КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ

УДК 004.008

АНАЛИЗ РАСПАРАЛЛЕЛИВАНИЯ ВЫЧИСЛЕНИЙ ПРИ РЕШЕНИИ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ СТАЦИОНАРНЫХ ФИЗИЧЕСКИХ ПОЛЕЙ МЕТОДОМ КОНЕЧНЫХ ЭЛЕМЕНТОВ

Н.И. КОРСУНОВ
С.К. ДЕДЮЛИН

Белгородский государственный национальный исследовательский университет

e-mail:
korsunov@intbel.ru
d__sergey@list.ru

В статье исследуется возможность распараллеливания вычислений при решении дифференциальных уравнений в частных производных в пространстве итераций, что позволяет связать изучение параллельной структуры программ с лексографическим графом.

Необходимость моделирования стационарных физических полей возникает при проектировании различных приборов полей композитных электрообогревателей, радиотехнических устройств, магнитных и электрических полей постоянных токов в диэлектрической среде, электромагнитных процессов, сопровождающих атмосферный разряд, и в других областях науки и техники.

Ключевые слова: нейронные сети, метод конечных разностей, расчет конструкций.

В статье исследуется возможность распараллеливания вычислений при решении дифференциальных уравнений в частных производных в пространстве итераций, что позволяет связать изучение параллельной структуры программ с лексографическим графом.

Необходимость моделирования стационарных физических полей возникает при проектировании различных приборов полей возникает при проектировании различных приборов композитных электрообогревателей, радиотехнических устройств, магнитных и электрических полей постоянных токов в диэлектрической среде, электромагнитных процессов, сопровождающих атмосферный разряд и в других областях науки и техники [1,2]. Моделирование стационарных физических полей сводиться к решению дифференциальных уравнений математической физики [2].

Получение точного аналитического решения дифференциальных уравнений в частных производных с различными граничными условиями с коэффициентами взывает значительные затруднения. По этой причине используется методы приближенного численного решения, к которым относиться метод конечных элементов, приводящих к решению систем линейных алгебраических уравнений [3].

Приближенное решение уравнений в частных производных имеется в виде [1]

$$U = F + \sum_{m} A_m N_m ,$$

F – функция, удовлетворяющая граничным условиям,
N_m – базисные функции, которые на границе обращаются в ноль,
A_m – искомые коэффициенты. Имеется такой набор коэффициентов, при котором

$$w_n RdS = 0,$$

R – приближенное решение,
w_n – некоторые весовые функции.

Для поиска весовых функций используется метод Галеркина [2], основанный на выборе пробных функций и поиске решений по всей области.

В методе конечных элементов невязка минимизируется внутри отдельных подобластей D простой формы – конечных элементов с велением ограничений на вид базисных функций [3]: обращение в единицу каждой базисной функции в одним из узлов аппроксимации и не равной нулям только в конечных элементах, содержащих данный узел.

В вариационной формулировке метода конечных элементов определяющим уравнением двумерной задачи является уравнение Лапласа [1]:

$$\Delta T^2 = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \qquad (1)$$

с граничным условиями Дирихле на части границ

$$T=50, y=0 \ (2), T=100, y=2 \qquad (2)$$

и условиями Неймана на остальной части границы задача определена на множестве R=D+S и границе S.

В [1] методом вариационного исчисления показано что решение [1] с граничным условием (2) совпадает с функцией минимизизирующий функционал.

$$X = \frac{1}{2}\iint_D \left(\frac{\partial T}{\partial x}\right)^2 + \left(\frac{\partial T}{\partial y}\right)^2 dxdy, \qquad (3)$$

где T(x,y) – функция из допустимого множества пробных функций, заданный в D. Принимая пробные функции непрерывными с кусочно-непрерывными первыми производными, область D разбивается треугольными элементами на l конечных подобластей с общим числом узлов n, это позволяется записать (1) в виде

$$X = \sum_{i=1}^{l} X^{e_i}, \ X^{l_i} == \frac{1}{2}\iint_{l_i} \left[\left(\frac{\partial T^{l_i}}{\partial x}\right) + \left(\frac{\partial T^{l_i}}{\partial y}\right) dxdy\right] \qquad (4)$$

