

Параллельный алгоритм реализации метода встречных циклических прогонок для двумерных сеточных областей*

Л. В. ЛОГАНОВА¹, Д. Л. ГОЛОВАШКИН²

¹Самарский государственный аэрокосмический университет, Россия

²Институт систем обработки изображений РАН, Самара, Россия

e-mail: tk@smr.ru, dimitriy@smr.ru

Работа посвящена построению параллельного алгоритма, основанного на функциональной декомпозиции метода встречных циклических прогонок с циклическим разбиением сеточной области. Данный алгоритм может быть применен для решения разностных уравнений схем методов расщепления и переменных направлений в случае периодических граничных условий. По сравнению с методами декомпозиции области и циклической редукции предлагаемый алгоритм характеризуется соответственно меньшим объемом арифметических операций и меньшим объемом коммуникаций.

Ключевые слова: параллельные алгоритмы, метод встречных циклических прогонок.

Введение

Применение параллельных вычислений в математическом моделировании открывает новые возможности для исследований физических явлений, позволяя рассматривать изучаемые процессы на более длительных временных интервалах либо на более обширных областях пространства. В частности, с появлением доступных многопроцессорных систем развитие компьютерной оптики [1] получило новый импульс. Так, современный переход к нанооптике обуславливает популярность Finite-Difference Time-Domain (FDTD) метода моделирования распространения электромагнитного излучения в рамках строгой теории дифракции. Известная монография Yu Wenhua и др. [2] подытоживает многолетнюю практику составления параллельных алгоритмов FDTD-метода. Однако в ней остались не рассмотренными важные разностные схемы Zheng, Chen, Zhang [3] и параллельные алгоритмы для этих схем.

Для восполнения данного пробела авторы настоящей работы рассматривают построение параллельного алгоритма реализации метода встречных циклических прогонок на двумерных сеточных областях. Предлагаемый алгоритм предназначен для решения разностных уравнений схем расщепления и переменных направлений с периодическими граничными условиями (например, из [3]). В оптике такие условия ставятся при рассмотрении задачи дифракции на периодических структурах — дифракционных решетках и фотонных кристаллах.

*Работа выполнена при финансовой поддержке РФФИ (гранты 10-07-00453-а и 10-07-00553-а).

Кроме того данный алгоритм рекомендуется использовать при изучении других задач вычислительной физики с периодическими краевыми условиями, решение которых приводит к системам линейных алгебраических уравнений (СЛАУ) ленточного вида.

Параллельный алгоритм основан на методе циклических прогонок [4] и является развитием подходов, предложенных в работах [5, 6] для метода встречных прогонок. Известны и другие прямые методы решения подобных СЛАУ на многопроцессорных системах: декомпозиции области и циклической редукции. Однако, как показано в [7], они менее эффективны, чем алгоритм встречных прогонок, либо по числу арифметических операций (декомпозиция области), либо по объему коммуникаций (циклическая редукция).

1. Метод встречных циклических прогонок для двумерных сеточных областей

Положим, что для решения некоторой задачи математической физики (например, теплопроводности или электродинамики) используется двумерная сеточная область $w_h = w_{h_1}^1 \times w_{h_2}^2$, где $w_{h_1}^1 = \{x_1(i) = ih_1, i = 0, 1, \dots, M - 1\}$, $w_{h_2}^2 = \{x_2(j) = jh_2, j = 0, 1, \dots, N - 1\}$. На w_h задана сеточная функция $g_{ij} = g(ih_1, jh_2)$. После составления некоторой разностной схемы (либо расщепления, либо переменных направлений [3, 8]) необходимо найти решение совокупности СЛАУ: каждой строке и столбцу сеточной области w_h соответствует одна система ленточного вида. Пример — разностная краевая задача из [4]

$$\begin{aligned} -a_0 y_{M-1} + c_0 y_0 - b_0 y_1 &= f_0, \quad i = 0, \\ -a_i y_{i-1} + c_i y_i - b_i y_{i+1} &= f_i, \quad 1 \leq i \leq M - 1, \\ y_M &= y_0. \end{aligned}$$

