

С. И. Мартыненко

ОЦЕНКИ ЭФФЕКТИВНОСТИ РАСПАРАЛЛЕЛИВАНИЯ УНИВЕРСАЛЬНОЙ МНОГОСЕТОЧНОЙ ТЕХНОЛОГИИ

Рассмотрены возможные схемы распараллеливания универсальной многосеточной технологии и получены оценки эффективности параллелизма. Теоретически обоснована возможность достижения высокой эффективности параллелизма при численном решении краевых задач на структурированных сетках.

E-mail: Martynenko@ciam.ru

Ключевые слова: универсальная многосеточная технология, краевые задачи, высокопроизводительные и параллельные вычисления.

Решение ряда прикладных задач требует огромного объема вычислительной работы, который, несмотря на развитие вычислительной техники, невозможно в разумные сроки выполнить даже на современных суперкомпьютерах. К таким задачам следует отнести моделирование трехмерных турбулентных течений вязкой среды, моделирование физико-химических превращений в вязких средах и многие другие. Одно из направлений повышения производительности вычислительных систем связано с разработкой алгоритмов параллельных вычислений, чему уделяется значительное внимание [1].

Анализ параллельных свойств основных итерационных методов для решения систем линейных алгебраических уравнений (СЛАУ) выполнен сравнительно давно [2–4]. Показано, что простейшие итерационные методы (простой итерации, Якоби, Зейделя, последовательной верхней релаксации) сравнительно легко распараллеливаются. Однако медленная сходимость данных итерационных методов не позволяет разработать на их основе эффективные алгоритмы для численного решения краевых задач. Вместе с тем быстро сходящиеся итерационные методы, такие как предобусловленные методы сопряженных градиентов, гораздо труднее распараллелить.

Определенный оптимизм был связан с классическими многосеточными методами (КММ), оптимальная (или близкая к ней) скорость сходимости достигается при использовании простейших итерационных методов. Однако и КММ невозможно эффективно распараллеливать: в КММ СЛАУ большей размерности алгебраическими или геометрическими методами сводится к СЛАУ меньшей размерности. Поэтому число уравнений СЛАУ самой меньшей размерности может оказаться меньше, чем число используемых процессоров. Даже если число уравнений СЛАУ самой меньшей размерности будет равно числу процессоров, то выполнение каждой сглаживающей итерации свя-

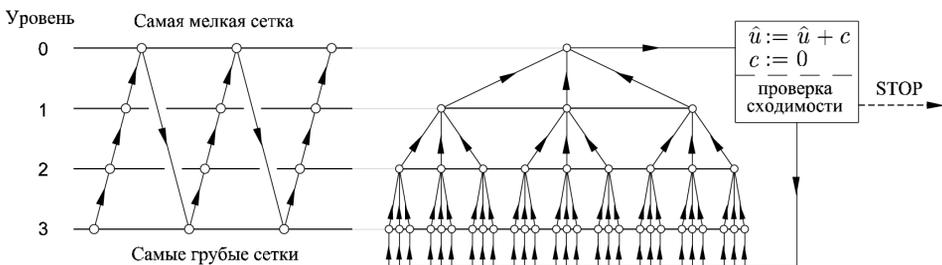


Рис. 1. Сглаживание на одномерной многосеточной структуре

зано с интенсивными обменами данными, которые существенно снижают эффективность распараллеливания. Обзор направлений развития методов распараллеливания КММ приведен в работе [5].

Несмотря на многообразие различных подходов к построению алгоритмов распараллеливания для численного решения краевых задач, становится очевидной необходимость разработки нового итерационного метода, который обладает скоростью сходимости, близкой к оптимальной, и естественным параллелизмом. Такой метод был предложен и получил название универсальной многосеточной технологии (УМТ) [6–8].

В УМТ каждая сетка представима в виде объединения 3^N ($N = 2, 3$) подсеток, которые не имеют между собой общих узлов [6–8]. Совокупность всех сеток, используемых для выполнения многосеточной итерации, получила название *многосеточной структуры*. Сглаживание на одномерной ($N = 1$) многосеточной структуре (многосеточная итерация) проиллюстрировано на рис. 1. Отметим, что построение многосеточной структуры возможно лишь для структурированных сеток.

Пусть \bar{N}_x , \bar{N}_y и \bar{N}_z есть число узлов самой мелкой сетки по каждому пространственному направлению. Номер сеточного уровня с самыми грубыми сетками определяется как

$$L^+ = \left\lceil \frac{\lg \max(\bar{N}_x, \bar{N}_y, \bar{N}_z)}{\lg 3} - 1 \right\rceil, \quad (1)$$

где квадратные скобки означают целую часть. С учетом самой мелкой сетки общее число сеточных уровней составит $L^+ + 1$. Зависимость числа сеточных уровней от числа узлов самой мелкой сетки (1) показана на рис. 2. Если число узлов самых грубых сеток превышает 8, то число сеточных уровней следует увеличить на единицу.

С точки зрения распараллеливания вычислений УМТ обладает следующими особенностями:

— грубые сетки каждого уровня не имеют общих точек, поэтому сглаживающие итерации на этих сетках могут проводиться параллельно вне зависимости от используемой сглаживающей процедуры;

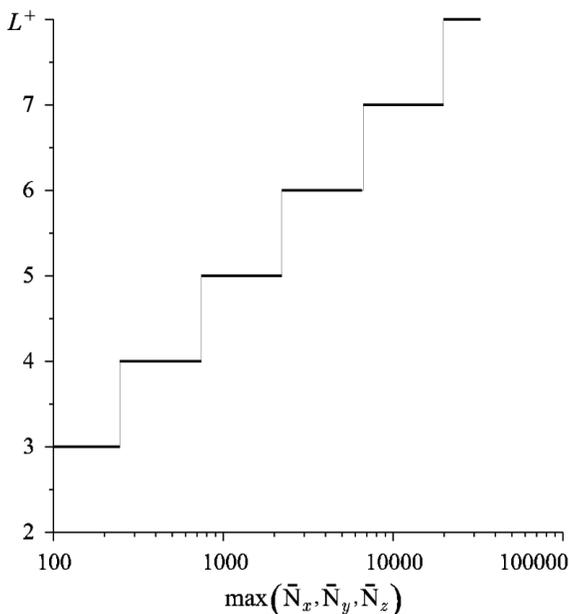


Рис. 2. Зависимость числа сеточных уровней от числа узлов самой мелкой сетки

- отсутствие общих узлов и граней уменьшает обмен данными между процессорами;
- фиксированное число сеток на каждом уровне позволяет заранее предсказать необходимое число процессоров для распараллеливания УМТ;
- почти одинаковое число точек на сетках одного уровня позволяет добиться равномерной загрузки процессоров.

Оценки минимальной эффективности распараллеливания УМТ приведены в [9, 10]. Целью данной работы является получение оценок эффективности параллельного исполнения УМТ.

