

О НЕКОТОРЫХ МЕТОДАХ РЕШЕНИЯ КРАЕВЫХ ЗАДАЧ НА МНОГОПРОЦЕССОРНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ¹

Т.А. Кудряшова, С.В. Поляков

Институт Математического Моделирования РАН

Численное решение краевых задач для обыкновенных дифференциальных уравнений и уравнений в частных производных является одной из центральных проблем математического моделирования. Решению краевых задач посвящено немало учебников и монографий. Однако особенности адаптации широко известных численных алгоритмов к вычислениям на многопроцессорных ЭВМ до сих пор слабо освещены в литературе. Исключение составляют работы Воеводина (см., например, [1]) и некоторых других авторов. Основной причиной сложившейся ситуации была быстрая смена архитектуры многопроцессорных вычислительных систем (МВС). В настоящее же время в связи с приближением к физическим пределам быстродействия вычислительной и коммуникационной аппаратуры ситуация стабилизировалась, и появилась возможность сосредоточиться на исследовании алгоритмов и концепций параллельных вычислений.

В настоящей работе рассмотрена проблема решения на МВС одномерных краевых задач для уравнений и систем второго порядка, а также обобщение полученных результатов на многомерный случай. В своем анализе мы остановились на случае МВС с распределенной архитектурой, поскольку они имеют наибольшее распространение в мире в силу известных технологических особенностей и преимуществ. В основу предлагаемых нами параллельных алгоритмов положен известный принцип суперпозиции решений линейных уравнений [2]. С его помощью и метода декомпозиции области были реализованы несколько вариантов метода прогонки, позволяющие решать одномерные краевые задачи, как для традиционных граничных условий 1-ого, 2-ого и 3-его рода, так и в случае периодических и нелокальных условий. В принципе, эта методика не является новой. Однако, по определенным причинам она практически не использовалась в параллельных вычислениях. Последнее, в частности, связано с тем, что теоретическая эффективность (не учитывающая реальное время обменов) исходного (базового) параллельного алгоритма

¹ Работа выполнена при поддержке РФФИ, грант N 99-07-90388.

прогонки на системах с распределенной памятью составляет 33%. Если прибавить к этому проблемы, связанные с пересылками данных, то реальная эффективность получится еще меньше. Тем не менее, мы считаем, что указанные ограничения не “ставят крест” на методике в целом. На наш взгляд у нее есть одно важное свойство, а именно, асимптотическая зависимость ускорения и/или эффективности от числа используемых процессоров. И все дело заключается в правильном использовании этого свойства.

В настоящей работе приведены несколько примеров такого использования. В частности, нами показано, что при решении одномерных стационарных краевых задач со слабо монотонным дифференциальным оператором теоретическая эффективность может быть повышена до 100%. Кроме того, при решении нелинейных краевых задач методом Ньютона также возможно достижение высокой эффективности при незначительном снижении скорости сходимости путем специальной схемы обновления оператора перехода на итерациях. 100 % эффективность может достигаться также при решении нестационарных одномерных и многомерных линейных и нелинейных уравнений и систем. В каждом из перечисленных случаев в основе высокой эффективности алгоритмов лежит указанное выше их асимптотическое свойство.

Постановки краевых задач

Рассмотрим одномерное уравнение второго порядка общего вида на интервале $(0,1)$:

$$\frac{d}{dx} \left(k \frac{du}{dx} + r_0 u \right) + r_1 \frac{du}{dx} - qu = -f, \quad 0 < x < 1. \quad (1)$$

В линейном случае в классической постановке коэффициенты этого уравнения зависят только от координаты x :

$$k = k(x), \quad r_i = r_i(x), \quad q = q(x), \quad f = f(x),$$

и являются кусочно-непрерывными функциями на $(0,1)$, причем $k(x) \geq k_0 > 0$. При формулировке обобщенной краевой задачи для уравнения (1) можно использовать одну из трех формулировок дополнительных условий:

$$\eta_m u(x_m) + \theta_m u'(x_m) = \mu_m, \quad \eta_m^2 + \theta_m^2 \neq 0, \quad m = 0, 1, \quad (2_1)$$

