Алгоритм агента:

1. Получение входных данных.
2. Формирование адреса узла сети.
3. Если этот адрес использовался, то переход к п.1.
4. Передача по сформированному адресу вербуемого узла служебной информации.
5. Если пришел ответ от вербуемого узла, то переход к п.1.
6. Передача служебной информации соседним узлам графа B о завербованном узле.
7. Конец работы алгоритма.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Лорин Г.* Распределенные вычислительные системы. – М.: Радио и связь, 1984. – 296 с.
2. *Таненбаум Э.* Распределенные системы: принципы и парадигмы. – СПб: Питер, 2003. – 877 с.
3. *Хованское С.А., Мельник Э.В, Блуишвили И.В.* Метод организации распределенных вычислений в управляющих системах // Мехатроника, автоматизация, управление. «Новые технологии», №4. 2003.
4. *Хованское С.А., Литвиненко В.А.* Организация распределенных вычислений на основе мультиагентного подхода // Известия ТРТУ. Тематический выпуск «Интеллектуальные САПР». – Таганрог: Изд-во ТРТУ, 2007, №1(73). – С. 246-250.

УДК 681.3

О.И. Овчаренко

ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ БЫСТРЫХ ПРЯМЫХ МЕТОДОВ РЕШЕНИЯ СЕТОЧНЫХ ЭЛЛИПТИЧЕСКИХ УРАВНЕНИЙ

Прогресс в численном моделировании важных прикладных задач теории поля, гидродинамики стал возможен благодаря появлению быстрых прямых методов решения сеточных уравнений [1]. К числу этих методов следует, в первую очередь, отнести циклическую редукцию (cyclic reduction (CR)), Фурье-алгоритм (Fourier-algorithm (FA)) и типа FACR(L), предназначенных для решения сеточных эллиптических уравнений, с оценками количества арифметических операций $O(N\log_2 N)$ или даже $O(N\mathrm{loq}_2(\log_2 N))$.

Основными этапами быстрых прямых методов являются решение трехдиагональных систем уравнений и выполнение Фурье-преобразований, которые можно реализовать различными алгоритмами. Для каждого алгоритма, в свою очередь, можно распределить исходную информацию по процессорам несколькими способами. В связи с этим основные этапы быстрых прямых методов могут быть реализованы как последовательно (в одном процессоре), так и параллельно (на всем решающем поле МВС).

Поэтому актуальной становится задача не только построения параллельных алгоритмов быстрых прямых методов, но и поиск наиболее эффективных в зависимости от размерности задачи, количества процессоров, значения параметра, характеризующего быстродействие каналов связи при заданной топологии многопроцессорной вычислительной системы (МВС).

В данной работе подводятся итоги разработки экономичных параллельных алгоритмов CR, FA и FACR(L), положенных в основу консультирующей системы.

В качестве исходных данных для выбора эффективного алгоритма в консультирующей системе используются параметры, характеризующие выбранную задачу, а также аппаратные характеристики конкретной МВС:

1) размерность задачи (N);
2) количество процессоров (p), объединенных между собой каналами системы коммутации (расчеты производятся в предположении, что каждый процессор имеет свою локальную память);
3) тип топологии связи между процессорами решающего поля (полнодоступная коммутация, жесткая коммутация («линейка», «матрица», «куб»));
4) параметр, характеризующий быстродействие каналов связи (k);
5) параметры, характеризующие среднее время выполнения арифметических операций и операций обмена данными между процессорами в МВС.

В качестве величин, характеризующих "распараллеливаемость" алгоритмов прямых методов, используются коэффициенты абсолютного ускорения $S_p=T_1/T_p$ и абсолютной эффективности алгоритма $E_p=S_p/p$. Величина T_1 характеризует время выполнения лучшего последовательного алгоритма (на множестве рассматриваемых), а T_p – время выполнения алгоритма с использованием p-процессоров. При расчете времен выполнения параллельных алгоритмов помимо основных характеристик, таких как время выполнения арифметических операций и операций обмена информацией между процессорами, учитывался тип связи между процессорами. А именно, для жесткой коммутации учитывалось, что процессоры не связанные непосредственно, должны обмениваться информацией через промежуточные процессоры.

Как правило, быстрые прямые методы применимы в том случае, если разностную схему можно свести к решению векторного уравнения вида [2]:

$$-Y_{j-1}+CY_j-Y_{j+1}=F_j, \qquad 1\le j\le N\text{-}1 \tag{1}$$

$$Y_0=F_0, \qquad Y_N=F_N, \tag{2}$$

где Y – вектор неизвестных , F – заданная правая часть и C – квадратная матрица.