Укажем основные определения параллелизма, которые будут использованы в дальнейшем.

Определение 1. Ускорением S_p параллельного алгоритма называется отношение

$$S_p = \frac{T(1)}{T(p)}, \quad (2)$$

где $T(1)$ — время, необходимое для выполнения программы на однопроцессорном компьютере; $T(p)$ — время счета на p процессорах. Ускорение S_p позволяет сравнить поведение данного алгоритма для одного и p процессоров.

Определение 2. Эффективностью параллельного алгоритма E_p называется величина

$$E_p = \frac{S_p}{p}. \quad (3)$$

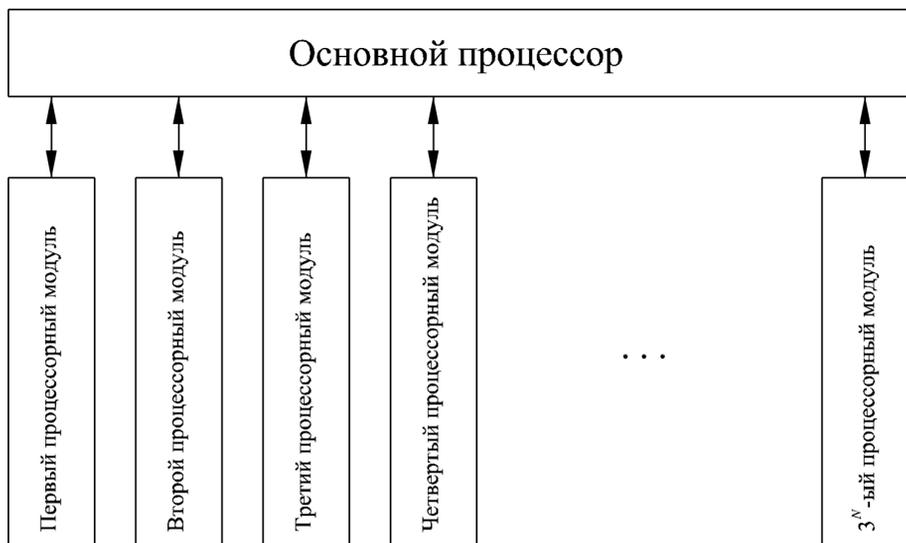


Рис. 3. Архитектура компьютера для оценки минимальной эффективности

Определение 3. *Крупнозернистость* задачи означает наличие в ней больших независимых подзадач, которые можно обрабатывать параллельно. *Мелкозернистость* соответствует возможности параллельного выполнения малых подзадач [3].

Неполное распараллеливание УМТ. Для наглядности получения оценки эффективности неполного распараллеливания УМТ спроектируем гипотетический многопроцессорный компьютер, который состоит из основного процессора и процессорных модулей (рис. 3). Процессорные модули связаны только с основным процессором, и любое общение между основным процессором и процессорными модулями осуществляется посредством передачи сообщений. Поскольку уровень l состоит из 3^{Nl} сеток ($N = 2, 3$), то число процессорных модулей должно составить 3^{Nk} , где $k = 1, \dots, L^+$. В дальнейшем величина k будет называться *глубиной распараллеливания* УМТ. Значение $k = 0$ соответствует однопроцессорному исполнению.

Отметим, что основной процессор используется только для наглядности, его роль может исполнять любой из процессорных модулей. Поэтому наличие основного процессора не будет учитываться при оценке ускорения и эффективности параллелизма.

Рассмотрим простейший случай, когда $k = 1$. Тогда число процессорных модулей p , необходимых для распараллеливания УМТ, составит $p = 3^N$, т.е. девять модулей для двумерных задач ($p = 9$) и двадцать семь ($p = 27$) — для трехмерных. Распределение вычислительных сеток между процессорными модулями показано на рис. 4. Вычисление поправки на каждой сетке первого уровня и ее подсетках осуществляется независимо на отдельном процессорном модуле.

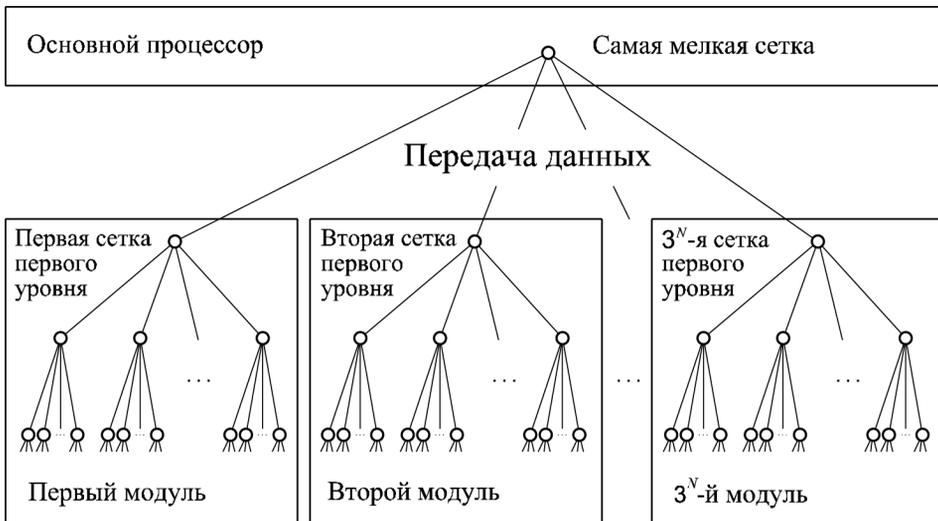


Рис. 4. Распределение грубых сеток среди процессорных модулей

Нетрудно видеть, что УМТ на уровнях с грубыми сетками ($L > 0$) обладает полным параллелизмом.

Статический цикл. Схема распараллеливания вычислений показана на рис. 5; здесь сглаживание на сетках первого уровня и их подсетках проводится независимо на процессорных модулях. После сглаживания на сетках первого уровня поправки передаются в основной процессор, на котором выполняется сглаживание на самой мелкой сетке, пересчитывается приближение к решению и проверяется критерий останова. Если полученное приближение не удовлетворяет критерию останова, то вычисление поправки начинается на грубых сетках на каждом процессорном модуле.

