

ПАРАЛЛЕЛЬНЫЕ БЛОЧНЫЕ МНОГОШАГОВЫЕ АЛГОРИТМЫ ЧИСЛЕННОГО РЕШЕНИЯ СИСТЕМ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ БОЛЬШОЙ РАЗМЕРНОСТИ

Дмитриева О.А.

Кафедра ПМИИ ДонГТУ

E-mail: dmitriv@r5.dgtu.donetsk.ua

Abstract

Dmitrieva O.A. Parallel block multisteps algorithms of numeral systems decision of usual differential equalizations of big dimensions. Lead error estimations for multisteps block methods, allowing to find the systems decisions of usual differential equalizations on parallel calculable systems with set exactness degree. In offered methods decision is for hours together for all of block points. The coefficients of equalizations for blocks with any amount of points determine-by the medium of packet Mathematica. Bowed modeling out of decision for diverse dimensions of blocks. Are got estimations, describing quality of worked up methods: acceleration coefficients and effectiveness

Введение

Существует широкий круг научно-технических задач, решение которых невозможно без использования высокопроизводительных вычислительных систем. К таким задачам можно отнести, в первую очередь, системы дифференциальных уравнений, как обыкновенных, так и в частных производных. Поскольку эти задачи, а это, в основном, задачи большой размерности, выдвигают повышенные и специфические требования к продуктивности обработки информации, возникает необходимость в эффективной структурной и алгоритмической организации параллельных вычислений. Имеющиеся в настоящее время результаты исследований позволяют получить приемлемые решения этих и других проблем лишь в отдельных случаях, так как очень трудной является, например, задача получения решения исходной системы дифференциальных уравнений с заданной степенью точности или задача оценки погрешности найденного приближенного решения.

1. Численное решение задачи Коши для систем обыкновенных дифференциальных уравнений многошаговыми блочными методами

Рассмотрим решение задачи Коши для системы обыкновенных дифференциальных уравнений

$$\frac{dx}{dt} = \mathbf{f}(t, \mathbf{x}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}^0 = (x_1^0, x_2^0, \dots, x_m^0)^T, \quad (1)$$

где

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{pmatrix} \quad \text{вектор} \\ \text{неизвестных}, \quad \mathbf{f} = \begin{pmatrix} f_1(t, x_1, x_2, \dots, x_m) \\ f_2(t, x_1, x_2, \dots, x_m) \\ \dots \\ f_m(t, x_1, x_2, \dots, x_m) \end{pmatrix} \quad \text{заданный} \\ \text{вектор}$$

k -точечним многошаговим блочним методом. Блочним будем называть метод, при котором для блока из k точек новые k значений функции вычисляются одновременно [1]. Эта особенность методов, во-первых, согласуется с архитектурой параллельных вычислительных систем, а, во-вторых, позволяет вычислять коэффициенты разностных формул не в процессе интегрирования, а на этапе разработки метода, что значительно увеличивает эффективность счета. Множество M точек равномерной сетки, такое, что $t_M = T$ с шагом τ разобьем на блоки, содержащие по k точек, $kN = M$. В каждом блоке введем номер точки $i = 0, 1, \dots, k$ и обозначим через $t_{n,i}$ точку n -го блока с номером i . Точку $t_{n,0}$ назовем началом блока n , а $t_{n,k}$ – концом блока. Очевидно, что имеет место соотношение $t_{n,k} = t_{n+1,0}$. Условимся начальную точку в блок не включать.

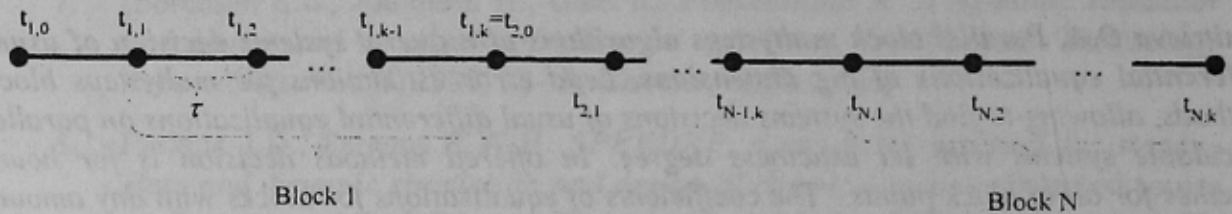


Рис. 1 Схема разбиения на блоки

Для расчета значений в новом блоке многошаговым многоточечным блочным методом будут использоваться все точки предшествующего блока, т.е.