$$u(0) = u(1), \quad u'(0) = u'(1), \quad (2_2)$$

$$\Phi_0(u) = c_0, \quad \Phi_1(u) = c_1. \quad (2_3)$$

Здесь предполагается, что $x_0 \equiv 0$, $x_1 \equiv 1$, η_m , θ_m , μ_m , c_m - некоторые константы ($m = 0, 1$). Краевые условия (2₁) включают в себя условия либо 1-ого, либо 2-ого, либо 3-его рода, а также могут быть смешанными. Краевые условия (2₂) соответствуют периодической задаче, для чего необходима также периодичность коэффициентов уравнения (1). В третьем случае, $\Phi_m(u)$ - линейные функционалы от $u(x)$, которые в можно представить в виде следующих интегралов:

$$\Phi_m(u) = \int_0^1 u(x') \rho_m(x') dx' \quad (m = 0, 1).$$

Мы также предполагаем, что все функции и константы, входящие в уравнение (1) и краевые условия (2_α), определяют единственное решение соответствующей краевой задачи.

В квазилинейном случае краевые задачи ставятся аналогичным образом с той лишь разницей, что коэффициенты уравнения (1) зависят еще и от решения задачи $u(x)$:

$$k = k(x, u) \geq k_0 > 0, \quad r_i = r_i(x, u), \quad q = q(x, u), \quad f = f(x, u),$$

в дополнительных условиях (2_α) величины $\eta_m = \eta_m(u)$, $\theta_m = \theta_m(u)$, $\mu_m = \mu_m(u)$ - нелинейные функции по u , а Φ_m - нелинейные функционалы от $u(x)$. Здесь также предполагается существование единственного решения каждой из рассматриваемых краевых задач.

Отметим далее, что общий нелинейный случай, когда коэффициенты уравнения (1) зависят еще и от производной $u'(x)$, рассмотрен здесь не будет, однако все результаты, полученные ниже для квазилинейных задач естественным образом обобщаются и на эту ситуацию.

В нестационарном случае мы имеем одномерное параболическое уравнение общего вида

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left(k \frac{\partial u}{\partial x} + r_0 u \right) + r_1 \frac{\partial u}{\partial x} - qu + f, \quad 0 < x < 1, \quad t > 0, \quad (3)$$

с начальным условием

$$u(x, 0) = g(x) \quad (4)$$

и граничными условиями вида (2). При этом в линейном случае коэффициенты задачи зависят также и от временной координаты t , а в квазилинейном - от t и u . Решение задачи предполагается классическим.

В заключение данного пункта отметим, что в дальнейшем мы будем различать четыре ситуации вхождения отдельных слагаемых в уравнение (1):

1) $r_0 = 0, r_1 = 0$; 2) $r_0 = 0, r_1 \neq 0$; 3) $r_0 \neq 0, r_1 = 0$; 4) $r_0 \neq 0, r_1 \neq 0$.

Это разделение обусловлено свойствами получаемого дифференциального решения и существенно влияет на выбор численного метода решения соответствующей краевой задачи.

Разностные задачи

Для численного решения сформулированных выше краевых задач мы используем метод конечных разностей, взяв за основу интегро-интерполяционный подход (ИИП) к построению разностных схем [3]. На отрезке $[0,1]$ по координате x вводим неравномерную сетку $\bar{\omega}_x = \{x = x_i, i = 0, \dots, N_x\}$, а в случае нестационарной задачи еще и равномерную сетку по времени $\bar{\omega}_t = \{t = t_j = \tau \cdot j, j = 0, \dots, N_t, \tau = L_t / N_t\}$ на отрезке интегрирования $[0, L_t]$. Тогда в стационарном случае с помощью ИИП получаем следующую разностную схему:

$$\Lambda_h y_h = -f_h, \quad (5)$$

где y_h - сеточный аналог искомой функции u , f_h - сеточный аналог правой части f , а Λ_h - соответствующий разностный оператор. Запишем определение Λ_h в индексной форме в самом общем случае во внутренних узлах сетки $\bar{\omega}_x$ ($0 < i < N_x - 1$):

$$\Lambda_h y_i = \frac{1}{\bar{h}_i} \left(\tilde{k}_{i+1/2} \frac{y_{i+1} e_{i+1} - y_i e_i}{e_{i+1/2} h_{i+1}} - \tilde{k}_{i+1/2} \frac{y_i e_i - y_{i-1} e_{i-1}}{e_{i-1/2} h_i} \right) - \tilde{q}_i y_i. \quad (6)$$