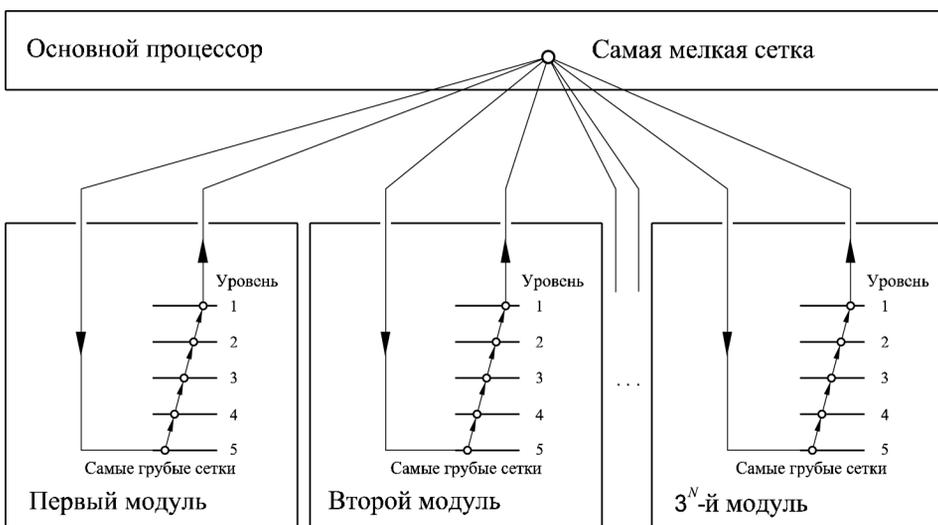


Рис. 5. Частичное распараллеливание УМТ (статический цикл)

Минимальное ускорение достигается при неполном распараллеливании УМТ, т.е. сглаживание на самой мелкой сетке проводится на одном процессоре, а $p - 1$ процессоров простаивают. Фактически при неполном распараллеливании УМТ используется только крупная зернистость многосеточной структуры.

При получении соответствующих оценок предположим, что время пересылки данных пренебрежимо мало и на всех сетках выполняется одинаковое число итераций одной и той же сглаживающей процедуры. Тогда время, необходимое для выполнения программы на однопроцессорном компьютере, составит

$$T(1) = \sum_{l=0}^{L^+} T_l,$$

а время счета на p процессорах

$$T(p) = T_0 + \frac{1}{p} \sum_{l=1}^{L^+} T_l,$$

поскольку сглаживание на самой мелкой сетке осуществляется на одном процессоре. Тогда с учетом определений (2) и (3) минимальные ускорение S_{\min} и эффективность E_{\min} определяются как

$$S_{\min} = \frac{\sum_{l=0}^{L^+} T_l}{T_0 + \frac{1}{p} \sum_{l=1}^{L^+} T_l}; \quad E_{\min} = \frac{1}{p} \frac{\sum_{l=0}^{L^+} T_l}{T_0 + \frac{1}{p} \sum_{l=1}^{L^+} T_l}, \quad (4)$$

где T_l есть время решения задачи на l -м уровне.

Из ранее сделанных допущений следует, что $T_l = \text{const}$. Тогда соотношение для минимальных ускорения и эффективности параллелизма (4) принимает вид

$$S_{\min} = \frac{L^+ + 1}{1 + \frac{1}{p}} = p E_{\min}. \quad (5)$$

Поскольку для распараллеливания первой глубины число используемых процессорных модулей p составит 3^N , то соотношение для минимальных ускорения и эффективности (5) может быть записано в виде

$$S_{\min} = 3^N E_{\min} = 3^N \frac{L^+ + 1}{L^+ + 3^N}. \quad (6)$$

Оценка (6) показывает, что минимальная эффективность неполного распараллеливания УМТ для численного решения двумерной краевой задачи ($N = 2$) на сетке $\sim 1000 \times 1000$ ($L^+ = 5$) составит 43%, а в

трехмерном случае ($N = 3$) на сетке $\sim 1000 \times 1000 \times 1000$ ($L^+ = 5$) — 19 %.

Большая эффективность распараллеливания УМТ в двумерном случае (по сравнению с трехмерным) обусловлена меньшим числом простаивающих процессоров (8 против 26) при сглаживании на самой мелкой сетке.

Для иллюстрации приведенных выше допущений рассмотрим первую краевую задачу для уравнения

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = -f(x, y, z) = \frac{1}{3} \exp\left(\frac{x + y + z}{3}\right)$$

в единичном кубе, которая имеет точное решение

$$u_a(x, y, z) = \exp\left(\frac{x + y + z}{3}\right),$$

откуда нетрудно найти соответствующие граничные условия. Численное решение данной задачи осуществим на пятиуровневой многосеточной структуре ($L^+ = 4$) с самой мелкой сеткой $201 \times 201 \times 201$, начиная с нулевого начального приближения. В качестве сглаживающей процедуры на уровнях с грубыми сетками использован вариант предобусловленного метода сопряженных градиентов, предложенный в [12].

Равномерность загрузки процессоров в процессе выполнения сглаживающих итераций на уровнях с грубыми сетками может быть показана при вычислительных экспериментах на однопроцессорном компьютере. Положим, что для распараллеливания данной трехмерной задачи ($N = 3$) используется $p = 3^N = 27$ процессоров. В соответствии со схемой, представленной на рис. 5, сглаживание на уровнях с грубыми сетками состоит из выполнения одинакового числа итераций предобусловленного метода сопряженных градиентов на 27 независимых сетках первого уровня и их подсетках.

Пусть T_i есть время выполнения сглаживающих итераций предобусловленного метода сопряженных градиентов на i -м процессоре ($i = 1, 2, \dots, 27$). Определим среднее время $\langle T \rangle$ выполнения сглаживающих итераций как среднее арифметическое значение:

$$\langle T \rangle = \frac{1}{3^N} \sum_{i=1}^{3^N} T_i = \frac{1}{27} \sum_{i=1}^{27} T_i.$$

В качестве меры равномерности загрузки процессоров выберем отклонение T_i от среднего времени $\langle T \rangle$

$$\delta_T = \frac{\langle T \rangle - T_i}{\langle T \rangle}.$$

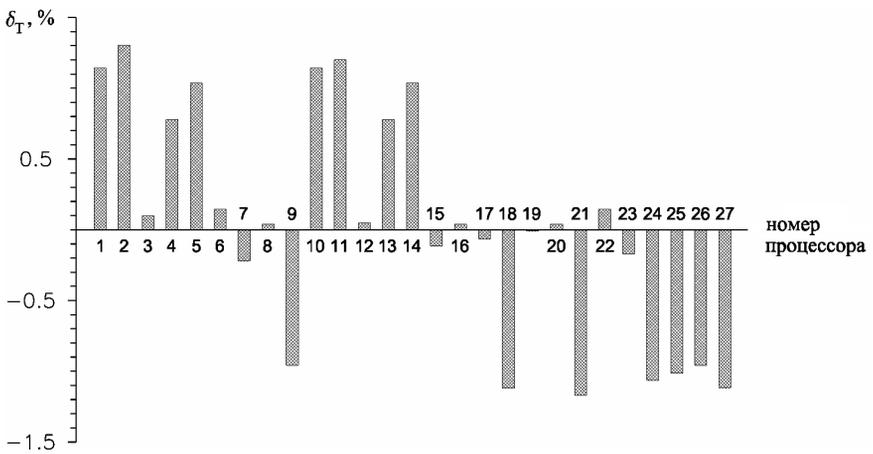


Рис. 6. Отклонение времени выполнения сглаживающих итераций предобусловленного метода сопряженных градиентов от среднего значения на сетках первого уровня и их подсетках

Результаты вычислительного эксперимента, приведенные на рис. 6, показывают, что максимальное отклонение времени выполнения сглаживающих итераций предобусловленного метода сопряженных градиентов на i -м процессоре от среднего значения $\langle T \rangle$ не превышает одного процента. Другими словами, сглаживающая процедура на сетках первого уровня и их подсетках состоит из 3^N независимых задач, решение которых требует почти одинакового времени. Тем самым достигается почти равномерная загрузка процессоров.

