{(2)}$, $G_-^{(2)}$ (рис. 2, б). В этом случае отличными от нуля в i -й строке будут только элементы, соответствующие узлам, лежащим на отрезках 1–5 границ подобластей $G_+^{(2)}$, $G_-^{(2)}$ (узлы на кусках Γ_+ , Γ_- при формировании матрицы A не учитываются).

2. Оценки эффективности алгоритма

Под эффективностью алгоритма будем понимать трудоемкость и объем запоминаемой информации, необходимые для его реализации.

2.1. Трудоемкость алгоритма

Наибольший объем вычислений при формировании СЛАУ (3) с квадратной матрицей A порядка n занимает решение вспомогательных подзадач (4)–(6) в подобластях G_k , на которые разделяется расчетная область макросеткой Ω_H .

Обозначим через K_A число подзадач, решаемых при расчете элементов матрицы A , равное

$$K_A = K_A^{(1)} + K_A^{(2)},$$

где $K_A^{(1)}$, $K_A^{(2)}$ — количества подзадач, решаемых соответственно во внутренних $G_k^{(1)}$ и в граничных $G_k^{(2)}$ подобластях. Верны оценки

$$K_A^{(1)} = \frac{n_{x,m} + n_{y,m}}{2}, \quad K_A^{(2)} < 2K_2' (n_{x,m} + n_{y,m} - 2).$$

Здесь $n_{x,m} = \max_k n_{x,k}$, $n_{y,m} = \max_k n_{y,k}$, где $n_{x,k}$, $n_{y,k}$ — числа интервалов подсеток $\Omega_{h,k}$, введенных выше, а K_2' — число граничных подобластей, для которых решаются вспомогательные задачи, причем $K_2' \leq K_2$. Знак неравенства здесь обусловлен следующими соображениями. Пусть Γ' — отрезок границы Γ , параллельный одной из координатных осей, причем на всем отрезке Γ' задано краевое условие одного вида, G_k' — граничные подобласти, внешняя граница которых содержит только куски Γ' . По аналогии

с внутренними подобластями в этом случае нет необходимости решать вспомогательные подзадачи (5) для всех G'_k , а достаточно это сделать лишь для одной подобласти на самой густой сетке.

Пусть K_b — число подзадач (4) и (5), необходимых для вычисления вектора b . Ясно, что $K_b \leq K$, так как в данном случае решаются подзадачи в граничных подобластях и только в тех внутренних подобластях, в которых функция g_1 отлична от нуля. Например, если в (1) речь идет об уравнении Лапласа, то во всех внутренних подобластях решение всех подзадач (4) при $j = 0$ будет нулевым.

Общее число $K_{A,b}$ подзадач, решаемых для формирования системы (3), равно

$$K_{A,b} = K_A^{(1)} + K_A^{(2)} + K_b.$$

Подзадачи во внутренних подобластях могут быть решены оптимальным по быстрдействию прямым методом циклической редукции [2]. Его основу составляют прогоночки вдоль координатных линий. Так как число узлов подсеток $\Omega_{h,k}$ сравнительно невелико, прогоночные коэффициенты можно вычислить и запомнить один раз для всех подсеток. Подзадачи в граничных подобластях, видимо, быстрее всего решать итерационными методами, выбирая в качестве начального приближения для очередной подзадачи решение предыдущей, но обоснованный ответ на этот вопрос требует дальнейших исследований.

Решение и той и другой групп задач вследствие их малой размерности проводится с привлечением быстрой кэш-памяти компьютера, что также значительно ускоряет вычисления. Пусть объем кэш-памяти составляет L Мбайт, а каждое вещественное число занимает в памяти компьютера 8 байт. Обозначим через M максимальное число узлов подсетки, равное $M = (n_{x,m} - 1)(n_{y,m} - 1)$. Реализация какого-либо алгоритма решения краевых подзадач в подобластях требует хранения двух массивов размерности M , отводимых под искомую функцию и правую часть сеточных уравнений, а также хранения вспомогательных величин объемом, который мы выразим как sM , где s — вещественный коэффициент, зависящий от конкретного алгоритма. Для эффективного использования кэш-памяти необходимо, чтобы в нее вмещались все величины, участвующие в работе рассматриваемого алгоритма. Тогда, учитывая только вещественные числа, для размера подсетки можно записать оценку

$$M \approx \frac{L2^{20}}{8(s+2)} = \frac{L2^{17}}{s+2}. \quad (9)$$

Приведем некоторое количественное выражение данной оценки. Предположим, что $M = 63 \times 63$. В этом случае, если даже макросетка Ω_H содержит небольшое число разбиений (например, 16×16), то результирующая сетка Ω_h имеет около 1 млн узлов, что представляется достаточным для решения большинства практических краевых задач. Подставляя приведенные числа в формулу (9), получим выражение для объема кэш-памяти под хранение вспомогательных величин $s \approx 32L - 2$. Например, новосибирский кластерный суперкомпьютер НКС-160 имеет кэш-память 3 Мбайта, и, следовательно, при расчетах на нем $s \approx 94$, что вполне достаточно для решателей, включая метод циклической редукции с предварительным расчетом прогоночных коэффициентов, который используется в настоящей работе.