Здесь используются следующие обозначения: h_i, h_{i+1} - левый и правый шаги сетки относительно узла x_i , $\bar{h}_i = 0.5(h_i + h_{i+1})$ - средний шаг в узле x_i ,

$\tilde{k}_{i\pm 1/2} = k_{i\pm 1/2} \cdot (\chi_i + \bar{h}_i \tilde{r}_{1,i}^\pm)$, $e_i = \exp \left\{ \int_0^{x_i} r_0(x) dx / k(x) \right\}$, $\tilde{q}_i = q_i + r_{0,i} r_{1,i} k_i^{-1}$, $k_{i\pm 1/2}$ -

некоторые аппроксимации интегралов $\left[\int_{x_i}^{x_{i+1}} dx / k(x) \right]^{-1}$ и $\left[\int_{x_{i-1}}^{x_i} dx / k(x) \right]^{-1}$,

$\chi_i = 1/(1 + R_i)$, $R_i = 0.5(\tilde{r}_{1,i}^+ h_{i+1} - \tilde{r}_{1,i}^- h_i)$, $\tilde{r}_{1,i}^\pm = 0.5(r_{1,i} \pm |r_{1,i}|) / k_i$, $q_i = q(x_i)$, $k_i = k(x_i)$, $r_{m,i} = r_m(x_i)$ ($m = 0, 1$).

Схема (5)-(6) в случае $r_0 = 0, r_1 = 0$, совпадает с однородной схемой Самарского для стационарного уравнения теплопроводности. В случае $r_0 = 0, r_1 \neq 0$, имеем регуляризованную однородную схему Самарского (см. [3]) для уравнения теплопроводности с переносом. Для $r_0 \neq 0, r_1 = 0$, получаем схему типа Кареткиной (см. [4]) для стационарного уравнения

теплопроводности со слабо монотонным оператором. В общем случае ($r_0 \neq 0, r_1 \neq 0$) получаем естественное обобщение двух предыдущих схем.

Отметим далее, что схема (5)-(6) включает также и условия на границе. Для первой краевой задачи они очевидны. В случае краевых задач 2-ого и 3-его рода, в граничных узлах записываются разностные аналоги уравнения (1) с учетом заданных на границе потоков. Для периодической задачи достаточно добавить лишь одно граничное уравнение, например, в узле $i = N_x$ и учесть равенства $y_0 = y_{N_x}, y_{N_x+1} = y_1$. В случае нелокальных граничных условий используются сеточные аналоги интегральных соотношений (2₃):

$$\sum_{i=0}^{N_x} y_i \rho_{m,i} \hbar_i = c_m \quad (m = 0, 1).$$

В квазилинейной задаче необходимо учитывать также, что коэффициенты разностных уравнений (5) как внутри области, так и на границе могут зависеть от решения y_h . Единственное предположение, которое здесь существенно - это локальный характер нелинейности.

В нестационарном случае предлагается использовать известную схему с весами

$$\frac{\hat{y}_h - y_h}{\tau} = \sigma \Lambda_h \hat{y}_h + (1 - \sigma) \Lambda_h y_h + \sigma \hat{f}_h + (1 - \sigma) f_h, \quad y_h|_{t=0} = g_h, \quad (7)$$

где \hat{y}_h и y_h относятся к верхнему и нижнему слоям по времени ($t + \tau$ и t), а Λ_h определен аналогично (6). В качестве веса σ будем брать 1 и 0.5, т. е. ограничимся чисто неявной и симметричной схемами.

Завершая обсуждение разностных формулировок задачи, отметим, что коэффициенты в разностных схемах (5)-(6) и (7)-(6) обычно выбирают таким образом, чтобы порядок аппроксимации получаемых схем составлял соответственно $O(\hbar^2)$ и $O(\hbar^2 + \tau^{1/\sigma})$. Для этого коэффициенты уравнений и граничных условий, содержащие интегралы, вычисляются точно или аппроксимируются со вторым порядком по пространству.