Следующий вычислительный эксперимент посвящен проверке допущения об одинаковом времени выполнения сглаживающих итераций на различных сеточных уровнях (т.е. $T_l = \text{const}$). Для этого рассмотрим первую краевую задачу для уравнения

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -f(x, y, z) = 2e^{x-y},$$

в единичном квадрате, которая имеет точное решение

$$u_a(x, y) = e^{x-y},$$

откуда нетрудно найти соответствующие граничные условия. Численное решение данной задачи проведем на семиуровневой многосеточной структуре ($L^+ = 6$) с самой мелкой сеткой 2001×2001 , начиная с нулевого начального приближения. В качестве сглаживающей процедуры использован метод Зейделя с точечным упорядочиванием неизвестных. Полное время выполнения многосеточной итерации принято за 100%.

В табл. 1 показано распределение относительного времени счета по сеточным уровням $T_l / \sum_{l=0}^{L^++1} T_l$ и уменьшение ошибки численного

Типичное распределение времени исполнения сглаживающих итераций по сеточным уровням

q	$\max_{ij} u_a(x_i, y_j) - u_{ij} $	T_0	T_1	T_2	T_3	T_4	T_5	T_6
1	$0,270 \cdot 10^{+1}$	13,18%	13,50%	14,12%	14,25%	14,50%	15,12%	15,32%
2	$0,181 \cdot 10^{+1}$	13,18%	13,38%	14,15%	14,35%	14,50%	15,00%	15,45%
3	$0,105 \cdot 10^{+1}$	13,09%	13,22%	14,15%	14,28%	14,92%	15,15%	15,18%
4	$0,222 \cdot 10^{-1}$	13,38%	13,50%	14,12%	14,25%	14,50%	15,00%	15,25%
5	$0,443 \cdot 10^{-3}$	13,15%	13,50%	14,25%	14,35%	14,50%	15,00%	15,25%
6	$0,194 \cdot 10^{-4}$	13,09%	13,12%	14,45%	14,48%	14,52%	15,15%	15,18%
7	$0,168 \cdot 10^{-5}$	13,38%	13,50%	14,12%	14,25%	14,50%	15,00%	15,25%
8	$0,117 \cdot 10^{-6}$	13,25%	13,50%	14,15%	14,35%	14,50%	15,00%	15,25%
9	$0,756 \cdot 10^{-8}$	13,09%	13,12%	14,35%	14,48%	14,62%	15,15%	15,18%
10	$0,347 \cdot 10^{-8}$	13,09%	13,12%	14,18%	14,62%	14,68%	15,12%	15,18%
11	$0,333 \cdot 10^{-8}$	13,18%	13,38%	14,25%	14,35%	14,60%	15,00%	15,25%
12	$0,331 \cdot 10^{-8}$	13,09%	13,12%	14,45%	14,48%	14,52%	15,15%	15,18%
13	$0,331 \cdot 10^{-8}$	13,07%	13,43%	14,40%	14,54%	14,63%	15,29%	15,64%
14	$0,331 \cdot 10^{-8}$	13,28%	13,38%	14,25%	14,35%	14,50%	15,00%	15,25%
15	$0,331 \cdot 10^{-8}$	13,09%	13,12%	14,45%	14,48%	14,52%	15,15%	15,18%

решения на каждой многосеточной итерации q . Незначительное увеличение времени выполнения сглаживающих итераций на уровнях с более грубыми сетками связано с аппроксимацией правой части Σ -модифицированной задачи, т.е. с осреднением невязки.

Динамический цикл. Очевидно, что УМТ обладает полным параллелизмом на уровнях с грубыми сетками в силу особенностей построения грубых сеток (отсутствие общих узлов). Поэтому дальнейшее повышение эффективности распараллеливания УМТ возможно путем перераспределения вычислительных усилий в пользу уровней с грубыми сетками.

Многосеточный цикл УМТ с фиксированным положением самой мелкой сетки называется *статическим*. В *динамическом* цикле каждая сетка на определенном этапе решения задачи может рассматриваться как самая мелкая. На рис. 7 показана схема распараллеливания динамического цикла, в котором сетки первого уровня рассматриваются как самые мелкие в многосеточной итерации на уровнях 5–1. Только после достижения критерия останова на сетках первого уровня осуществляется сглаживание на самой мелкой сетке ($L^+ = 0$).

Пусть q^* — число многосеточных итераций, выполняемых на каждом процессорном модуле в динамическом цикле ($q^* = 3$ на рис. 7).

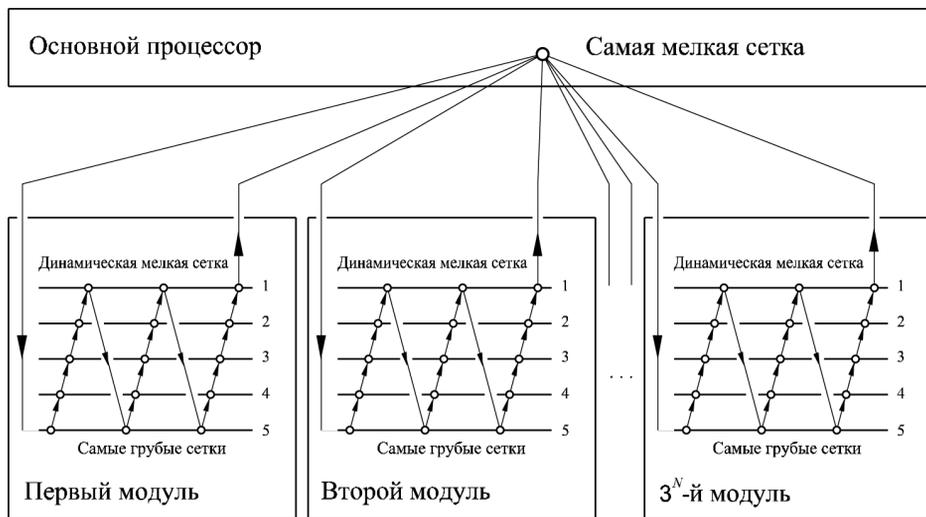


Рис. 7. Частичное распараллеливание УМТ (динамический цикл с $q^* = 3$)

Тогда оценка (6) принимает вид

$$S_{\min} = p \frac{q^* L^+ + 1}{q^* L^+ + p} = p E_{\min}. \quad (7)$$

Значения минимальных ускорения (S_{\min}) и эффективности параллелизма (E_{\min}) при решении двумерных ($N = 2$) и трехмерных ($N = 3$) задач приведены в табл. 2. Очевидно, что эффективность распараллеливания динамического цикла ($q^* > 1$) выше, чем статического ($q^* = 1$) из-за перераспределения вычислительных усилий в пользу уровней с грубыми сетками, где УМТ обладает полным параллелизмом.