В каждой строке матрицы A , имеющей диагональный элемент, соответствующий узлу, лежащему на x -линии, будут отличны от нуля в общем случае $q_x = 3(n_{x,m} - 1) + 4(n_{y,m} - 1)$ элементов, а для y -линии это число равно $q_y = 3(n_{y,m} - 1) + 4(n_{x,m} - 1)$.

Порядок n матрицы A равен числу узлов сетки ω_h , и он намного меньше числа узлов результирующей сетки Ω_h .

На последнем этапе предлагаемого алгоритма находится окончательное решение в подобластях. При этом нет необходимости решать подзадачи во всех K подобластях, на которые разбивается расчетная область G , это достаточно выполнить лишь для подобластей, представляющих интерес для последующего анализа и обработки.

Конкретизируем полученные оценки на примере решения модельной задачи Дирихле для уравнения Пуассона в единичном квадрате. Будем считать граничными все подобласти, примыкающие к границе квадрата. Предположим, что а) $N_x = N_y = N_H$, б) $n_{x,k} = n_{y,k} = m$, в) вспомогательные подзадачи решаются для каждой граничной подобласти ($K'_2 = K_2 = 4(N_H - 1)$) (напомним, что N_x, N_y — числа интервалов макросетки Ω_H по x, y). Тогда имеем

$$K_{A,b} = m + 16(N_H - 1)(m - 1) + N_H^2.$$

Сравним эту величину с количеством K_I подзадач в том случае, если бы решение во всей области искалось методом итераций по подобластям. Предположим для этой оценки, что $m \approx N_H$. Так как $K_I = N_I N_H^2$, где N_I — число итераций, и $K_{A,b} < 17N_H^2$, то $K_{AB} \approx K_I$ при $N_I < 17$, т.е. формирование системы (3) требует решения меньшего количества подзадач, чем это делается при проведении 17 итераций по подобластям. Отметим, что это число уменьшается при $m < N_H$, а также то, что вспомогательные подзадачи имеют специальный вид, позволяющий решать их по сверхбыстрым алгоритмам.

Порядок n матрицы A в данном случае равен

$$n = 2N_H(N_H - 1)(m - 1).$$

Сравним эту величину с числом внутренних узлов $Q_h = (N_h - 1)(N_h - 1)$ результирующей сетки Ω_h , где N_h — число интервалов данной сетки в одном направлении. Тогда $m = \frac{N_h}{N_H}$ и в предположении, что $N_h \gg N_H \gg 1$, получим

$$\frac{Q_h}{n} \approx \frac{N_h}{2N_H} = \frac{m}{2}.$$

2.2. Объем запоминаемой информации

Наибольший объем оперативной памяти компьютера отводится под хранение элементов $a_{i,j}$ матрицы A , вычисляемых по формулам (7) и (8) в узлах сетки ω_h . Способ хранения и доступа к коэффициентам зависит от вида подобластей, на границе сопряжения которых лежит рассматриваемый i -й узел.

Для внутренних подобластей достаточно сохранить величины $f_{h,i}^j$, рассчитанные по формуле (7) на двух сторонах одной внутренней подобласти G_0 . В этом случае хранится массив, состоящий из Q_1 элементов, где Q_1 выражается как

$$Q_1 = (n_{x,m} - 1) [3(n_{x,m} - 1) + 4(n_{y,m} - 1)] + (n_{y,m} - 1) [3(n_{y,m} - 1) + 4(n_{x,m} - 1)].$$

Эти величины используются для формирования элементов матрицы A в строках, соответствующих узлам сетки ω_h , лежащим на границе сопряжения внутренних подобластей, согласно рассмотренному выше принципу.

Пусть теперь γ'_k — отрезок границы сопряжения двух подобластей, из которых хотя бы одна является граничной ($k = \overline{1, K_\gamma^{(2)}}$, где $K_\gamma^{(2)}$ — суммарное число отрезков в K_2 граничных подобластях). В этом случае рассчитываются и запоминаются коэффициенты для каждого узла $T_i \in \gamma'_k$ и для каждого отрезка γ'_k . Число Q_2 хранимых величин зависит от конфигурации границы расчетной области и удовлетворяет неравенству

$$Q_2 < 7K_\gamma^{(2)} (m - 1)^2,$$

где $m = \max(n_{x,m}, n_{y,m})$. В общем случае $Q_2 \gg Q_1$. Так же, как и число вспомогательных задач, Q_2 может быть уменьшено за счет подобластей, примыкающих к отрезкам границы Γ , параллельным одной из координатных осей.