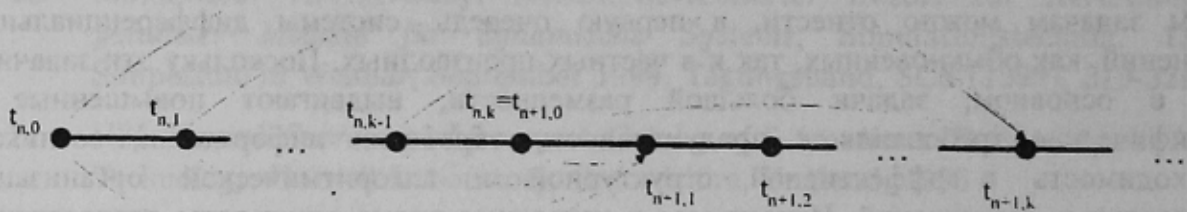


Рис. 2 Схема вычислений многошаговым блочным методом

В общем случае уравнения многошаговых разностных методов для блока из k точек с учетом введенных обозначений можно записать в виде

$$\frac{u_{n,q,i} - u_{n,q,0}}{i\tau} = \sum_{j=1}^k b_{i,j} F_{n-1,q,j} + \sum_{j=1}^k a_{i,j} F_{n,q,j}, \quad i = \overline{1, k}, \quad q = \overline{1, m}, \quad n = 1, 2, \dots \quad (2)$$

где $F_{n,q,j} = f_q(t_n + j\tau, u_1(t_n + j\tau), u_2(t_n + j\tau), \dots, u_q(t_n + j\tau))$.

2. Погрешность аппроксимации многошаговых блочных методов для систем обыкновенных дифференциальных уравнений

Выражение для погрешности аппроксимации рассматриваемого разностного метода (2) на решении уравнения (1) имеет вид

$$r_{n,q,i} = -\frac{x_{n,q,i} - x_{n,q,0}}{i\tau} + \sum_{j=1}^k b_{i,j} x'_{n-1,q,j} + \sum_{j=1}^k a_{i,j} x'_{n,q,j}, \quad i = \overline{1, k}, \quad q = \overline{1, m} \quad (3)$$

где $x_{n,q,i} = x_q(t_n + i\tau)$,

$$x'_{n,q,j} = f_q(t_{n,j}, x_{n,1,j}, x_{n,2,j}, \dots, x_{n,q,j}) =$$

$$= f(t_n + jt, x_1(t_n + j\tau), x_2(t_n + j\tau), \dots, x_q(t_n + j\tau))$$

$$x'_{n-1,q,j} = f_q(t_{n-1,j}, x_{n-1,1,j}, x_{n-1,2,j}, \dots, x_{n-1,q,j}) =$$

$$= f(t_n - (k-j)\tau, x_1(t_n - (k-j)\tau), x_2(t_n - (k-j)\tau), \dots, x_q(t_n - (k-j)\tau)).$$

Воспользуемся подходом, предложенным в [2]. Раскладывая $x_{n,q,i}$, $x'_{n,q,j}$ и $x'_{n-1,q,j}$ в ряды Тейлора в окрестности точки t_n , получим

$$\frac{x_{n,q,i} - x_{n,q,0}}{i\tau} = \sum_{l=1}^p \frac{(i\tau)^{l-1}}{l!} x_{q,n,0}^{(l)} + O(\tau^p), \quad i = \overline{1, k}, \quad q = \overline{1, m}$$

$$x'_{n-1,q,j} = \sum_{l=1}^p \frac{((j-k)\tau)^{l-1}}{(l-1)!} x_{n,q,0}^{(l)} + O(\tau^p), \quad i = \overline{1, k}, \quad q = \overline{1, m}.$$

$$x'_{n,q,j} = \sum_{l=1}^p \frac{(j\tau)^{l-1}}{(l-1)!} x_{n,q,0}^{(l)} + O(\tau^p), \quad i = \overline{1, k}, \quad q = \overline{1, m}.$$