Полное распараллеливание УМТ. Из оценки (7) также следует, что эффективность распараллеливания УМТ сильно зависит от эффективности распараллеливания сглаживающих итераций на самой мелкой сетке. Сначала рассмотрим распараллеливание первой глубины при тех же допущениях (на каждой сетке выполняется одинаковое число одних и тех же сглаживающих итераций).

Первая глубина. Предположим, что на самой мелкой сетке (нулевой сеточный уровень $l = 0$) удалось некоторым образом распараллелить сглаживающие итерации с ускорением S_0 и эффективностью E_0 . Тогда, согласно определению (2), время выполнения сглаживающих итераций на самой мелкой сетке составит T_0/S_0 . В данном случае выражения (4) принимают вид

$$S_{\Sigma} = \frac{T_0 + q^* \sum_{l=1}^{L^+} T_l}{\frac{T_0}{S_0} + \frac{q^*}{p} \sum_{l=1}^{L^+} T_l}; \quad E_{\Sigma} = \frac{1}{p} \frac{T_0 + q^* \sum_{l=1}^{L^+} T_l}{\frac{T_0}{S_0} + \frac{q^*}{p} \sum_{l=1}^{L^+} T_l}, \quad (8)$$

Значения минимальных ускорения и эффективности параллелизма при решении двумерных и трехмерных задач

$N = 2 (p = 9)$				$N = 3 (p = 27)$			
L^+	q^*	S_{\min}	$E_{\min}, \%$	L^+	q^*	S_{\min}	$E_{\min}, \%$
4	1	3,5	38,4	4	1	4,4	16,1
4	2	4,8	52,9	4	2	6,9	25,7
4	3	5,6	61,9	4	3	9,0	33,3
4	4	6,1	68,0	4	4	10,7	39,5
5	1	3,9	42,9	5	1	5,1	18,8
5	2	5,2	57,9	5	2	8,0	29,7
5	3	6,0	66,7	5	3	10,3	38,1
5	4	6,5	72,4	5	4	12,1	44,7
6	1	4,2	46,7	6	1	5,7	21,2
6	2	5,6	61,9	6	2	9,0	33,3
6	3	6,3	70,4	6	3	11,4	42,2
6	4	6,8	75,8	6	4	13,2	49,0
7	1	4,5	50,4	7	1	6,4	23,5
7	2	5,9	65,9	7	2	9,9	36,6
7	3	6,6	73,9	7	3	12,4	45,8
7	4	7,0	78,0	7	4	14,2	52,7

где S_{Σ} и E_{Σ} — ускорение и эффективность параллельного исполнения УМТ. Отметим, что $S_0^{\min} = 1$ и $E_0^{\min} = 1/p$.

Тогда в силу допущения $T_l = \text{const}$ выражения (8) принимают вид

$$S_{\Sigma} = pE_{\Sigma} = p \frac{q^*L^+ + 1}{q^*L^+ + \frac{p}{S_0}} = p \frac{q^*L^+ + 1}{q^*L^+ + \frac{1}{E_0}}. \quad (9)$$

В табл. 3 приведены значения ускорения и эффективности параллелизма при решении трехмерных задач ($N = 3, p = 3^N = 27$). Полученное значение эффективности E_{Σ} оказалось не ниже 90% даже при использовании статического цикла ($q^* = 1$).

В работе [11] показано, что сглаживание на уровнях с более мелкими сетками оказывает большее влияние на скорость сходимости УМТ, чем сглаживание на уровнях с более грубыми сетками. Поэтому при решении прикладных задач возможен случай, когда на различных уровнях l выполняется разное число сглаживающих итераций ν_l . В этом случае время, необходимое для выполнения сглаживающих итераций на сетках уровня l , составит $T_l = \alpha \nu_l$, где α есть некий коэффициент пропорциональности, зависящий от сглаживающей процедуры. Оценка ускорения и эффективности параллелизма в данном

**Значения ускорения и эффективности параллелизма при решении
трехмерных задач на 27 процессорах**

$E_0 \text{б} \%$	L^+	q^*	S_Σ	$E_\Sigma, \%$	$E_0, \%$	L^+	q^*	S_Σ	$E_\Sigma, \%$
60,0	5	1	24,3	90,0	70,0	5	1	25,2	93,3
60,0	5	2	25,5	94,3	70,0	5	2	26,0	96,3
60,0	5	3	25,9	96,0	70,0	5	3	26,3	97,4
60,0	5	4	26,2	96,9	70,0	5	4	26,5	98,0
60,0	6	1	24,7	91,3	70,0	6	1	25,4	94,2
60,0	6	2	25,7	95,1	70,0	6	2	26,1	96,8
60,0	6	3	26,1	96,6	70,0	6	3	26,4	97,8
60,0	6	4	26,3	97,4	70,0	6	4	26,5	98,3
60,0	7	1	24,9	92,3	70,0	7	1	25,6	94,9
60,0	7	2	25,9	95,7	70,0	7	2	26,2	97,2
60,0	7	3	26,2	97,1	70,0	7	3	26,5	98,1
60,0	7	4	26,4	97,8	70,0	7	4	26,6	98,5
80,0	5	1	25,9	96,0	90,0	5	1	26,5	98,2
80,0	5	2	26,4	97,8	90,0	5	2	26,7	99,0
80,0	5	3	26,6	98,5	90,0	5	3	26,8	99,3
80,0	5	4	26,7	98,8	90,0	5	4	26,9	99,5
80,0	6	1	26,1	96,6	90,0	6	1	26,6	98,4
80,0	6	2	26,5	98,1	90,0	6	2	26,8	99,2
80,0	6	3	26,6	98,7	90,0	6	3	26,8	99,4
80,0	6	4	26,7	99,0	90,0	6	4	26,9	99,6
80,0	7	1	26,2	97,0	90,0	7	1	26,6	98,6
80,0	7	2	26,6	98,4	90,0	7	2	26,8	99,3
80,0	7	3	26,7	98,9	90,0	7	3	26,9	99,5
80,0	7	4	26,8	99,2	90,0	7	4	26,9	99,6

случае принимает вид

$$S_\Sigma = pE_\Sigma = \frac{q^* \sum_{l=1}^{L^+} \nu_l + \nu_0}{\frac{q^*}{p} \sum_{l=1}^{L^+} \nu_l + \frac{\nu_0}{S_0}} = p \frac{q^* \sum_{l=1}^{L^+} \nu_l + \nu_0}{q^* \sum_{l=1}^{L^+} \nu_l + \frac{\nu_0}{E_0}}. \quad (10)$$

Увеличение числа сглаживающих итераций на самой мелкой сетке ($\nu_0 > \nu_l$) приведет к незначительному уменьшению эффективности параллелизма.