Приведем оценки величины Q_2 для модельной задачи. Число отрезков γ'_k в ней равно $K_\gamma^{(2)} = 4(2N_H - 3)$. Тогда справедливо равенство

$$Q_2 = 28(2N_H - 3)(m - 1)^2.$$

Сравнивая эту величину с числом внутренних узлов Q_h результирующей сетки Ω_h , получим оценку

$$\frac{Q_2}{Q_h} < \frac{56}{N_H}.$$

3. Численные эксперименты

Численные эксперименты ставились с целью показать работоспособность предложенного в [1] подхода на примере решения тестовой задачи, которой служила задача Дирихле для уравнения Лапласа:

$$\Delta u = 0, \quad u|_\Gamma = g,$$

в квадрате $\bar{G} = \{0.1 \leq x \leq 0.6, 0 \leq y \leq 0.5\}$. Данная задача представляла собой фрагмент более общей задачи о решении уравнения Лапласа в области, ограниченной двумя концентрическими окружностями с радиусами $R_1 = 0.1$, $R_2 = 1$, на которых задаются значения искомой функции \bar{u} , равные соответственно 0, 1. Последняя задача имеет точное решение

$$\bar{u} = \ln \frac{r}{R_1} / \ln \frac{R_2}{R_1},$$

где r — радиус-вектор. Функция g выбиралась, как $g = \bar{u}|_\Gamma$, и тогда точное решение исходной задачи есть $u = \bar{u}$ в G .

Расчетная область покрывалась квадратной сеткой

$$\Omega_h = \{x = x_i, y = y_j, i, j = \overline{1, N_h - 1}\}$$

с шагом $h = 0.5/N_h$ и разбивалась линиями $x = X_I$, $y = Y_J$ ($I, J = \overline{1, N_h - 1}$), совпадающими с линиями сетки Ω_h и представляющими собой линии макросетки Ω_H , на $N_G = N_H^2$ прямоугольных подобластей G_k , где $N_h, N_H > 1$ — заданы и представляют собой соответственно числа интервалов сеток Ω_h и Ω_H в одном направлении. В качестве операторов $\Delta_h^{(k)}$ выбирались обычные пятиточечные разностные операторы Лапласа [2]. Оператором d_h , аппроксимирующим нормальные производные, служила трехточечная схема второго порядка, которая для соседних узлов, обозначенных цифрами 0, 1, 2, выглядит как

$$(d_h u_h)_0 = \frac{-3(u_h)_0 + 4(u_h)_1 - (u_h)_2}{2h}.$$

Относительные погрешности расчетов, %

N_H	N_h					
	16	32	64	128	256	512
2	$1.6 \cdot 10^{-1}$	$8.5 \cdot 10^{-2}$	$2.9 \cdot 10^{-2}$	$8.1 \cdot 10^{-3}$	—	—
4	$2.0 \cdot 10^{-1}$	$8.7 \cdot 10^{-2}$	$2.9 \cdot 10^{-2}$	$8.1 \cdot 10^{-3}$	$2.0 \cdot 10^{-3}$	
8		$8.5 \cdot 10^{-2}$	$2.9 \cdot 10^{-2}$	$8.1 \cdot 10^{-3}$	$2.0 \cdot 10^{-3}$	$5.0 \cdot 10^{-4}$
16			$1.7 \cdot 10^{-2}$	$7.0 \cdot 10^{-3}$	$2.0 \cdot 10^{-3}$	$4.8 \cdot 10^{-4}$
32				$4.4 \cdot 10^{-3}$	$1.4 \cdot 10^{-3}$	$4.0 \cdot 10^{-4}$
64					$1.0 \cdot 10^{-3}$	$2.3 \cdot 10^{-4}$

Расчеты проводились при различных параметрах сеток Ω_h , Ω_H : $N_h = N_{h,k} = 2^{k+3}$; $N_H = N_{H,k} = 2^{k+1}$, $k = \overline{1,6}$. Краевые подзадачи решались методом циклической редукции с предварительным расчетом прогоночных коэффициентов для числа интервалов $m = \frac{N_h}{N_H}$ подсеток, лежащего в пределах $4 \leq m \leq 64$, т.е. для некоторых сочетаний N_h , N_H расчеты не проводились. В таблице даны значения относительной погрешности ε , %, рассчитанной по формуле

$$\varepsilon\% = \max_{i,j} \left| \frac{u_{i,j} - (u_h)_{i,j}}{u_{i,j}} \right| \cdot 100\%,$$

где (i, j) — номера узлов сетки Ω_h .