Обозначая вершины треугольника в порядке движения по часовой стрелке $i(x_i,y_i)$, $m(x_m,y_m)$, $j(x_j,y_j)$ и выбирая для l_i элемента линейную пробную функцию

$$T^{l_i}(x.,y) = a_1^{l_i} + a_2^{l_i} + a_3^{l_i}\,; x, y \in l_i, \qquad (5)$$

пробные функции для узлов i,m,j представляются в виде

$$\begin{aligned} \overline{T_i} &= a_1 + a_2 x_i + a_3 y_i \quad , \\ \overline{T_j} &= a_1 + a_2 x_j + a_3 y_j \quad , \\ \overline{T_m} &= a_1 + a_2 x_m + a_3 y_m \quad . \end{aligned} \qquad (6)$$

При известных значений T в узлах i,j,m и постоянных $\alpha_1, \alpha_2, \alpha_3$ система (6) имеет единственное решение a_i, a_j, a_m [1]:

$$\begin{aligned} \alpha_1 &= \left(\frac{1}{2\Delta}\right)\left(a_i \overline{T_i} + a_j \overline{T_j} + a_m \overline{T_m}\right), \\ \alpha_2 &= \left(\frac{1}{2\Delta}\right)\left(b_i \overline{T_i} + b_j \overline{T_j} + b_m \overline{T_m}\right), \\ \alpha_3 &= \left(\frac{1}{2\Delta}\right)\left(c_i \overline{T_i} + c_j \overline{T_j} + c_m \overline{T_m}\right), \end{aligned} \qquad (7)$$

где Δ – определитель системы (6),

$$\begin{aligned} a_i &= x_j y_m - x_m y_j, \\ b_i &= y_j - y_m, \\ c_i &= x_m - x_j \quad . \end{aligned}$$

В этом случае $T^{e_i}(x,y)$ представляются через базисные функции в виде

$$T^{l_i}(x,y) = \frac{1}{2\Delta}\left[\left(a_i + b_i x + c_i y\right)\overline{T_i} + \left(a_j + b_j x + c_j y\right)\overline{T_j} + \left(a_m + b_m x + c_m y\right)\overline{T_m}\right], \qquad (8)$$

которые в векторной форме представляются в виде

$$T^{l_i} = NT; N = \left[N_1 \ldots N_n\right], T^{l_i} = \left[\overline{T_1} \ldots \overline{T_n}\right]^t \qquad (9)$$

и при этом

$$\begin{aligned} \frac{\partial T^{l_i}}{\partial x} &= \frac{1}{2\Delta}\left(b_i \overline{T_i} + b_j \overline{T_j} + b_m \overline{T_m}\right), \\ \frac{\partial T^{l_i}}{\partial y} &= \frac{1}{2\Delta}\left(c_i \overline{T_i} + c_j \overline{T_j} + c_m \overline{T_m}\right). \end{aligned} \qquad (10)$$

Здесь $[\ \ ]^t$ – транспонированная матрица.

С учетом (10) выражение (4) представляется как

$$X^{l_i} = \frac{1}{8\Delta^2}\iint_{l_i}\left[\left(b_i\overline{T_i} + b_j\overline{T_j} + b_m\overline{T_m}\right)^2 + \left(c_i\overline{T_i} + c_j\overline{T_j} + c_m\overline{T_m}\right)^2\right]dxdy, \qquad (11)$$

которое вследствие $\iint_{l_i} dxdy = \Delta$ дает

$$X^{l_i} = \frac{1}{8\Delta}\left[\left(b_i\overline{T_i} + b_j\overline{T_j} + b_m\overline{T_m}\right) + \left(c_i\overline{T_i} + c_j\overline{T_j} + c_m\overline{T_m}\right)\right]. \qquad (11)$$

С учетом (11) функционал (4) может быть представлен функцией всех узловых значений в виде

$$X = X\left(\overline{T_1},\ldots,\overline{T_n}\right) \qquad (12)$$

и определению подлежат параметры $\overline{T_1},\ldots,\overline{T_n}$.