Система алгебраических уравнений

Рассмотрим теперь алгебраическую задачу, которая возникает в результате построения разностных схем и собственно дает искомое разностное решение (стационарное, либо относящееся к очередному слою по времени). Простой анализ показывает, что мы получаем в результате систему уравнений с трехдиагональной матрицей, или близкую к ней. Этот факт записывают обычно с помощью следующей системы уравнений

$$A_i y_{i-1} - C_i y_i + B_i y_{i+1} = -F_i, \quad 1 \leq i \leq N-1. \quad (8)$$

На границе в случае 1-ой, 2-ой, 3-ей или смешанной краевых задач получаются уравнения

$$-C_0 y_0 + B_0 y_1 = -F_0, \quad -C_N y_N + A_N y_{N-1} = -F_N. \quad (9_1)$$

В случае периодических граничных условий уравнения (8) имеют место для всех $i = 2, \dots, N-1$, а в узлах $i = 1$ и $i = N$ справедливы уравнения

$$A_1 y_N - C_1 y_1 + B_1 y_2 = -F_1, \quad A_N y_{N-1} - C_N y_N + B_N y_1 = -F_N. \quad (9_2)$$

В случае нелокальных граничных условий первое и последнее уравнения системы принимают вид

$$-C_0 y_0 + \sum_{i=1}^N B_{0,i} y_i = -F_0, \quad \sum_{i=0}^{N-1} A_{0,i} y_i - C_N y_N = F_N. \quad (9_3)$$

Относительно свойств коэффициентов системы (8)-(9) предполагается следующее:

$$A_i, B_i \geq 0, \quad C_i > 0, \quad i = 0, \dots, N, \quad (10)$$

и либо

$$C_i = A_i + B_i + D_i, \quad D_i > 0, \quad i = 0, \dots, N, \quad (11_1)$$

либо

$$C_i = A_{i+1} + B_{i-1} + D_i, \quad D_i > 0, \quad i = 1, \dots, N-1. \quad (11_2)$$

Последнее условие возникает в случае немонотонного оператора ($r_0 \neq 0$) во внутренних узлах сетки. На границе в этом случае должны быть выполнены условия (11₁). Если краевая задача является нелокальной, то должны выполняться условия ее разрешимости.

Базовый алгоритм распараллеливания

Рассмотрим алгоритм решения полученной алгебраической задачи на примере системы (8)-(9₁), предполагая использование МВС с p процессорами. Для этого введем равномерное линейное разбиение множества номеров узлов сетки $\Omega = \{0, 1, \dots, N\}$ на связные подмножества $\Omega_m = \{i_1^{(m)}, \dots, i_2^{(m)}\}$ ($m = 0, \dots, p-1$), соответствующие разбиению вектора неизвестных по процессорам (см. рис.1).

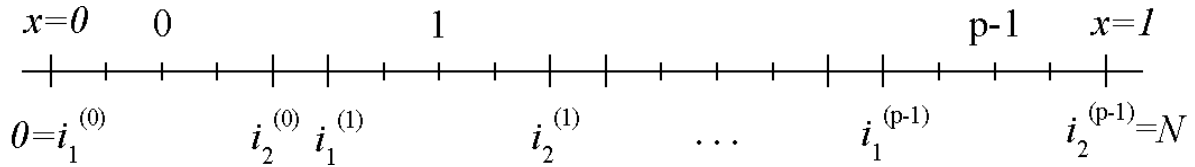


Рис. 1. Разбиение расчетной области по процессорам.

В результате такого разбиения процессор с номером m будет обрабатывать $(i_2^{(m)} - i_1^{(m)} + 1)$ точек. Представим далее решение на каждом внутреннем $(0 < m < p - 1)$ процессоре в виде следующей линейной комбинации:

$$y_i \equiv y_i^{(m)} = y_i^{(I,m)} + y_{i_1^{(m)}}^{(III,m)} y_i^{(III,m)} + y_{i_2^{(m)}}^{(II,m)} y_i^{(II,m)}, \quad (13_1)$$

где функции $y_i^{(\alpha,m)}$ ($\alpha = I, II, III$) полностью определены на множестве узлов Ω_m и играют роль некоторого базиса (см. рис. 2), а значения искомой функции на границе Ω_m - $y_{i_1^{(m)}}$ и $y_{i_2^{(m)}}$ - пока не известны. Во внутренних узлах Ω_m функция $y_i^{(I,m)}$ находится из уравнений (8), а функции $y_i^{(II,m)}$, $y_i^{(III,m)}$ из уравнений (8) с нулевой правой частью.