При рассмотрении параллельных алгоритмов быстрых прямых методов в качестве модельной рассматривается следующая краевая задача:

$$\begin{gathered} \Lambda y=-f(x), \quad x\in\omega, \quad y(x)=0, \quad x\in\gamma, \\ \Lambda=\Lambda_1+\Lambda_2, \quad \Lambda_\alpha y=y_{x_\alpha \bar{x}_\alpha}, \quad \alpha=1,2. \end{gathered} \tag{3}$$

Необходимо найти решение разностной задачи Дирихле для уравнения Пуассона (3) в прямоугольнике $G=(0\le x_\alpha\le l_\alpha, \alpha=1,2)$ на прямоугольной сетке $\varpi=(x_{ij}=(ih_1, jh_2\in G, 0\le i\le N_1, 0\le j\le N_2, h_1=l_1/N_1, h_2=l_2/N_2)$ с границей γ. Задача (3) может быть записана в виде векторного уравнения (1) с граничными условиями (2).

При реализации прямых методов на МВС возможны несколько подходов к распределению сеточной информации по процессорам.

Первый способ предполагает разрезание исходных данных на блоки по OX_1, где $x_1(i)=ih_1$, i=l, 2, …, N_1-1. При втором способе исходный массив "режется" по OX_2, где $x_2(j)=jh_2$, j=l, 2,…,N_2-1.

Для метода циклической редукции построены два семейства параллельных алгоритмов, характеризующихся различными способами распределение сеточной информации. Каждое семейство, в свою очередь, состоит из двух групп, характеризующихся различными способами и методами решений систем линейных алгебраических уравнений (СЛАУ), потребность в решении которых возникает на отдельных этапах метода.

Реализация серий СЛАУ может осуществляться двумя способами:

- ♦ серии СЛАУ для различных значений j(i) решаются параллельно на всех процессорах, а отдельная СЛАУ для фиксированного j(i) реализуется в одном процессоре обычным последовательным алгоритмом;
- ♦ серии систем уравнений для различных значений j решаются последовательно, а отдельная СЛАУ для фиксированного j(i) реализуется на всех процессорах параллельным алгоритмом.

Для обозначения параллельных алгоритмов введем индекс, характеризующий способ реализации СЛАУ: (1) – последовательный, (p) – параллельный.

Существует достаточно много методов решения СЛАУ с трехдиагональной матрицей, но в данном случае выбор остановлен на двух наиболее эффективных и часто используемых на практике методах циклической редукции (CR_1) и прогонки (PR_1). Для второго способа реализации СЛАУ рассмотрены параллельные аналоги предложенных выше последовательных методов, а именно параллельный алгоритм циклической редукции (CR_p) и метод параллельной прогонки (PR_p) Яненко-Коновалова [3].

Поиск оптимальных параллельных алгоритмов заключается в их поэтапном сравнении по различным критериям (способам распределения сеточной информации, методам и способам решения СЛАУ).

В результате первого этапа сравнения получается, что при параллельном способе решения СЛАУ эффективнее оказываются алгоритмы из семейства, использующего первый способ распределения информации, а при последовательном способе решения – второй. Для оценки качества параллельных алгоритмов используются характеристики, введенные ранее. На основе анализа коэффициентов абсолютного ускорения S_p и эффективности E_p (рис. 1) выбраны оптимальные алгоритмы для различных значений N, p и k.

Еще одним перспективным методом решения сеточных эллиптических задач является Фурье-алгоритм. Классический вариант Фурье-алгоритма, в качестве основного этапа использует алгоритм дискретного преобразования Фурье. При поиске эффективных алгоритмов для выполнения Фурье-преобразований на отдельных этапах прямых методов, следует учитывать тот факт, что последовательность является не только вещественной, но и обладает дополнительной симметрией.