Подставляя эти разложения в выражение для погрешности аппроксимации (3) и, учитывая, что конечная точка предыдущего блока совпадает с начальной точкой следующего блока, будем иметь

$$r_{n,q,i} = - \sum_{l=1}^p \frac{(i\tau)^{l-1}}{l!} x_{q,n,0}^{(l)} + \sum_{j=1}^{k-1} b_{i,j} \sum_{l=1}^p \frac{((j-k)\tau)^{l-1}}{(l-1)!} x_{n,q,0}^{(l)} + b_{i,k} x'_{n,q,0} +$$

$$+ \sum_{j=1}^k a_{i,j} \sum_{l=1}^p \frac{(j\tau)^{l-1}}{(l-1)!} x_{n,q,0}^{(l)} + O(\tau^p) \quad (4)$$

Сгруппируем члены с одинаковыми производными и изменим порядок суммирования в последнем выражении, тогда получим

$$r_{n,q,i} = x'_{n,q,0} \left(\sum_{j=1}^k b_{i,j} + \sum_{j=1}^k a_{i,j} - 1 \right) + \sum_{l=2}^p \frac{\tau^{l-1}}{(l-1)!} x_{n,q,0}^{(l)} *$$

$$* \left[\sum_{j=1}^k a_{i,j} * j^{l-1} + \sum_{j=1}^{k-1} b_{i,j} (j-k)^{l-1} - \frac{i^{l-1}}{l} \right] + O(\tau^p) \quad (5)$$

Отсюда следует, что погрешность аппроксимации имеет порядок p , если выполнены условия для каждого из уравнений системы (1)

$$\sum_{j=1}^k b_{i,j} + \sum_{j=1}^k a_{i,j} = 1, \quad \sum_{j=1}^k j^{l-1} a_{i,j} + \sum_{j=1}^{k-1} (j-k)^{l-1} b_{i,j} = \frac{i^{l-1}}{l}, \quad i = \overline{1, k}, \quad l = \overline{2, p}. \quad (6)$$

Система уравнений (6) для каждого фиксированного i содержит p уравнений и $2k$ неизвестных $b_{i,1}, b_{i,2}, \dots, b_{i,k}, a_{i,1}, a_{i,2}, \dots, a_{i,k}$. Потребуем, чтобы $p=2k$, тогда из системы

(6) можно будет определить все неизвестные коэффициенты. Отсюда следует, что наивысший порядок аппроксимации многошагового k - точечного блочного метода равен $2k$. Его погрешность в соответствии с (5) определяется формулой

$$r_{n,q,i} = \frac{\tau^{2k}}{(2k)!} x_{n,q,0}^{(2k+1)} \left[\sum_{j=1}^k j^{2k} a_{i,j} + \sum_{j=1}^{k-1} (j-k)^{2k} b_{i,j} - \frac{i^{2k}}{2k+1} \right] + O(\tau^{2k+1}), \quad i = \overline{1, k}, \quad q = \overline{1, m}$$

3. Алгоритм параллельного решения нелинейной разностной задачи

Для вычисления приближенных значений решения задачи Коши (1) необходимо решить нелинейную систему уравнений (2). Используем для этого следующий итерационный процесс

$$u_{n,q,i,0} = u_{n,0} + i\tau \sum_{j=1}^k c_{i,j} F_{n-1,q,j}, \quad i = \overline{1,k}, \quad q = \overline{1,m}, \quad n = 1, 2, \dots$$

$$u_{n,q,i,s+1} = u_{n,q,0} + i\tau \left(\sum_{j=1}^k b_{i,j} F_{n-1,q,j} + \sum_{j=1}^k a_{i,j} F_{n,q,j,s} \right), \quad s = \overline{0, k-1}, \quad (7)$$