В целом решение всех разностных уравнений схемы (на одном временном слое, если схема нестационарная) связано с решением M уравнений (1) по N строкам w_h и N уравнений (2) по M столбцам:

$$A_k y_k = b_k, \tag{1}$$

где $A_k \in R^{M \times M}$, $y_k, b_k \in R^M$, $y_k = (g_{k,0}, \dots, g_{k,M-1})$, $k = \overline{0 : M - 1}$, $a_{ij} \neq 0$ для $i \leq j + 1$, $j \leq i + 1$, а также $a_{1M} \neq 0$, $a_{M1} \neq 0$,

$$A_l y_l = b_l, \tag{2}$$

где $A_l \in R^{N \times N}$, $y_l, b_l \in R^N$, $y_l = (g_{0,l}, \dots, g_{N-1,l})^T$, $l = \overline{0 : M - 1}$, $a_{ij} \neq 0$ для $i \leq j + 1$, $j \leq i + 1$, а также $a_{1N} \neq 0$, $a_{N1} \neq 0$.

Для решения СЛАУ (1) запишем метод циклических встречных прогонок. Как и в методе циклической прогонки [4], представим решение в виде линейной комбинации сеточных функций

$$y_i = u_i + y_0 v_i, \quad 0 \leq i \leq M - 1. \tag{3}$$

Для отыскания u_i , v_i обратимся к методу встречных прогонок [4]. Тогда

$$\alpha_2 = \frac{a_{23}}{a_{22}}, \quad \beta_2 = \frac{b_1}{a_{22}}, \quad \gamma_2 = \frac{a_{21}}{a_{22}}, \tag{4}$$

$$\alpha_{i+1} = \frac{a_{(i+1)(i+2)}}{a_{(i+1)(i+1)} - a_{(i+1)i}\alpha_i}, \quad \beta_{i+1} = \frac{b_i + a_{(i+1)i}\beta_i}{a_{(i+1)(i+1)} - a_{(i+1)i}\alpha_i},$$

$$\gamma_{i+1} = \frac{a_{(i+1)i}\gamma_i}{a_{(i+1)(i+1)} - a_{(i+1)i}\alpha_i}, \quad i = \overline{2, m-1}, \quad (5)$$

$$\zeta_{M-1} = \frac{a_{M(M-1)}}{a_{MM}}, \quad \eta_{M-1} = \frac{b_{M-1}}{a_{MM}}, \quad \psi_{M-1} = \frac{a_M 1}{a_{MM}}, \quad (6)$$

$$\zeta_{i-1} = \frac{a_{i(i-1)}}{a_{ii} - a_{i(i+1)}\zeta_i}, \quad \eta_{i-1} = \frac{b_{i-1} + a_{i(i+1)}\eta_i}{a_{ii} - a_{i(i+1)}\zeta_i},$$

$$\psi_{i-1} = \frac{a_{i(i+1)}\psi_i}{a_{ii} - a_{i(i+1)}\zeta_i}, \quad i = \overline{M-1, m} \quad (7)$$

$$u_i = \alpha_{i+1}u_{i+1} + \beta_{i+1}, \quad v_i = \alpha_{i+1}v_{i+1} + \gamma_{i+1}, \quad i = \overline{m-1, 1}, \quad (8)$$

$$u_{i+1} = \zeta_{i+1}u_i + \eta_{i+1}, \quad v_{i+1} = \zeta_{i+1}v_i + \psi_{i+1},$$

$$i = \overline{m, M-2}, \quad v_M = 1, \quad u_M = 0, \quad u_0 = 0, \quad v_0 = 1, \quad (9)$$

$$y_0 = \frac{b_0 + a_{1M}u_{M-1} + a_{12}u_1}{a_{11} - a_{1M}v_{M-1} - a_{12}v_1},$$

где $\alpha, \beta, \gamma, \zeta, \eta, \psi$ — прогоночные коэффициенты, v, u — промежуточные сеточные функции. Значения искомой сеточной функции y_i найдем по формуле (3).

СЛАУ (2) решается аналогично.