Вторая глубина или массивное распараллеливание. Современная многопроцессорная техника позволяет использовать сотни и даже тысячи процессоров для распараллеливания вычислений. Рассмотрим распараллеливание второй глубины при решении трехмерных задач, т.е. число используемых процессоров составит $p = 3^6 = 729$. Предпо-

ложим, что на самой мелкой сетке (нулевой сеточный уровень $l = 0$) и сетках первого уровня ($l = 1$) удалось некоторым образом распараллелить сглаживающие итерации с ускорениями S_0 и S_1 и эффективностью E_0 и E_1 . Тогда оценка ускорения и эффективности параллельного исполнения УМТ составит

$$S_{\Sigma} = pE_{\Sigma} = \frac{q^* \sum_{l=2}^{L^+} \nu_l + \nu_0 + \nu_1}{\frac{q^*}{p} \sum_{l=2}^{L^+} \nu_l + \frac{\nu_0}{S_0} + \frac{\nu_1}{S_1}} = p \frac{q^* \sum_{l=2}^{L^+} \nu_l + \nu_0 + \nu_1}{q^* \sum_{l=2}^{L^+} \nu_l + \frac{\nu_0}{E_0} + \frac{\nu_1}{E_1}}. \quad (11)$$

В случае выполнения на каждом уровне одинакового числа сглаживающих итераций ($\nu_l = \text{const}$) оценка (11) принимает вид

$$S_{\Sigma} = pE_{\Sigma} = \frac{q^*(L^+ - 1) + 2}{\frac{q^*}{p}(L^+ - 1) + \frac{1}{S_0} + \frac{1}{S_1}} = p \frac{q^*(L^+ - 1) + 2}{q^*(L^+ - 1) + \frac{1}{E_0} + \frac{1}{E_1}}. \quad (12)$$

Значения ускорения и эффективности параллелизма сведены в табл. 4 и 5. В данном случае эффективность параллельного исполнения УМТ также высока и превышает 80%. Кроме того, из оценки (12) следует, что при $L^+ \gg 1$

$$S_{\Sigma} = pE_{\Sigma} \rightarrow p \left[1 - \frac{1}{q^* L^+} \left(\frac{1}{E_0} + \frac{1}{E_1} \right) \right],$$

т.е. при сгущении самой мелкой сетки ($L^+ \uparrow$) и/или увеличении числа многосеточных итераций на уровнях с грубыми сетками ($q^* \uparrow$) и/или повышении эффективности распараллеливания сглаживающих итераций на уровнях с мелкими сетками ($E_0 \uparrow$ и/или $E_1 \uparrow$) ускорение и эффективность параллелизма будут увеличиваться.

Оценки (10) и (11) имеют важное прикладное значение. Написание и отладку компьютерных программ для параллельного исполнения УМТ следует начинать с распараллеливания сглаживающих итераций на самой мелкой сетке (первая глубина $p = 3^N$) и на сетках первого уровня (вторая глубина $p = 3^{2N}$). В результате отладки данной части программы можно экспериментально получить значения S_0 (и S_1) и E_0 (и E_1). Далее, используя оценки (10) и (11), можно оценить ускорение и эффективность параллельного варианта УМТ при решении данной задачи, поскольку на остальных уровнях УМТ обладает полным параллелизмом.

Как следствие, ускорение и эффектность УМТ в параллельном исполнении оказываются больше, чем ускорение и эффектность при распараллеливании сглаживающих итераций на самой мелкой сетке

**Значения ускорения и эффективности параллелизма при решении
трехмерных задач (массивное распараллеливание на 729 процессорах)**

E_0 , %	E_1 , %	L^+	q^*	S_Σ	E_Σ , %	E_0 , %	E_1 , %	L^+	q^*	S_Σ	E_Σ , %
60,0	60,0	5	1	596,5	81,8	70,0	60,0	5	1	616,5	84,6
60,0	60,0	5	2	643,2	88,2	70,0	60,0	5	2	657,0	90,1
60,0	60,0	5	3	665,6	91,3	70,0	60,0	5	3	676,1	92,7
60,0	60,0	5	4	678,7	93,1	70,0	60,0	5	4	687,2	94,3
60,0	60,0	6	1	612,4	84,0	70,0	60,0	6	1	630,4	86,5
60,0	60,0	6	2	656,1	90,0	70,0	60,0	6	2	668,0	91,6
60,0	60,0	6	3	676,0	92,7	70,0	60,0	6	3	684,9	94,0
60,0	60,0	6	4	687,3	94,3	70,0	60,0	6	4	694,4	95,3
60,0	60,0	7	1	624,9	85,7	70,0	60,0	7	1	641,2	88,0
60,0	60,0	7	2	665,6	91,3	70,0	60,0	7	2	676,1	92,7
60,0	60,0	7	3	683,4	93,8	70,0	60,0	7	3	691,2	94,8
60,0	60,0	7	4	693,4	95,1	70,0	60,0	7	4	699,5	96,0
60,0	70,0	5	1	616,5	84,6	70,0	70,0	5	1	637,9	87,5
60,0	70,0	5	2	657,0	90,1	70,0	70,0	5	2	671,4	92,1
60,0	70,0	5	3	676,1	92,7	70,0	70,0	5	3	686,9	94,2
60,0	70,0	5	4	687,2	94,3	70,0	70,0	5	4	695,9	95,5
60,0	70,0	6	1	630,4	86,5	70,0	70,0	6	1	649,5	89,1
60,0	70,0	6	2	668,0	91,6	70,0	70,0	6	2	680,4	93,3
60,0	70,0	6	3	684,9	94,0	70,0	70,0	6	3	694,0	95,2
60,0	70,0	6	4	694,4	95,3	70,0	70,0	6	4	701,7	96,3
60,0	70,0	7	1	641,2	88,9	70,0	70,0	7	1	658,5	90,3
60,0	70,0	7	2	676,1	92,7	70,0	70,0	7	2	686,9	94,2
60,0	70,0	7	3	691,2	94,8	70,0	70,0	7	3	699,0	96,0
60,0	70,0	7	4	699,5	96,9	70,0	70,0	7	4	705,7	96,8
60,0	80,0	5	1	632,4	86,8	70,0	80,0	5	1	654,9	89,8
60,0	80,0	5	2	667,8	91,6	70,0	80,0	5	2	682,7	93,6
60,0	80,0	5	3	684,2	93,9	70,0	80,0	5	3	695,3	95,4
60,0	80,0	5	4	693,7	95,2	70,0	80,0	5	4	702,5	96,4
60,0	80,0	6	1	644,6	88,4	70,0	80,0	6	1	664,6	91,2
60,0	80,0	6	2	677,3	92,9	70,0	80,0	6	2	690,0	94,6
60,0	80,0	6	3	691,7	94,9	70,0	80,0	6	3	701,0	96,2
60,0	80,0	6	4	699,8	96,0	70,0	80,0	6	4	707,2	97,0
60,0	80,0	7	1	654,1	89,7	70,0	80,0	7	1	672,0	92,2
60,0	80,0	7	2	684,2	93,9	70,0	80,0	7	2	695,3	95,4
60,0	80,0	7	3	697,1	95,6	70,0	80,0	7	3	705,1	96,7
60,0	80,0	7	4	704,2	96,6	70,0	80,0	7	4	710,5	97,5
60,0	90,0	5	1	645,3	88,5	70,0	90,0	5	1	668,8	91,7
60,0	90,0	5	2	676,4	92,8	70,0	90,0	5	2	691,7	94,9
60,0	90,0	5	3	690,6	94,7	70,0	90,0	5	3	701,9	96,3
60,0	90,0	5	4	698,8	95,9	70,0	90,0	5	4	707,8	97,1
60,0	90,0	6	1	656,1	90,0	70,0	90,0	6	1	676,8	92,8
60,0	90,0	6	2	684,6	93,9	70,0	90,0	6	2	697,6	95,7
60,0	90,0	6	3	697,1	95,6	70,0	90,0	6	3	706,6	96,9
60,0	90,0	6	4	704,1	96,6	70,0	90,0	6	4	711,5	97,6
60,0	90,0	7	1	664,4	91,1	70,0	90,0	7	1	682,9	93,7
60,0	90,0	7	2	690,6	94,7	70,0	90,0	7	2	701,9	96,3
60,0	90,0	7	3	701,7	96,3	70,0	90,0	7	3	709,8	97,4
60,0	90,0	7	4	707,8	97,1	70,0	90,0	7	4	714,2	98,0