который позволяет проводить вычисления параллельно для каждого узла блока. Покажем, что после выполнения k шагов вычислений по формулам (7) локальная ошибка в узлах блока будет иметь порядок $O(\tau^{2k+1})$. Обозначим через $u_{n,q,i}^{[p]}$ значение решения в точке $t_{n,i}$, найденное с локальной погрешностью $O(\tau^p)$, а через

$$F_{n,q,i}^{[p]} = f(t_{n,i}, u_{n,1,i}^{[p]}, u_{n,2,i}^{[p]}, \dots, u_{n,q,i}^{[p]}), \quad (8)$$

вычисленные при этом значения правых частей уравнений системы (1). Допустим, что значения $u_{n,q,0}$ вычислены с локальной ошибкой $O(\tau^{2k+1})$, тогда и правые части уравнений могут быть вычислены с такой же погрешностью, т. е. имеем $F_{n-1,q}^{[2k+1]}$.

Выполнив первый шаг вычислений по первой формуле (7), получим

$$u_{n,q,i,0}^{[k+1]} = u_{n,q,0}^{[2k+1]} + i\tau \sum_{j=1}^k c_{i,j} F_{n-1,q,j}^{[2k+1]}, \quad i = \overline{1,k}, \quad q = \overline{1,m}$$

так как локальная ошибка экстраполяционной формулы Адамса имеет порядок $O(\tau^{k+1})$. Параллельные вычисления по второй формуле (7) при $s = 0$ дадут следующие результаты:

$$u_{n,q,i,1}^{[k+2]} = u_{n,q,0}^{[2k+1]} + i\tau \left(\sum_{j=1}^k b_{i,j} F_{n-1,q,j}^{[2k+1]} + \sum_{j=1}^k a_{i,j} F_{n,q,j,0}^{[k+1]} \right), \quad i = \overline{1,k}, \quad q = \overline{1,m}$$

так как функции $F_{n,q,j,0}^{[k+1]} = f(t_{n,j}, u_{n,1,0}^{[k+1]}, u_{n,2,0}^{[k+1]}, \dots, u_{n,q,0}^{[k+1]}), j = \overline{1,k}$ могут быть вычислены с локальной ошибкой $O(\tau^{k+1})$. Вычисления на следующем шаге улучшат точность результатов на один порядок. Продолжая вычисления до $s = k-1$, получим результаты, соответствующие предельным точностям приближенных формул (7), т. е.

$$u_{n,q,i,k}^{[2k+1]} = u_{n,q,0}^{[2k+1]} + i\tau \left(\sum_{j=1}^k b_{i,j} F_{n-1,q,j}^{[2k+1]} + \sum_{j=1}^k a_{i,j} F_{n,q,j,2k-1}^{[2k]} \right), \quad i = \overline{1,k},$$

поскольку разностные уравнения (2) аппроксимируют систему дифференциальных уравнений (1) с порядком $O(\tau^{2k})$, дальнейшие шаги не дадут повышения порядка точности результатов.

4. Реализация многошагового блочного метода на параллельных вычислительных структурах

Будем считать, что начальные разгонные значения получены любым из методов, поддерживающим заданную точность. Начальные приближения для нового блока находятся с помощью метода Адамса-Башфорта [3]. Условимся, что к началу выполнения расчетов матрицы коэффициентов уже сформированы, исходя из соотношений (6). Модель, на которую ориентируется решение, имеет следующие

особенности: используется вычислительная система SIMD структуры с вычислительным полем $2D$ – тор, размерностью не менее $k \times m$ процессорных элементов (k – размерность блока, m – размерность системы обыкновенных дифференциальных уравнений). Каждый процессорный элемент может выполнить любую арифметическую операцию за один такт; временные затраты, связанные с обращением к запоминающему устройству, отсутствуют. Для эффективной работы методов используется алгоритм систолического умножения матрицы на вектор, который подробно рассмотрен в [4].

Определение характеристик параллелизма осуществлялось с помощью пакета *Mathematica*® (Wolfram Research Inc.). Для оценки ускорения рассчитывалось время вычисления вектора x_n на однопроцессорной ЭВМ с быстродействием арифметического процессора и объемом ОЗУ, равным суммарному объему всех 3У арифметических процессоров, и с необходимым числом внешних устройств, имеющих скорости обмена такие же, как в многопроцессорной вычислительной системе типа SIMD. Поскольку заранее неизвестно число операций, требуемое для вычисления значений $f_{n,q}(x_{n,1}, x_{n,2}, \dots, x_{n,m})$, $q = \overline{1, m}$, $n = 1, 2, \dots$, вводилась характеристика максимального времени, необходимого для вычисления правой части t_f [5].