2. Параллельный алгоритм для четырехпроцессорной вычислительной системы

Приступая к синтезу четырехпроцессорного параллельного алгоритма, проведем двумерное разбиение сеточной области w_h на четыре подобласти (рис. 1) $w_h^1, w_h^2, w_h^3, w_h^4$ (где w_h^i принадлежит i -му процессору). На первом этапе алгоритма процессоры 1, 3 выполняют встречные прогоночки для строк с 1 по $N/2$, процессоры 2, 4 — для строк с $N/2+1$ по N , производя вычисления по (4)–(7) и обмениваясь между собой прогоночными коэффициентами (рис. 1, а) (например, процессор 1 производит обмен с процессором 3, процессор 2 — с процессором 4). Обратный ход сопровождается определением промежуточных значений сеточных функций для тех же строк по формулам (8), (9) и обменами этими значениями (рис. 1, б). В завершение процессоры 1 и 2, получив граничные значения промежуточных сеточных функций от процессоров 3, 4 соответственно, вычисляют y_0 и пересылают его значение процессорам 3, 4. Нахождение значений искомых сеточных функций согласно (3) выполняют все процессоры одновременно (рис. 1, в). На втором этапе процессоры 1, 2 и 3, 4 осуществляют встречные прогоночки для столбцов с 1 по $M/2$ и с $M/2+1$ по M , вычисляя значения прогоночных коэффициентов по (4)–(7) и обмениваясь ими (рис. 1, г). Например, процессор 1 производит обмен с процессором 2, процессор 3 — с процессором 4. Во время выполнения обратного хода определяют значения промежуточных сеточных функций согласно (8), (9) для тех же столбцов (рис. 1, д). После этого процессоры 1, 3, получив граничные значения промежуточных сеточных функций от процессоров 2, 4 соответственно, вычисляют y_0 и пересылают его значение процессорам 2, 4. Выполняя прямой ход, согласно (3) процессоры определяют решение (2) (рис. 1, е).

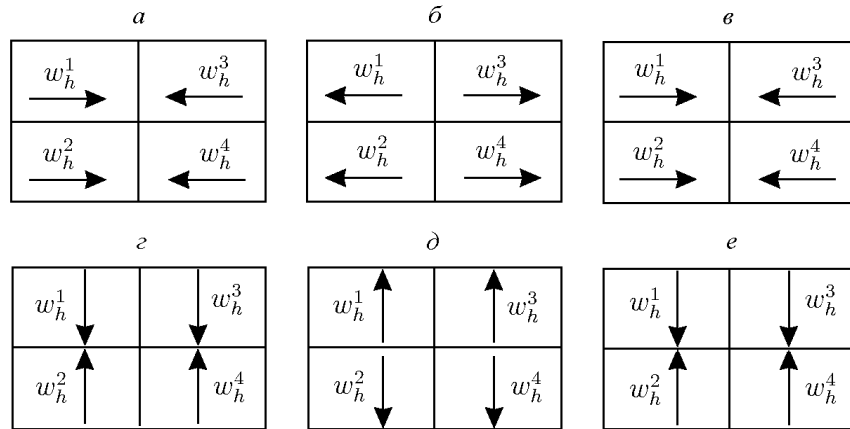


Рис. 1. Этапы вычислений по параллельному алгоритму, реализующему метод встречных циклических прогонок на двумерной сеточной области для четырех процессоров: *a* — прямой ход по строкам, *б* — обратный ход по строкам, *в* — определение значений искомых сеточных функций, *г* — прямой ход по столбцам, *д* — обратный ход по столбцам, *е* — определение значений искомых сеточных функций

Поскольку все процессоры производят вычисления одновременно, удастся избежать простоев, связанных с ожиданием завершения арифметических действий другими процессорами, неизбежных при линейной декомпозиции [9]. Здесь и далее не рассматриваются временные затраты на организацию коммуникаций (время латенции, длительность пересылки сообщений), которые определяются особенностями конкретных вычислительных комплексов и не характеризуют собственно алгоритм.

3. Параллельный алгоритм с циклическим разбиением

Распространим идею алгоритма с четырех процессоров на P . Описание алгоритма будет сопровождаться изложением общего случая (P процессоров) и примером для 16 процессоров. Выполним циклическое разбиение двумерной области w_h на $P^2/4$ подобласти (64 для 16 процессоров), где $P = 2^q$, $q \in N$ (рис. 2). Процессор p ($0 \leq p \leq P$) на i -м шаге будет производить вычисления для подобластей $w_h^p(i)$.