Минимум (12) может быть определен как

$$\frac{dx}{d\overline{T_p}} = \sum_{i=1}^{l}\frac{dx^{l_i}}{d\overline{T_p}} = 0, p = 1,\ldots,n. \qquad (13)$$

Дифференцирование (11) по $\overline{T_p}$ определяет вклад элемента l_i по соотношению

$$\frac{dx^{l_i}}{d\overline{T_p}} = \frac{1}{4\Delta}\left[b_p\left(b_p\overline{T_p} + b_q\overline{T_q} + b_r\overline{T_r}\right) + c_p\left(c_p\overline{T_p} + c_q\overline{T_q} + c_r\overline{T_r}\right)\right] \qquad (14)$$

при следующем условии: номера узлов i,j,m элемента l_i имеют во множестве номеров узлов системы значения p,q и r соответственно.

В соответствии с (7), (8), (9), (11), (14) решение уравнения (1) сводиться к формированию матрицы жесткости (6) и объединению матрицы жесткости по узлам. При этом алгоритмы численного решения содержат вложенные циклы, первый из которых вычисляет заданные, используемые вторым циклом, а данные из второго цикла используются в третьем. Следовательно, все три цикла выполняются последовательно.

Получение решения дифференциальных уравнений в частных производных средствами вычислительной техники выдвигает необходимость обеспечения высокого быстродействия, что может быть достигнуто распараллеливанием вычислений [4].

Подавляющее большинство методов, позволяющих распараллелить вычисления такого рода, в большей или меньшей степени плохо масштабируются и накладывают определенные ограничения на вид обрабатываемой матрицы [5]. Для получения эффекта работы параллельных алгоритмов необходимо сильно разряженная матрица, а метод конечных элементов в большинстве случаев не дает таких матриц. Следствием этого являются необходимость анализа последовательных программ на выделение фрагментов выполняющихся параллельно.

Текст программы, реализующий метод конечных элементов путем формирования матрицы жесткости отдельных элементов и объединения их по узлам приведены в [6].

Использование суперкомпьютеров с общей или расширенной памятью требуют решения вопроса распараллеливания данной программы. В анализе последовательных программ на возможность их распараллеливания можно использовать несколько методов. В настоящее время находим широкое применение метод построения информационного графа программы и последующего его анализа, в результате которого выделяются циклы с независимыми итерациями. [4]

В соответствии с данным методом по тексту последовательной программы формируется циклический профиль, представляемый горизонтальными скобками и помеченной скобки, соответствующие независимым циклам. Каждый из циклов с внутренними операторами преобразователями, изменяющими значения переменных, ставят в соответствие вершинам информационного графа, ребра которого соответствуют условиям срабатывания операторов.

Выделив в информационном графе вершины, являющиеся итерационно-независимыми определяют возможность параллельного выполнения фрагментов последовательной программы.

Применим приведенную методику [4] к анализу распараллеливания последовательной программы решения дифференциальных уравнений методом конечных элементов [6]. Прежде всего, строится информационный граф, используемый для определения информационно независимых вершин. Так как программа [6] содержит много циклов, то, не теряя общности, проведем анализ фрагментов формирования матрицы жесткости отдельных элементов и их объединения по узлам.

Фрагмент программы формирования матрицы жесткости элементов с разделением на отмечаемые отдельные следующие друг за другом части представляется следующим образом.

```
for (int i=0;i<nElems;i++)
{
        for(int j=0;j<3;j++) {
                int iNodeJ=elems[i,j];
                x[j]=nodes[iNodeJ].x;
                j[j]=nodes[iNodeL].y;
        }
}
for (int J=0;J<3;J++)
{
int iNode1=J+1;
int uNode2=J+2;
if (iNode1>=3) {
        iNode1=0;
        iNode2=1;
} else if (iNode1 == ) {
        iNode2=0;
}
b[j]=y[iNode1]-y[iNode2];
c[j]=x[iNode2]-x[iNode1];
}
double delta=(c[2]*b[2]-c[1]*b[2])/2.0;
for (int ir=0;ir<3;ir++) {
        for (int ic=0;ic<3;ic++) {
                elemsM[i,ir,ic]=(b[ir]*b[ic]+c[ir]*c[ic])/(4.0*delta);
        }
}
```

Обозначая операторы-преобразователи прямоугольником и стрелками отношения информационной зависимости между операторами, информационный граф фрагмента программы представляется в виде, приведенном на рис. 1.