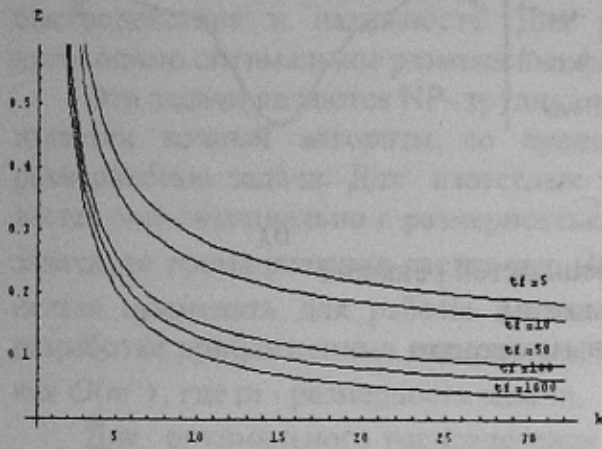


Рис. 3 Зависимость эффективности от размерности блока ($m=100$) для решетки процессоров

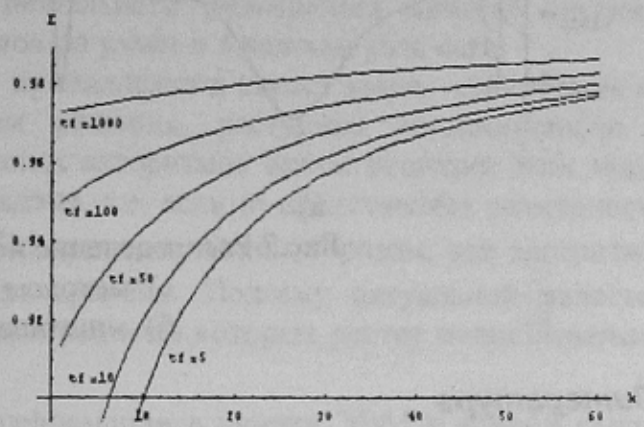


Рис. 4 Зависимость эффективности от размерности блока ($m=100$) для линейки процессоров

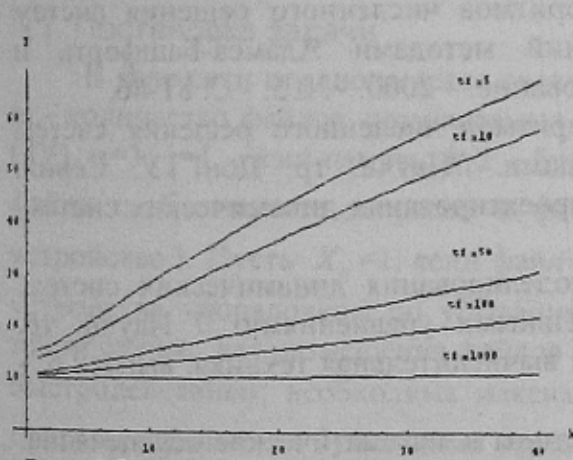


Рис. 5 Зависимость ускорения от размерности блока ($m=100$) для решетки процессоров

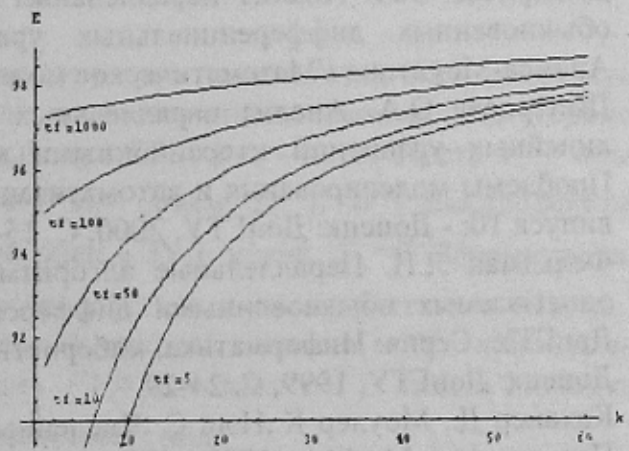


Рис. 6 Зависимость ускорения от размерности блока ($m=100$) для линейки процессоров