**Значения ускорения и эффективности параллелизма при решении
трехмерных задач (массивное распараллеливание на 729 процессорах)**

$E_0, \%$	$E_1, \%$	L^+	q^*	S_Σ	$E_{\Sigma}, \%$	$E_0, \%$	$E_1, \%$	L^+	q^*	S_Σ	$E_{\Sigma}, \%$
80,0	60,0	5	1	632,4	86,7	90,0	60,0	5	1	645,3	88,5
80,0	60,0	5	2	667,8	91,6	90,0	60,0	5	2	676,4	92,8
80,0	60,0	5	3	684,2	93,8	90,0	60,0	5	3	690,6	94,8
80,0	60,0	5	4	693,7	95,1	90,0	60,0	5	4	698,8	95,9
80,0	60,0	6	1	644,6	88,4	90,0	60,0	6	1	656,1	90,0
80,0	60,0	6	2	677,3	92,9	90,0	60,0	6	2	684,6	93,9
80,0	60,0	6	3	691,7	94,9	90,0	60,0	6	3	697,1	95,6
80,0	60,0	6	4	699,8	96,0	90,0	60,0	6	4	704,1	96,6
80,0	60,0	7	1	654,1	89,7	90,0	60,0	7	1	664,4	91,1
80,0	60,0	7	2	684,2	93,8	90,0	60,0	7	2	690,6	94,7
80,0	60,0	7	3	697,1	95,6	90,0	60,0	7	3	701,7	96,3
80,0	60,0	7	4	704,2	96,6	90,0	60,0	7	4	707,8	97,1
80,0	70,0	5	1	654,9	89,8	90,0	70,0	5	1	668,8	91,7
80,0	70,0	5	2	682,7	93,6	90,0	70,0	5	2	691,7	94,9
80,0	70,0	5	3	695,3	95,4	90,0	70,0	5	3	701,9	96,3
80,0	70,0	5	4	702,5	96,4	90,0	70,0	5	4	707,8	97,1
80,0	70,0	6	1	664,6	91,2	90,0	70,0	6	1	676,8	92,8
80,0	70,0	6	2	690,0	94,6	90,0	70,0	6	2	697,6	95,7
80,0	70,0	6	3	701,0	96,2	90,0	70,0	6	3	706,6	96,9
80,0	70,0	6	4	707,2	97,0	90,0	70,0	6	4	711,5	97,6
80,0	70,0	7	1	672,0	92,2	90,0	70,0	7	1	682,9	93,7
80,0	70,0	7	2	695,3	95,4	90,0	70,0	7	2	701,9	96,3
80,0	70,0	7	3	705,1	96,7	90,0	70,0	7	3	709,8	97,4
80,0	70,0	7	4	710,5	97,5	90,0	70,0	7	4	714,2	98,0
80,0	80,0	5	1	672,9	92,3	90,0	80,0	5	1	687,6	94,3
80,0	80,0	5	2	694,3	95,2	90,0	80,0	5	2	703,6	96,5
80,0	80,0	5	3	703,9	96,5	90,0	80,0	5	3	710,7	97,5
80,0	80,0	5	4	709,3	97,3	90,0	80,0	5	4	714,7	98,0
80,0	80,0	6	1	680,4	93,3	90,0	80,0	6	1	693,2	95,1
80,0	80,0	6	2	699,8	96,0	90,0	80,0	6	2	707,7	97,1
80,0	80,0	6	3	708,2	97,1	90,0	80,0	6	3	713,8	97,9
80,0	80,0	6	4	712,8	97,8	90,0	80,0	6	4	717,2	98,4
80,0	80,0	7	1	686,1	94,1	90,0	80,0	7	1	697,5	95,7
80,0	80,0	7	2	703,9	96,5	90,0	80,0	7	2	710,7	97,5
80,0	80,0	7	3	711,2	97,5	90,0	80,0	7	3	716,1	98,2
80,0	80,0	7	4	715,2	98,1	90,0	80,0	7	4	719,0	98,6
80,0	90,0	5	1	687,6	94,3	90,0	90,0	5	1	703,0	96,4
80,0	90,0	5	2	703,6	96,5	90,0	90,0	5	2	713,2	97,8
80,0	90,0	5	3	710,7	97,5	90,0	90,0	5	3	717,6	98,4
80,0	90,0	5	4	714,7	98,0	90,0	90,0	5	4	720,1	98,8
80,0	90,0	6	1	693,2	95,1	90,0	90,0	6	1	706,6	96,9
80,0	90,0	6	2	707,7	97,1	90,0	90,0	6	2	715,7	98,2
80,0	90,0	6	3	713,8	97,9	90,0	90,0	6	3	719,6	98,7
80,0	90,0	6	4	717,2	98,4	90,0	90,0	6	4	721,7	99,0
80,0	90,0	7	1	697,5	95,7	90,0	90,0	7	1	709,3	97,3
80,0	90,0	7	2	710,7	97,5	90,0	90,0	7	2	717,6	98,4
80,0	90,0	7	3	716,1	98,2	90,0	90,0	7	3	721,0	98,9
80,0	90,0	7	4	719,0	98,6	90,0	90,0	7	4	722,8	99,1

(первая глубина $p = 3^N$) и на сетках первого уровня (вторая глубина $p = 3^{2N}$), т.е. $S_\Sigma > \max(S_0, S_1)$ и $E_\Sigma > \max(E_0, E_1)$.