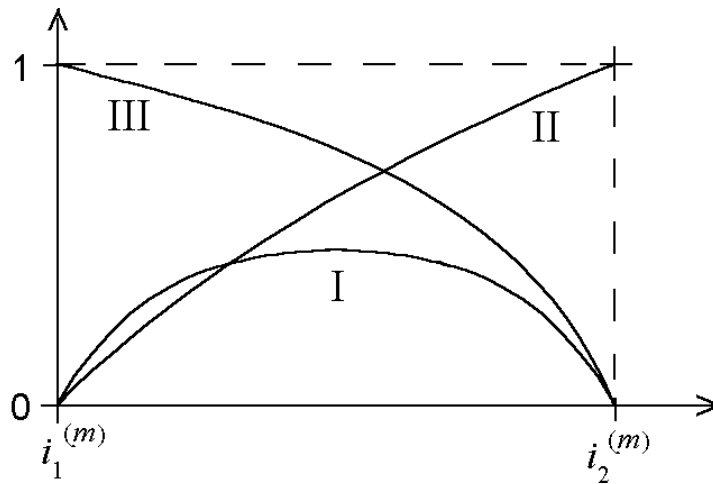


Рис. 2. Базисные функции $y_i^{(\alpha,m)}$ на процессоре с номером m .

Граничные условия для $y_i^{(\alpha,m)}$ ставятся следующим образом:

$$y_{i_1^{(m)}}^{(I,m)} = 0, y_{i_2^{(m)}}^{(I,m)} = 0; \quad y_{i_1^{(m)}}^{(II,m)} = 0, y_{i_2^{(m)}}^{(II,m)} = 1; \quad y_{i_1^{(m)}}^{(III,m)} = 1, y_{i_2^{(m)}}^{(III,m)} = 0. \quad (14_1)$$

Очевидно, что на нулевом и последнем процессорах можно использовать неполные линейные комбинации:

$$y_i^{(0)} = y_i^{(I,0)} + y_{i_2^{(0)}}^{(II,0)}, \quad (13_2)$$

$$y_i^{(p-1)} = y_i^{(I,p-1)} + y_{i_1^{(p-1)}}^{(III,p-1)}. \quad (13_3)$$

Из левого граничного условия исходной задачи для базисных функций нулевого процессора получаем:

$$\begin{aligned} C_0 y_0^{(I,0)} - B_0 y_1^{(I,0)} &= F_0, & y_{i_2^{(0)}}^{(I,0)} &= 0; \\ C_0 y_0^{(II,0)} - B_0 y_1^{(II,0)} &= 0, & y_{i_2^{(0)}}^{(II,0)} &= 1. \end{aligned} \quad (14_2)$$

Из правого граничного условия исходной задачи для базисных функций последнего процессора получаем:

$$\begin{aligned} y_{i_1^{(p-1)}}^{(I,p-1)} &= 0, & C_N y_N^{(I,p-1)} - A_N y_{N-1}^{(I,p-1)} &= F_N; \\ y_{i_1^{(p-1)}}^{(III,p-1)} &= 1, & C_N y_N^{(III,p-1)} - A_N y_{N-1}^{(III,p-1)} &= 0. \end{aligned} \quad (14_3)$$

Отметим, что базисные функции при выполнении условий (10)-(11) удовлетворяют принципу максимума:

$$\|y^{(I,m)}\|_C \leq \|D^{-1}F\|_C, \quad 0 \leq y^{(II,m)} \leq 1, \quad 0 \leq y^{(III,m)} \leq 1, \quad m = 0, \dots, p-1, \quad (15_1)$$

$$0 \leq y_i^{(II,m)} + y_i^{(III,m)} \leq 1, \quad \text{для всех } i \text{ и } m. \quad (15_2)$$

Эти свойства обеспечивают устойчивость вычислений по формулам (13).