$w_h^0(1)$	$w_h^3(2)$	$w_h^2(3)$	$w_h^1(4)$	$w_h^9(4)$	$w_h^{10}(3)$	$w_h^{11}(2)$	$w_h^8(1)$
$w_h^1(1)$	$w_h^0(2)$	$w_h^3(3)$	$w_h^2(4)$	$w_h^{10}(4)$	$w_h^{11}(3)$	$w_h^8(2)$	$w_h^9(1)$
$w_h^2(1)$	$w_h^1(2)$	$w_h^0(3)$	$w_h^3(4)$	$w_h^{11}(4)$	$w_h^8(3)$	$w_h^9(2)$	$w_h^{10}(1)$
$w_h^3(1)$	$w_h^2(2)$	$w_h^1(3)$	$w_h^0(4)$	$w_h^8(4)$	$w_h^9(3)$	$w_h^{10}(2)$	$w_h^{11}(1)$
$w_h^4(1)$	$w_h^5(2)$	$w_h^6(3)$	$w_h^7(4)$	$w_h^{15}(4)$	$w_h^{14}(3)$	$w_h^{13}(2)$	$w_h^{12}(1)$
$w_h^5(1)$	$w_h^6(2)$	$w_h^7(3)$	$w_h^4(4)$	$w_h^{12}(4)$	$w_h^{15}(3)$	$w_h^{14}(2)$	$w_h^{13}(1)$
$w_h^6(1)$	$w_h^7(2)$	$w_h^4(3)$	$w_h^5(4)$	$w_h^{13}(4)$	$w_h^{12}(3)$	$w_h^{15}(2)$	$w_h^{14}(1)$
$w_h^7(1)$	$w_h^4(2)$	$w_h^5(3)$	$w_h^6(4)$	$w_h^{14}(4)$	$w_h^{13}(3)$	$w_h^{12}(2)$	$w_h^{15}(1)$

Рис. 2. Циклическое разбиение двумерной сеточной области для шестнадцатипроцессорного параллельного алгоритма, реализующего метод встречных циклических прогонок

Любой процессор будет выполнять вычисления для $P/4$ прямоугольных подобластей, и никакие две подобласти, вычисления для которых выполняются на одном процессоре, не содержат одну и ту же строку или столбец сеточной области. Данная особенность позволяет производить арифметические действия всем процессорам одновременно.

Шаги первого этапа алгоритма, связанного с циклическими прогонами по строкам сеточной области, приведены в табл. 1.

На i -м шаге (см. табл. 1) процессоры (с номерами из столбца 2) производят вычисления по алгоритму (столбец 4) над подобластью, указанной в столбце 3. В столбце 5 определены коммуникации, которые осуществляет каждый процессор. Описание коммуникаций выполнено в нотации Дж. Голуба, предложенной в фундаментальной работе [10]. Так, запись вида if $p = 0$ then send $(\alpha, \beta, \gamma, P/4 - 1)$ (столбец 5, шаг 1, процессор $0 \leq p \leq P/4 - 1$) означает, что 0-й процессор посылает значения α, β, γ , определенные