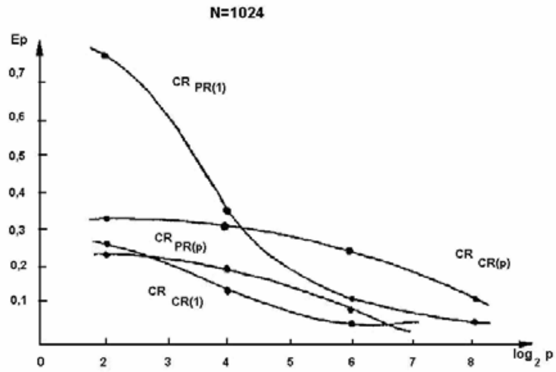

Рис. 1. Графики изменения абсолютной эффективности для параллельных алгоритмов метода циклической редукции

Наиболее эффективными оказываются алгоритмы, специально разработанные для вычисления симметрических преобразований, учитывающие особенности последовательности, а также ряд других факторов и являющиеся экономичными по количеству операций. Один из таких алгоритмов, получивший название метода разложения по синусам, подробно изложен в [3]. Единственным недостатком является то, что алгоритмы, полученные таким способом, практически не поддаются распараллеливанию. Поэтому реализацию алгоритмов такого типа имеет смысл рассматривать в одном процессоре.

Свободным от указанного недостатка является подход, основанный на использовании быстрого преобразования Фурье (БПФ) вещественных последовательностей, которые являются аналогами комплексных преобразований. Один из таких алгоритмов, известный как алгоритм Доллимора [4], по объему вычислений уступает методу разложения по синусам, но регулярная структура связей дает возможность рассматривать его реализацию как в одном процессоре, так и на всем решающем поле МВС. Очевидно, что для Фурье-алгоритма существует значительно больше параллельных алгоритмов, что объясняется наличием этапа, связанного с преобразованием последовательностей.

Введем обозначения параллельных алгоритмов в зависимости от методов и способов выполнения различных этапов. Верхний индекс будет означать следующие алгоритмы выполнения БПФ: I – метод разложения по синусам; II – алгоритм Доллимора; III – параллельный алгоритм Доллимора. Методы и способы решения СЛАУ остаются прежними.

Сравнение параллельных, алгоритмов было проведено по следующим критериям: способу "разрезания" сеточной области; способам и методам решения СЛАУ; способам и алгоритмам реализации БПФ.

Первый этап сравнения показал, что первый способ распределения информации по процессорам оказывается предпочтительнее для алгоритмов, использующих последовательное выполнение Фурье-преобразований. Дальнейшее сравнение показало, что наилучшими являются алгоритмы, использующие последовательные алгоритмы выполнения БПФ и СЛАУ. В качестве примера, на рис. 2 приведены графики изменения величины E_p в зависимости от p для N=1024.

Минимальное число операций достигается комбинированием Фурье-алгоритма и циклической редукции в алгоритме FACR(L) , где L-шаг, на котором останавливается редукция. Данный параметр можно варьировать для минимизации числа арифметических операций и операций обменов.

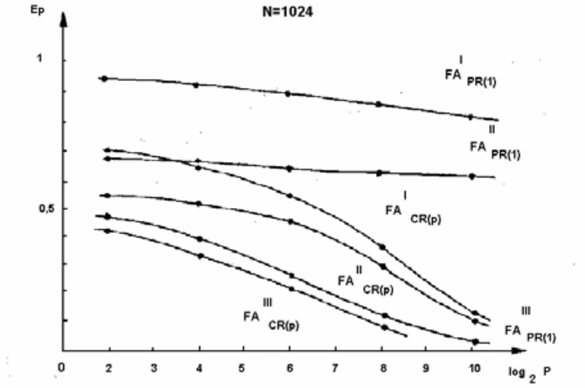

Рис. 2. Графики изменения величины E_p для параллельных алгоритмов метода разложения по базису

Построенные эффективные параллельные алгоритмы методов типа FACR(L) характеризуются различными способами и методами выполнения отдельных этапов. Очевидно, что задача поиска оптимального FACR(L)-алгоритма существенно усложняется по сравнению с методами его составляющими, так как "поведение" алгоритма зависит не только от параметров N, p и k, но и от L. Поэтому сравнение FACR (L)-алгоритмов было проведено в два этапа [5].

Первый этап заключается в поиске оптимального FACR(L)-алгоритма в пределах каждой группы алгоритмов, характеризующихся одинаковыми алгоритмами и способами выполнения БПФ и СЛАУ, но различными значениями L. Для этого используются величины S_p и E_p, введенные ранее. В результате были получены оптимальные значения параметра L={3-5} для различных значений N, p и k, позволяющие по сравнению с использовавшимися ранее значениями параметра L={1,2}, повысить эффективность параллельных FACR(L)-алгоритмов на 20%-50%.