Для нахождения неизвестных значений искомой функции в граничных узлах подобластей Ω_m запишем исходные уравнения (8) в двух соседних точках, принадлежащих процессорам с номерами m и $m+1$:

$$A_{i_2^{(m)}} y_{i_2^{(m)}-1} - C_{i_2^{(m)}} y_{i_2^{(m)}} + B_{i_2^{(m)}} y_{i_2^{(m)}+1} = -F_{i_2^{(m)}},$$

$$A_{i_1^{(m+1)}} y_{i_1^{(m+1)}-1} - C_{i_1^{(m+1)}} y_{i_1^{(m+1)}} + B_{i_1^{(m+1)}} y_{i_1^{(m+1)}+1} = -F_{i_1^{(m+1)}}.$$

Если учесть теперь в этих уравнениях очевидные связи

$$y_{i_2^{(m)}-1}^{(I,m)} = y_{i_2^{(m)}-1}^{(I,m)} + y_{i_1^{(m)}}^{(III,m)} y_{i_2^{(m)}-1}^{(II,m)} + y_{i_2^{(m)}}^{(II,m)} y_{i_2^{(m)}-1}^{(I,m)}, \quad y_{i_2^{(m)}+1}^{(I,m)} = y_{i_1^{(m+1)}}^{(I,m)},$$

$$y_{i_1^{(m+1)}-1}^{(I,m)} = y_{i_2^{(m)}}^{(I,m)}, \quad y_{i_1^{(m+1)}+1}^{(I,m)} = y_{i_1^{(m+1)}+1}^{(I,m)} + y_{i_1^{(m+1)}}^{(III,m)} y_{i_1^{(m+1)}+1}^{(II,m)} + y_{i_2^{(m+1)}}^{(II,m)} y_{i_1^{(m+1)}+1}^{(I,m)},$$

то несложно получить следующие уравнения

$$\begin{aligned}\tilde{A}_{i_2^{(m)}} y_{i_1^{(m)}} - \tilde{C}_{i_2^{(m)}} y_{i_2^{(m)}} + \tilde{B}_{i_2^{(m)}} y_{i_1^{(m+1)}} &= -\tilde{F}_{i_2^{(m)}}, \\ \tilde{A}_{i_1^{(m+1)}} y_{i_2^{(m)}} - \tilde{C}_{i_1^{(m+1)}} y_{i_1^{(m+1)}} + \tilde{B}_{i_1^{(m+1)}} y_{i_2^{(m+1)}} &= -\tilde{F}_{i_1^{(m+1)}},\end{aligned}\tag{16}$$

с коэффициентами

$$\begin{aligned}\tilde{A}_{i_2^{(m)}} &= A_{i_2^{(m)}} y_{i_2^{(m)}-1}^{(III,m)}, \quad \tilde{B}_{i_2^{(m)}} = B_{i_2^{(m)}}, \\ \tilde{C}_{i_2^{(m)}} &= C_{i_2^{(m)}} - A_{i_2^{(m)}} y_{i_2^{(m)}-1}^{(II,m)}, \quad \tilde{F}_{i_2^{(m)}} = F_{i_2^{(m)}} + A_{i_2^{(m)}} y_{i_2^{(m)}-1}^{(I,m)}, \\ \tilde{A}_{i_1^{(m+1)}} &= A_{i_1^{(m+1)}}, \quad \tilde{B}_{i_1^{(m+1)}} = B_{i_1^{(m+1)}} y_{i_1^{(m+1)}+1}^{(II,m+1)}, \\ \tilde{C}_{i_1^{(m+1)}} &= C_{i_1^{(m+1)}} - B_{i_1^{(m+1)}} y_{i_1^{(m+1)}+1}^{(III,m+1)}, \quad \tilde{F}_{i_1^{(m+1)}} = F_{i_1^{(m+1)}} + B_{i_1^{(m+1)}} y_{i_1^{(m+1)}+1}^{(I,m+1)}.\end{aligned}$$

Действуя аналогично для каждой пары процессоров, получим следующую систему из $2p - 2$ уравнений для искомым $2p - 2$ неизвестных

$$\tilde{A}_i y_{i-1} - \tilde{C}_i y_i + \tilde{B}_i y_{i+1} = -\tilde{F}_i, \quad i \in \tilde{\Omega} = \{i_2^{(0)}, i_1^{(1)}, i_2^{(1)}, \dots, i_1^{(p-1)}\},\tag{17}$$

где под индексом $i \pm 1$ понимается переход к соответствующему соседнему элементу из множества $\tilde{\Omega}$.