Т а б л и ц а 1. Шаги первого этапа алгоритма

Шаг (i)	Процессоры	Подобласть	Вычисляемые величины	Коммуникации
1	2	3	4	5
1	$0 \leq p \leq P/4 - 1$	$w_h^p(1)$	α, β, γ	if $p = 0$ then send $(\alpha, \beta, \gamma, P/4 - 1)$ else send $(\alpha, \beta, \gamma, p - 1)$
	$P/4 \leq p \leq P/2 - 1$	$w_h^p(1)$	α, β, γ	if $p = P/2 - 1$ then send $(\alpha, \beta, \gamma, P/4)$ else send $(\alpha, \beta, \gamma, p + 1)$
	$P/2 \leq p \leq 3 \cdot P/4 - 1$	$w_h^p(1)$	ζ, ξ, φ	if $p = P/2$ then send $(\zeta, \xi, \varphi, 3 \cdot P/4 - 1)$ else send $(\zeta, \xi, \varphi, p - 1)$
	$3 \cdot P/4 \leq p \leq P - 1$	$w_h^p(1)$	ζ, ξ, φ	if $p = P - 1$ then send $(\zeta, \xi, \varphi, 3 \cdot P/4)$ else send $(\zeta, \xi, \varphi, p + 1)$
С 2 по $P/4 - 1$	$0 \leq p \leq P/4 - 1$	$w_h^p(i)$	α, β, γ	if $p = P/4 - 1$ then recv $(\alpha, \beta, \gamma, 0)$ else recv $(\alpha, \beta, \gamma, p + 1)$
		$w_h^p(i)$	α, β, γ	if $p = 0$ then send $(\alpha, \beta, \gamma, P/4 - 1)$ else send $(\alpha, \beta, \gamma, p - 1)$
	$P/4 \leq p \leq P/2 - 1$	$w_h^p(i)$	α, β, γ	if $p = P/4$ then recv $(\alpha, \beta, \gamma, P/2 - 1)$ else recv $(\alpha, \beta, \gamma, p - 1)$
		$w_h^p(i)$	α, β, γ	if $p = P/2 - 1$ then send $(\alpha, \beta, \gamma, P/4)$ else send $(\alpha, \beta, \gamma, p + 1)$
	$P/2 \leq p \leq 3 \cdot P/4 - 1$	$w_h^p(i)$	α, β, γ	if $p = 3 \cdot P/4 - 1$ then recv $(\zeta, \xi, \varphi, P/2)$ else recv $(\zeta, \xi, \varphi, p + 1)$
		$w_h^p(i)$	ζ, ξ, φ	if $p = P/2$ then send $(\zeta, \xi, \varphi, 3 \cdot P/4 - 1)$ else send $(\zeta, \xi, \varphi, p - 1)$
	$3 \cdot P/4 \leq p \leq P - 1$	$w_h^p(i)$	ζ, ξ, φ	if $p = 3 \cdot P/4$ then recv $(\zeta, \xi, \varphi, P - 1)$ else recv $(\zeta, \xi, \varphi, p - 1)$
		$w_h^p(i)$	ζ, ξ, φ	if $p = P - 1$ then send $(\zeta, \xi, \varphi, 3 \cdot P/4)$ else send $(\zeta, \xi, \varphi, p + 1)$
$P/4$	$0 \leq p \leq P/2 - 1$	$w_h^p(P/4)$	α, β, γ	send $(\alpha, \beta, \gamma, p + P/2)$ recv $(\zeta, \xi, \varphi, p + P/2)$
	$P/2 \leq p \leq P - 1$	$w_h^p(P/4)$	ζ, ξ, φ	send $(\zeta, \xi, \varphi, p - P/2)$ recv $(\alpha, \beta, \gamma, p - P/2)$

1	2	3	4	5
С $P/4 + 1$ по $P/2 - 1$	$0 \leq p \leq P/4 - 1$	$w_h^p(i)$ $w_h^p(i)$	u, v u, v	if $p = 0$ then recv $(u, v, P/4 - 1)$ recv $(u, v, p - 1)$ if $p = P/4 - 1$ then send $(u, v, 0)$ else send $(u, v, p + 1)$
	$P/4 \leq p \leq P/2 - 1$	$w_h^p(i)$ $w_h^p(i)$	u, v u, v	if $p = P/2 - 1$ then recv $(u, v, P/4)$ else recv $(u, v, p + 1)$ if $p = P/4$ then send $(u, v, P/2 - 1)$ else send $(u, v, p - 1)$
	$P/2 \leq p \leq 3 \cdot P/4 - 1$	$w_h^p(i)$ $w_h^p(i)$	u, v u, v	if $p = P/2$ then recv $(u, v, 3 \cdot P/4 - 1)$ else recv $(u, v, p - 1)$ if $p = 3 \cdot P/4 - 1$ then send $(u, v, P/2)$ else send $(u, v, p + 1)$
	$3 \cdot P/4 \leq p \leq P - 1$	$w_h^p(i)$ $w_h^p(i)$	u, v u, v	if $p = P - 1$ then send $(u, v, 3 \cdot P/4)$ else send $(u, v, p + 1)$ if $p = 3 \cdot P/4$ then send $(u, v, P - 1)$ else send $(u, v, p - 1)$
$P/2$	$0 \leq p \leq P/2 - 1$	$w_h^p(P/2)$	u, v y_0	recv $(u, v, p + P/2)$ send $(y_0, p + P/2)$
	$P/2 \leq p \leq P - 1$	$w_h^p(P/2)$	u, v	send $(u, v, p - P/2)$ recv $(y_0, v, p - P/2)$
С $P/2 + 1$ по $3 \cdot P/4$	$0 \leq p \leq P - 1$	$w_h^p(i)$	y	

по формулам (4), (5), процессору $P/4 - 1$, остальные процессоры указанного диапазона посылают значения α, β, γ процессорам с номерами на единицу меньше своих номеров.