Второй этап заключается в нахождении оптимальных FACR(L)-алгоритмов среди алгоритмов, обладающими лучшими характеристиками внутри групп. Для этого вводятся величины $S^{\wedge}_p=T^{\wedge}_1/T^{\wedge}_p$ и $E^{\wedge}_p=S^{\wedge}_p/p$, где $T^{\wedge}_1$ – время выполнения лучшего последовательного алгоритма среди всех групп для фиксированного N, а $T^{\wedge}_p$ – лучшего параллельного алгоритма в данной группе.

Оптимальный FACR(L)-алгоритм для любых значений N, p и k выбирается на основе анализа величин $S^{\wedge}_p$ и $E^{\wedge}_p$. На рис. 3 изображены графики изменения величины $E^{\wedge}_p$ для N=1024 и различных значений p. Данные графики характеризуют изменение величины $E^{\wedge}_p$ для целого семейства оптимальных FACR(L)-алгоритмов и выделены области равных значений L.

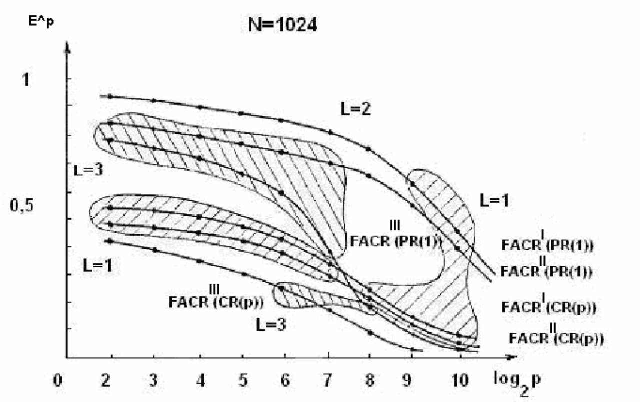

Рис. 3. Графики изменения величины $E^{\wedge}_p$ для параллельных FACR(L)-алгоритмов

При оценке временных затрат при реализации алгоритмов на топологиях: «кольцо», «матрица» и «куб» учитывалось, что процессоры не связанные непосредственно, должны обмениваться информацией через промежуточные процессоры. Анализ значений величин $S^{\wedge}_p$ и $E^{\wedge}_p$, полученных для оценки качества параллельных алгоритмов быстрых прямых методов на данных топологиях, позволяет сделать следующие выводы:

- наибольших значений величины $S^{\wedge}_p$ и $E^{\wedge}_p$ достигают при реализации на топологии "куб", а наименьших на топологии "кольцо";
- при реализации параллельных алгоритмов быстрых прямых методов на топологиях "кольцо" и "матрица" количество процессоров следует вы-

бирать в диапазоне p={16-1024}, а для топологии "куб" – выбор p неограничен;

- использование полученных оптимальных значений L={3-6} для параллельных FACR(L)-алгоритмов позволяет сократить время решения сеточных эллиптических задач в среднем на 50%-80% по сравнению с использовавшимися ранее значениями L={1-2} .

Полученные оценки и построенные параллельные алгоритмы, положенные в основу консультирующей системы по выбору наилучшего параллельного алгоритма для решения сеточных эллиптических задач, позволяют:

- выбрать наилучший алгоритм (из семейства предложенных параллельных алгоритмов) для конкретной задачи и МВС;
- определить оптимальное количество процессоров для решения задачи с учетом ее размерности, быстродействия каналов связи и топологии МВС;
- выбрать оптимальное количество шагов циклической редукции L в FACR(L)-алгоритме, позволяющее существенно сократить время решения сеточных эллиптических задач.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Swarztrauber, P.N.* (1977). The Methods of Cyclic Reduction, Fourier Analysis, and the FACR Algorithm for the Discrete Solution of Poisson's Equations on the Rectangle, SIAM ftev., vol. 19, pp. 490-501.
2. *Самарский А.А., Николаев Е.С.* Методы решения сеточных уравнений. – М.: Наука, 1978. – 592 с.
3. *Яненко Н.Н., Коновалов А.Н., Бугров А.И., Шустов Г.Б.* Об организации параллельных вычислений и "распараллеливании" прогонки // Численные методы механики сплошной среды. – 1978, №7. – С.139-146.
4. *Dollimore J.* - J. Inst. Math. Appl., 1973, vol.12, pp. 115-117.
5. *Овчаренко О.И.* Разработка параллельных FACR(L)-алгоритмов решения сеточных эллиптических уравнений // Сборник трудов IV Международной научно-практической конференции «Интеллектуальные и многопроцессорные системы». – Таганрог: НИИ МВС, 2004, Т.1. – С. 270-273.