Заключення

Оценки эффективности использованных методов, реализованных на линейке, значительно превосходят те же показатели на решетчатых структурах, что, с одной стороны, не согласовывается с теоретическими оценками, а, с другой стороны, еще раз доказывает, что в параллельных вычислительных структурах время обменов данными может составлять большую часть общего времени решения задачи. Поэтому подходить к выбору структуры вычислительной системы для решения каждой конкретной задачи необходимо не только с позиций наращивания количества процессорных элементов, но и оптимизации алгоритмов для сокращения количества обменов между ними.

Решение тестовых задач [6] показало, что многошаговые блочные методы превосходят по точности решения известные методы Ньютона, а по такому показателю как трудоемкость, они требуют значительно меньшего числа операций.

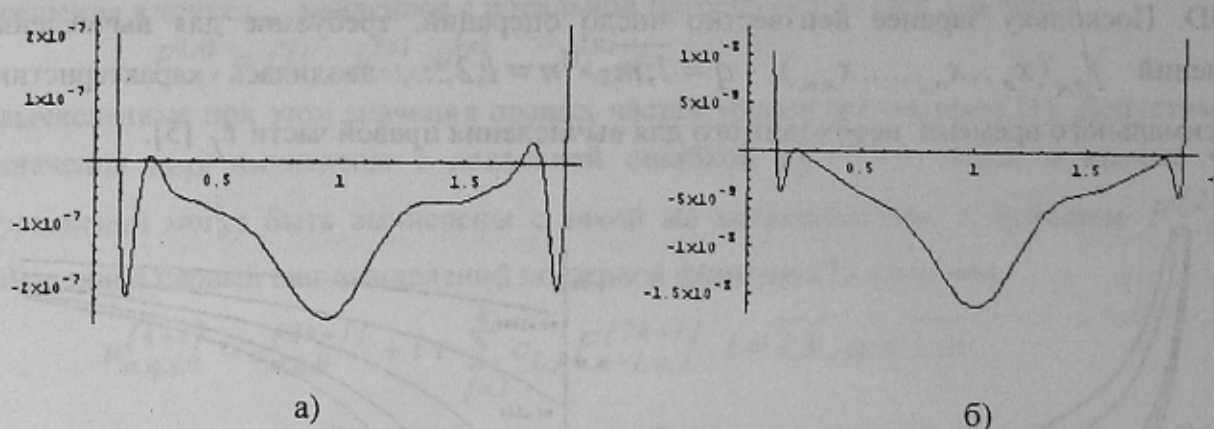


Рис.7 Распределение погрешностей решения

а) методом Ньютона

б) многошаговым методом

Литература

1. Системы параллельной обработки: Пер. с англ./ Под. ред. Ивенса Д. - М.: Мир, 1985, 416с.
2. Самарский А.А., Гулин А.В. Численные методы. - М.: Гл. ред. физ.-мат. лит., 1989, 432 с.
3. Дмитриева О.А. Анализ параллельных алгоритмов численного решения систем обыкновенных дифференциальных уравнений методами Адамса-Башфорта и Адамса-Моултона //Математическое моделирование. - 2000. - № 5. - С. 81-86.
4. Дмитриева О.А. Анализ параллельных алгоритмов численного решения систем линейных уравнений итерационными методами. //Научн. тр. ДонГТУ. Серия: Проблемы моделирования и автоматизации проектирования динамических систем, выпуск 10: - Донецк: ДонГТУ, 2000, С. 15-22.
5. Фельдман Л.П. Параллельные алгоритмы моделирования динамических систем, описываемых обыкновенными дифференциальными уравнениями. // Научн. тр. ДонГТУ. Серия: Информатика, кибернетика и вычислительная техника, выпуск 6: - Донецк: ДонГТУ, 1999, С. 24-29.
6. Каханер Д., Моулер К., Нэш С. Численные методы и математическое обеспечение: Пер. с англ. - М.: Мир, 1998.- 575 с.