На втором этапе аналогично выполняются встречные циклические прогонки по столбцам.

Для произвольного P каждый процессор проводит обмены не более чем с четырьмя соседями. Для нашего примера процессор 0 во время выполнения прогонок по строкам совершает обмены с процессорами 2 и 8, а прогонки по столбцам сопровождаются обменами с процессорами 1 и 4. Таким образом, топология коммуникаций предложенного алгоритма — тор.

В табл. 2 приведены сравнительные характеристики параллельных алгоритмов встречных циклических прогонок с линейным [9] и циклическим (из настоящей работы) разбиением сеточной области. При этом количество арифметических действий по линейному алгоритму составляет $C_1 \cdot M$, длительность выполнения арифметической операции — τ_a , длительность одной коммуникации между процессорами — τ_k . Приведен-

Т а б л и ц а 2. Сравнительные характеристики параллельных алгоритмов

Вид разбиения сеточной области	Линейное	Циклическое
Простой	$\frac{C_1 \cdot M \cdot \tau_a}{P} \cdot \left(\frac{P}{2} - 1\right) \cdot N$	Отсутствуют
Коммуникации	$2 \cdot P \cdot N \cdot \tau_k$	$2 \cdot \left(1 - \frac{2}{P}\right) \cdot (N + M) \cdot \tau_k + \frac{3}{2} \cdot P \cdot \tau_k - 2 \cdot \tau_k$

ные характеристики указывают на преимущество алгоритма с циклическим разбиением на квадратных сеточных областях.

4. Результаты вычислительных экспериментов

Для подтверждения эффективности предложенного алгоритма проведены вычислительные эксперименты. Так, при решении классической неявной разностной схемы [4] двумерной задачи теплопроводности методом покоординатного расщепления рассматривалась квадратная сеточная область w_h размером начиная с 4000 и заканчивая 120 000 по каждому измерению. Переход от слоя k к слою $k + 1$ по времени осуществлялся следующим образом:

$$\frac{\bar{U}_{n,m} - U_{n,m}^k}{\tau} - \Lambda_1 \bar{U}_{n,m} = \frac{1}{2} f_{n,m}^k, \quad n = 1, 2, \dots, N - 1, \quad m = 1, 2, \dots, M - 1$$

с циклическими граничными условиями при $n = 0, N$,

$$\frac{U_{n,m}^{k+1} - \bar{U}_{n,m}}{\tau} - \Lambda_2 U_{n,m}^{k+1} = \frac{1}{2} f_{n,m}^{k+1}, \quad n = 1, 2, \dots, N - 1, \quad m = 1, 2, \dots, M - 1$$

с циклическими граничными условиями при $m = 0, M$. Здесь $U_{n,m}^k$ — сеточная функция в узле (nh_1, mh_2) на k -ом временном слое, $\bar{U}_{n,m}$ — изменение U на временном интервале τ за счет теплопроводности только по направлению x . При этом

$$\Lambda_1 U_{n,m}^k = \mu_1 \frac{U_{n+1,m}^k - 2U_{n,m}^k + U_{n-1,m}^k}{h_x^2}, \quad \Lambda_2 U_{n,m}^k = \mu_2 \frac{U_{n,m+1}^k - 2U_{n,m}^k + U_{n,m-1}^k}{h_y^2}.$$

Эксперименты проводились на кластере лаборатории “Математические основы дифракционной оптики и обработки изображений” Самарского государственного аэрокосмического университета им. акад. С.П. Королева. Использовались четыре ЭВМ AMD Sempron(tm) Processor 3000+ (1.8 ГГц), ОЗУ 1 Гб, связанные локальной сетью Ethernet 100 Мб. На компьютерах установлена операционная система Fedora. Алгоритмы реализованы на языке Fortran с использованием библиотеки MPI.