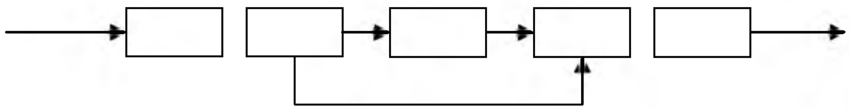

Рис 1. Информационный граф фрагмента программы формирование жесткости отдельного элемента

Из данного фрагмента и его информационного графа следует, что первый вложенный цикл вычисляет данные, которые затем используются вторым циклом, а данные второго цикла используется третьим циклом, т.е. все три вложенных циклов работают последовательно.

Фрагмент кода программы объединения матрицы жесткости всех элементов по узлам.

```
for (int iNode=0;iNode<nNodes;iNode++) {
        boul isBoundary = false;
        for (int i=0;i<iBounds;i++) {
                if (iNode == boundNodes[i]) {
                        stm[iNode,iNode]=1;
                        rigthP[iNode]=bounds[i];
                        isBoundaty=true;
                        break;
                }
        }
        if (isBoundary == false) {
        int nSurElems = arrSurrNodes[iNode].Count;
        for (int ind=0;intd<nSurElems;ind++) {
                int iElem = (int) arrSurrNodesElems[iNode][ind];
                for (int ir=0;ir<3;ir++) {
                        if (elems[iElem,ir]==iNode) {
                                for (int ic=0;ic<3;ic++) {
                                        int iCurrNode = elems[iElem,ic];
                                }
                        }
                }
        }
}
```

Из данного фрагмента программы следует, что на каждой итерации цикла вычисляется очередная строка общей матрицы жесткости и все итерации внешнего цикла выполняются независимо друг от друга.

Анализ информационного графа показывает, что при решении дифференциальных уравнений в частный производный методом конечных элементов допустимо параллельное выполнение двух фрагментов программы: фрагмент, формирующий матрицы жесткости для каждого элемента и фрагмент, объединения матрицы жесткости по узлам.

Следствии этого является высокая эффективность решений дифференциальных уравнений в частных производных методом конечных разностей векторного конвейерными компьютерами и неэффективное использование многопроцессорных компьютеров.

Таким образом, повешение эффективность распараллеливание вычислений при решении дифференциальных уравнений в частных производных методом конечных элементов требует разработки новых численных методов позволяющих использовать все возможности мощных суперкомпьютеров.

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры для инновационной России» на 2009-2013 годы, гос. контракт № 14.740.11.0390.

Список литературы

1. Тихонов А.Н., Самарский А.А., Уравнения математической физики. – М: Изд-во МГУ; изд-во «Наука», 1999.
2. Арутюнов В.А., Бухмиров В.В., Крупненников С.А., Математическое моделирование тепловой работы промышленных печей. – М: Металлургия, 1990.
3. Численные методы [Текст] / Н.С. Юахвалов, Н.П. Жидков, Г.М. Кобельников. – М: Физмат, 2002.
4. Воеводин В.В., Воеводин Вл.В., Параллельные вычисления. – Спб: БХВ – Петербург, 2002г.
5. Бухалов А.А., Дациок В.Н., Жигулов А.И., Программирование многопроцессорных систем.
6. Арфкен Г. Математические матоды в физике. – М: Атомиздат, 1969.

THE ANALYSIS OF MULTISEQUENCING OF CALCULATIONS AT DECISIONS OF DIFFERENTIAL EQUATIONS OF STATIONARY PHYSICAL FIELDS THE FINITE-ELEMENT METHOD

N.I. KORSUNOV
S.K. DEDULIN

Belgorod National Research University

e-mail:
korsunov@intbel.ru
d__sergey@list.ru

In article possibility of multisequencing of calculations is researched at the decision of differential equations in private derivative iterations in space that learning of parallel structure of programs with лексографическим the graph allows to connect.

Necessity of modeling of stationary physical fields arises at designing of various instruments of fields arises at designing of various instruments of composite electroheaters, radio engineering devices, magnetic and electric fields of constant currents in the dielectric environment, the electromagnetic processes accompanying atmospheric discharge and in other areas of a science and technicians.

Key words: neuronnet, method of finite differences, calculation of construction.