Нетрудно проверить, что в граничных узлах $i_2^{(0)}$ и $i_1^{(p-1)}$ уравнения (17) принимают вид (9₁). Кроме того, в силу свойств (15) коэффициенты новой “короткой” системы уравнений также удовлетворяют условиям принципа максимума вида (10)-(11). Таким образом, решение системы (17) существует и является единственным. Определив его методом обычной (скалярной) прогонки, можно с помощью формул (13), вычислить решение исходной задачи (8)-(9).

Приведем теперь полную последовательность действий в алгоритме параллельной прогонки. Сначала на каждом процессоре с помощью алгоритма скалярной прогонки решаются три (или две) задачи для нахождения базисных функций $y^{(\alpha,m)}$. Затем находятся коэффициенты для новой задачи относительно неизвестных $y_{i_1}^{(m)}, y_{i_2}^{(m)}$ ($m = 0, \dots, p - 1$). Эти коэффициенты пересылаются корневому процессору, например, нулевому. Корневой процессор осуществляет решение короткой системы уравнений (17) и рассылает полученные значения $y_{i_1}^{(m)}$ и $y_{i_2}^{(m)}$ соответствующему процессору. Получив эти данные, каждый процессор восстанавливает свою часть искомого решения по формулам (13). На этом процедура решения заканчивается.

Рассмотрим теперь вопрос об эффективности изложенного алгоритма. Для этого напомним, что каждый параллельный алгоритм

оценивается по двум параметрам - ускорению S_p и эффективности E_p , которые определяются по формулам

$$S_p = \frac{t_1}{t_p}, \quad E_p = \frac{S_p}{p} \cdot 100\%, \quad (18)$$

где t_1 - время решения исходной задачи на одном процессоре, t_p - время решения исходной задачи по параллельному алгоритму на p процессорах. В теоретических исследованиях обычно не принимают во внимание накладные расходы, связанные с временем обменов, поскольку последнее является неконтролируемым фактором и сильно зависит от конкретной МВС. В связи с этим времена t_1 и t_p в формулах (18) заменяют на оценки числа обобщенных арифметических операций Q_1 и Q_p . Последние оценивают по количеству элементарных объектов вычислений, например, как в нашем случае, по числу узлов расчетной сетки N .

Как известно, при решении исходной задачи с помощью скалярного алгоритма прогонки приходится выполнить $Q_1 = C_1 N$ обобщенных арифметических действий. При решении задачи по изложенному параллельному алгоритму производится $Q_p = 3C_2 \frac{N}{p} + 2C_1 p$ действий.

Последняя оценка следует из того, что на каждом процессоре, кроме корневого, решаются три задачи размерности N/p , а корневой процессор решает еще и короткую задачу размерности $2p-2$. Константы C_1, C_2 не зависят от N и p и близки по величине ($C_2/C_1 \sim 1.2$). Теперь мы можем оценить величину ускорения и эффективность параллельного алгоритма:

$$S_p = \frac{p}{3C_2/C_1 + 2p^2/N}, \quad E_p = \frac{1}{3C_2/C_1 + 2p^2/N} \cdot 100\%.$$

Эта формула показывает, что при $p \ll \sqrt{N}$ ускорение S_p близко к величине $p/3$, а эффективность E_p составляет примерно 33 %. Безусловно, полученные оценки можно уточнить, если рассмотреть структуру вычислений и учесть соотношения между различными видами арифметических операций. Однако полученная асимптотика не сильно изменится.

Комментарии и обобщения

Рассмотрим теперь особенности использования изложенного выше алгоритма в нестандартных ситуациях.

Во-первых, в случае немонотонного оператора ($r_0 \neq 0$) используется алгоритм немонотонной прогонки, который даже в скалярном случае требует введения аналогичных базисных функций. При этом аналогом разбиения по процессорам здесь является разбиение исходной области точками смены знака функции r_0 . Если k - число смен знака, то короткая система уравнений содержит в скалярном алгоритме $2k$, а в параллельном - не более $2p + 2k$ неизвестных. В обоих вариантах она решается ленточным методом Гаусса. Эффективность параллельного алгоритма в данной ситуации может достигать 100% за счет того, что его скалярный аналог является более трудоемким по сравнению с алгоритмом обычной прогонки.