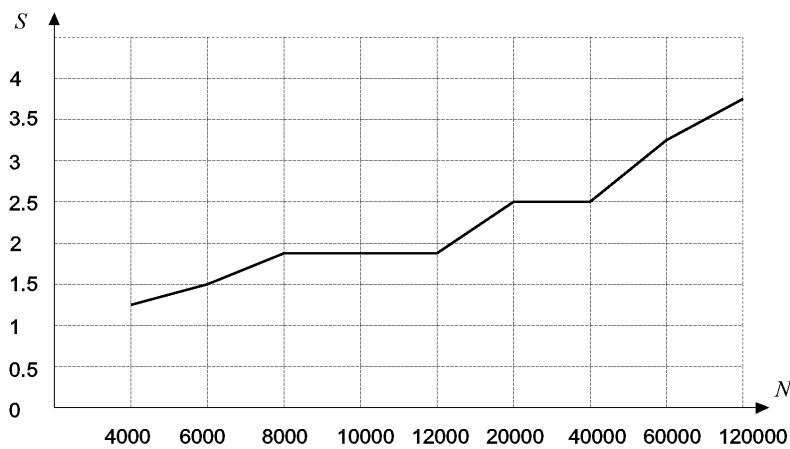


Рис. 3. Зависимость ускорения вычислительного процесса, порожденного четырехпроцессорным алгоритмом, от размерности сеточной области

Результаты исследований позволяют говорить об эффективности предложенного алгоритма. Из графика на рис. 3 видна тенденция к увеличению величины ускорения, достигающего максимальной величины 3.8. Проведенное сравнение с пакетом Pardiso библиотеки Intel[®]MKL [11] показало значительное преимущество разработанного подхода, которое растет с увеличением числа узлов за счет учета ленточной структуры матрицы.

Таким образом, проведенные исследования показали, что применение функциональной декомпозиции к синтезу параллельных алгоритмов, основанных на методе встречных циклических прогонок, для решения сеточных уравнений трехдиагонального вида на двумерных пространственных областях приводит к алгоритмам, имеющим высокие ускорение и эффективность. Представляется целесообразным развитие данного подхода для синтеза векторно-параллельного алгоритма реализации метода.

Список литературы

- [1] СОЙФЕР В. А. Дифракционная компьютерная оптика. М.: Физматлит, 2007. 736 с.
- [2] PARALLEL Finite-Difference Time-Domain Method / Yu Wenhua, Mittra Raj, Su Tao, Liu Yongjun, Y. Xiaoling. Artech House Publ., 2006. 274 p.
- [3] TAFLOVE A., HAGNESS S. Computational Electrodynamics: The Finite-Difference Time-Domain Method: 3rd. ed. Artech House Publ., 2005. 832 p.
- [4] САМАРСКИЙ А.А., НИКОЛАЕВ Е.С. Методы решения сеточных уравнений. М.: Наука, 1978. 561 с.
- [5] МИРЕНКОВ Н.Н. Параллельные алгоритмы для решения задач на однородных вычислительных системах // Вычисл. системы. 1973. № 57. С. 3–32.
- [6] БИРЮКОВА Л.Н., ЧЕТВЕРУШКИН Б.Н. О возможности реализации квазигидродинамической модели полупроводниковой плазмы на многопроцессорных вычислительных системах // Матем. моделирование. 1991. Т. 3, № 6. С. 61–71.
- [7] ГОЛОВАШКИН Д.Л. Параллельные алгоритмы решения сеточных уравнений трехдиагонального вида, основанного на методе встречных прогонок // Там же. 2005. Т. 17, № 11. С. 118–128.
- [8] САМАРСКИЙ А.А., ВАБИЩЕВИЧ П.Н. Вычислительная теплопередача. М.: Едиториал УРСС, 2003. 784 с.
- [9] ЕЛИЗАРОВА Т.Г., ЧЕТВЕРУШКИН Б.Н. Применение многопроцессорных транспьютерных систем для решения задач математической физики // Матем. моделирование. 1992. Т. 4, № 11. С. 75–100.
- [10] ГОЛУВ ДЖ., ВАН ЛОУН Ч. Матричные вычисления М.: Мир, 1999. 548 с.
- [11] КОМПИЛЯТОР Intel[®] Visual Fortran Compiler Professional Edition для ОС Windows [электронный ресурс] // Центр ознакомительного ПО Intel. <http://software.intel.com/ru-ru/articles/intel-software-evaluation-center/> (дата обращения 14.04.10).

*Поступила в редакцию 26 мая 2009 г.,
с доработки — 11 февраля 2011 г.*