Влияние обмена данными на эффективность параллелизма.

Обмен данными между процессорами приводит к уменьшению эффективности параллелизма. Время исполнения многопроцессорной программы можно представить в виде

$$T(p) = \frac{T(1)}{p} + T_0,$$

где $T(1)$ есть время, необходимое для выполнения программы на однопроцессорном компьютере, а T_0 — время, необходимое для обменов данными. Тогда, согласно (2) и (3), выражения для ускорения и параллелизма принимают вид

$$S_p = pE_p = \frac{T(1)}{\frac{T(1)}{p} + T_0}.$$

Положим, что время, необходимое для обменов данными, пропорционально времени, необходимому для выполнения программы на однопроцессорном компьютере, т.е.

$$T_0 = \alpha T(1),$$

где $\alpha \geq 0$ — коэффициент пропорциональности. Тогда

$$S_p = pE_p = \frac{p}{1 + p\alpha},$$

т.е. наличие обменов данными ($\alpha \neq 0$) приводит к снижению ускорения и эффективности параллелизма. К сожалению, значение коэффициента α зависит от программных и аппаратных средств многопроцессорного компьютера и от решаемой задачи, поэтому получить оценку α теоретическими методами весьма затруднительно. Проще экспериментально измерить величину эффективности параллелизма и затем определить значение параметра α по формуле

$$\alpha = \frac{1}{p} \left(\frac{1}{E_p} - 1 \right).$$

В частности, при $E_p = 50\%$ величина α составит $1/p$.

В оценках типа (10) или (12) время обмена данными при выполнении сглаживающих итераций на уровнях с мелкими сетками учтено значениями эффективностей E_0 и E_1 . Однако отсутствует учет времени обмена данными при переходе к уровням с числом сеток, меньшим, чем число процессоров. Поэтому в общем случае оценка (9) принимает вид

$$\tilde{S}_\Sigma = p\tilde{E}_\Sigma = p \frac{q^*L^+ + 1}{q^*L^+ + \frac{1}{E_0} + \alpha p},$$

где \tilde{S}_Σ и \tilde{E}_Σ — действительные значения ускорения и эффективности параллелизма, а параметр α учитывает время обмена данными при переходе к уровням с числом сеток, меньшим, чем число процессоров. Отсюда при

$$\alpha p \ll q^*L^+ + \frac{1}{E_0}$$

нетрудно получить

$$\frac{\tilde{S}_\Sigma}{S_\Sigma} = \frac{\tilde{E}_\Sigma}{E_\Sigma} \approx 1 - \frac{\alpha p}{q^*L^+ + \frac{1}{E_0}},$$

т.е. действительные значения ускорения и эффективности параллелизма будут несколько ниже, чем следует из оценки (9), из-за негативного влияния обмена данными между процессорами.

Аналогичную оценку можно получить и для распараллеливания второй глубины.

Выводы. Крупная зернистость УМТ позволяет эффективно распараллеливать вычисления при решении (не)линейных краевых задач вне зависимости от выбора сглаживающей процедуры для уровней с грубыми сетками. Отсутствие интерполяции в УМТ позволяет снизить требования к выбору сглаживающей процедуры для уровней с мелкими сетками, и тем самым появляется возможность использовать хорошо распараллеливаемые сглаживатели. Однако столь высокая эффективность распараллеливания УМТ обусловлена свойствами самой мелкой сетки, т.е. возможностью построения многосеточной структуры. Неструктурированные сетки не обладают данным свойством, поэтому эффективность распараллеливания вычислительных алгоритмов для численного решения краевых задач на неструктурированных сетках оказывается гораздо ниже.

Работа выполнена при поддержке РФФИ (проект 09-01-00151). Автор выражает искреннюю признательность профессору М.П. Галанину за поддержку работы и ряд ценных критических замечаний по полученным результатам.

СПИСОК ЛИТЕРАТУРЫ

1. Воеводин В. В., Воеводин В. л. В. Параллельные вычисления. – СПб.: БХВ-Петербург, 2002. – 608 с.
2. Марчук Г. И. Методы вычислительной математики. – М.: Наука, 1989. – 608 с.

3. Ортега Дж. Введение в параллельные и векторные методы решения линейных систем: Пер. с англ. – М.: Мир, 1991. – 367 с.
4. Воеводин В. В. Математические модели и методы в параллельных процессах. – М.: Наука, 1986. – 296 с.
5. Edmond Chow, Robert D. Falgout, Jonathan J. Hu, Raymond S. Tuminaro, and Ulrike Meier Yang. A survey of parallelization techniques for multigrid solvers, in *Frontiers of Parallel Processing for Scientific Computing*, SIAM Publications, Philadelphia, PA, 2005.
6. Мартыненко С. И. Универсальная многосеточная технология для численного решения систем дифференциальных уравнений в частных производных // *Вычислительные методы и программирование*. – 2001. – Т. 1, раздел 1. – С. 1–11.
7. Martynenko S. I. Robust multigrid technique for black box software // *Comp. Meth. in Appl. Math.* – 2006. – V. 6, no. 4. – P. 413–435.
8. Мартыненко С. И. Универсальная многосеточная технология // *Математическое моделирование*. – 2009. – Т. 21, № 9. – С. 66–79.
9. Мартыненко С. И. Распараллеливание универсальной многосеточной технологии // *Вычислительные методы и программирование*. – 2003. – Т. 4, раздел 1. – С. 45–51.
10. Martynenko S. I. Potentialities of the robust multigrid technique // *Comp. Meth. in Appl. Math.* – 2010. – V. 10, no. 1. – P. 87–94.
11. Мартыненко С. И. К вопросу о сходимости универсальной многосеточной технологии // *Математическое моделирование*. – 2010. – Т. 22, № 10. – С. 18–34.
12. Кучеров А. Б., Макаров М. М. Метод приближенной факторизации для решения разностных смешанных эллиптических краевых задач. – В кн.: *Разностные методы математической физики / Под ред. Е.С. Николаева*. – М.: Изд-во Моск. ун-та, 1984. – С. 54–65.

Статья поступила в редакцию 23.05.2011



Сергей Иванович Мартыненко окончил в 1988 г. МВТУ им. Н.Э. Баумана. Канд. физ.-мат. наук, научный сотрудник отдела химмотологии и спецдвигателей ФГУП “Центральный институт авиационного моторостроения им. П.И. Баранова”.

S.I. Martynenko graduated from the Bauman Moscow Higher Technical School in 1988. Ph. D. (Phys.-Math.), researcher of department for chemistry of motor combustive-lubricating materials and special motors of the Federal State Unitary Enterprise “Central Institute of Aviation Motors n.a. P.I. Baranov”.