Если же функция r_0 не меняет знак, то введение базисных функций в скалярном алгоритме не нужно, и можно использовать левую или правую (в зависимости от знака r_0) немонотонную прогонку (см. [4]). Соответственно эффективность параллельного варианта немонотонной прогонки в этом случае равна 33%.

Во-вторых, в случае периодической задачи, в скалярном варианте используется алгоритм циклической прогонки. В нем фактически вводится аналогичный базис из двух функций на всем отрезке интегрирования, а короткая система содержит 2 неизвестных и решается любым способом. В параллельном алгоритме, как и выше, на каждом процессоре вводится базис из трех функций, а короткая система содержит $2p - 1$ неизвестных и решается с помощью скалярного алгоритма циклической прогонки. Эффективность параллельного варианта циклической прогонки может достигать 67%.

В-третьих, случай интегральных граничных условий аналогичен периодической задаче. Отличие состоит лишь в том, что короткая система уравнений в параллельном алгоритме содержит $2p$ неизвестных и может решаться с помощью любого варианта метода скалярной прогонки. Эффективность параллельного варианта прогонки здесь также может быть равной 67%.

Скажем несколько слов о квазилинейной и нестационарной задачах, а также обобщениях на многомерный случай и случай систем уравнений.

В квазилинейной постановке для реализации построенной разностной схемы необходим итерационный процесс. Обычно в качестве такового используют итерации по Ньютону. Тогда при условии локального характера нелинейности (Ньютоновский оператор перехода не выходит за рамки шаблона разностной схемы, за исключением быть может граничных узлов) на каждой итерации мы получаем задачу типа (8)-(9) и можем использовать параллельный алгоритм прогонки. При этом, если оператор

перехода меняется не на каждой итерации, то пересчитывать вторую и третью базисные функции на каждой итерации не нужно. В этом случае число операций в параллельном алгоритме существенно сокращается, и есть возможность получить эффективность параллелизации близкую к 100%.

Аналогичная ситуация имеет место и в нестационарном случае. Если функции k , r_i и q , составляющие оператор перехода со слоя на слой, постоянны или зависят только от x , то базисные функции $y^{(II)}$, $y^{(III)}$ можно вычислить только один раз (в начале расчета). Тогда на каждом слое по времени пересчитывается только функция $y^{(I)}$, и эффективность распараллеливания увеличивается до 100%. Если k , r_i и q слабо зависят от t , то, как и в методе Ньютона, их можно вычислять не на каждом слое, а через несколько шагов по времени. В этом случае тоже можно получить высокую эффективность параллельного алгоритма.

В случае многомерной краевой задачи в областях прямоугольной формы возможно эффективное использование базового параллельного алгоритма на этапах реализации итерационных или временных схем с факторизованным оператором. В качестве примера можно привести решение многомерного уравнения Пуассона с помощью метода переменных направлений или решение многомерного уравнения теплопроводности с помощью локально-одномерных разностных схем. Нетрудно показать, что эффективность получаемых здесь параллельных реализаций также может достигать 100%.

В случае систем уравнений вида (8)-(9) можно воспользоваться алгоритмом параллельной матричной прогонки, который строится на изложенных выше принципах. Очевидно, что во многих случаях и здесь удастся получить высокую эффективность распараллеливания.

В заключение можно сформулировать следующие выводы.

1. Рассмотренный базовый алгоритм параллельной прогонки обладает асимптотическим свойством, которое позволяет эффективно использовать его, как непосредственно, так и в комбинациях с другими методами.

2. Эффективность базового алгоритма прогонки составляет от 33 до 100% в зависимости от вычислительной сложности исходной задачи в скалярном случае и способа агрегирования параллельной прогонки с другими алгоритмами.

Список литературы

1. Архитектура ЭВМ и численные методы. Сб. науч. трудов / Под ред. В.В. Воеводина. Москва: ОВМ АН СССР, 1983.

2. Коновалов А.Н. Введение в вычислительные методы линейной алгебры. Новосибирск: ВО "Наука", Сибирская издательская фирма, 1993.
3. Самарский А.А. Теория разностных схем. М.: Наука, 1983.
4. Кареткина Н.В. Безусловно устойчивая разностная схема для параболических уравнений, содержащих первые производные. Журнал вычислительной математики и математической физики, 1980, 20(7), 236-240.