Результаты данной таблицы при больших N_h свидетельствуют о том, что предложенный алгоритм при данных аппроксимациях имеет второй порядок точности.

Кроме того, рассматриваемая задача решалась на сетках Ω_h по обычной пятиточечной схеме без разбиения на подобласти. Полученные значения погрешности $\varepsilon\%$ с принятой точностью совпали с результатами таблицы при $N_H = 2$. Эти факты свидетельствуют о том, что предложенный алгоритм не вносит дополнительных погрешностей в расчеты по сравнению с обычным подходом решения без разбиения на подобласти.

Заключение

Отметим ряд свойств алгоритма, основанного на прямом методе декомпозиции без наложения подобластей для решения краевых задач на прямоугольных квазиструктурированных сетках, предложенного в работе [1].

1. Решение краевых задач ищется на предложенных адаптивных сетках: за счет локальной модификации они учитывают конфигурацию границы, а за счет изменения плотности сеточных узлов в подобластях — неоднородности правой части решаемого уравнения.

2. Работа с данными сетками проводится по простым алгоритмам за счет их структурированного построения. Например, часто встречающаяся в практике вычислительная задача об определении сеточного элемента, которому принадлежит заданная точка T , решается в два этапа: на первом из них по очевидным предельно простым правилам определяется подобласть, в которую попала T , а на втором этапе по аналогичным правилам — сеточный элемент w , лежащий в данной подобласти. Этот алгоритм применим для большинства сеточных элементов. Лишь в том случае, если w содержит хотя бы один модифицированный узел, второй этап требует дальнейшей детализации.

3. Проведенные оценки и численные эксперименты подтвердили работоспособность алгоритма. На примере модельной задачи показано, что для своей реализации он требует решения вспомогательных подзадач, число которых меньше числа подзадач при проведении 17 итераций по подобластям, но при этом в силу специфики вспомогательные подзадачи решаются по быстрым алгоритмам. Объем оперативной памяти компьютера, на котором проводятся вычисления, для хранения необходимой для расчетов информации составляет приемлемую величину. Так, если результирующая сетка Ω_h имеет 1024×1024 интервалов и проводится разбиение на 32×32 подобласти, то дополнительно требуется менее двух массивов, каждый из которых имеет размерность, равную числу узлов Ω_h . При этом матрица A системы линейных алгебраических уравнений для вычисления значений искомой функции на границе сопряжения подобластей имеет порядок, приблизительно в 16 раз меньший порядка матрицы системы разностных уравнений, построенных в узлах Ω_h при решении краевой задачи без разбиения на подобласти. Необходимо отметить, что при этом число ненулевых коэффициентов в одном уравнении может достигать величины $7 \times 31 = 217$.

4. Краевые подзадачи в подобластях решаются независимо друг от друга, т.е. в данной части алгоритм параллельный.

5. Краевые подзадачи во внутренних подобластях могут решаться оптимальным по быстрдействию методом циклической редукции.

6. Блочный характер алгоритма позволяет эффективно использовать сверхбыструю кэш-память компьютера.

7. На заключительном этапе алгоритма нет необходимости в решении краевых подзадач во всех подобластях, это достаточно сделать лишь в тех из них, которые представляют интерес.

8. Алгоритм может быть перенесен с различной степенью усложнения на решение других уравнений помимо уравнения Пуассона в декартовых координатах, рассмотренного в [1]. Так, в краевых задачах для уравнения Пуассона в цилиндрических координатах (r, z) требуется решение базовых вспомогательных подзадач вместо определения одной внутренней подобласти не более чем в N_r подобластях, где N_r — соответственно число интервалов макросетки в направлении оси r .

9. Предлагаемый подход эффективен при решении нестационарных задач и нелинейных задач, так как при этом матрица A формируется один раз до проведения нестационарного процесса или итераций по нелинейности.

10. Наиболее эффективен данный метод при решении краевых задач в областях, границы которых состоят большей частью из кусков, параллельных координатным осям. В частности, пиковая эффективность достигается при решении краевых задач для уравнения Лапласа в областях, составленных из прямоугольников.

Список литературы

- [1] СВЕШНИКОВ В.М. Прямой метод декомпозиции без наложения подобластей для решения краевых задач на прямоугольных квазиструктурированных сетках // Вычисл. технологии. 2008. Т. 13, № 2. С. 106–118.
- [2] САМАРСКИЙ А.А. Теория разностных схем. М.: Наука, 1977.

*Поступила в редакцию 25 апреля 2007 г.,
в переработанном виде — 14 апреля 2008 г.*