

УДК 004.932.4

Р.Ю.Никишин , И.А. Назарова , Л.П.Фельдман
Донецкий национальный технический университет

КЛАСТЕРНАЯ РЕАЛИЗАЦИЯ ПАРАЛЛЕЛЬНЫХ ИТЕРАЦИОННЫХ МЕТОДОВ РЕШЕНИЯ ЖЕСТКИХ ЗАДАЧ КОШИ

Аннотация. Никишин Р.Ю., Назарова И.А., Фельдман Л.П. Кластерная реализация параллельных итерационных методов решения жестких задач Коши рассматриваются параллельные алгоритмы решения нелинейной задачи Коши. выполняется анализ решения одного дифференциального уравнения и системы таких уравнений большой размерности. Приведены особенности выполнения параллельной реализации алгоритмов. Произведено сравнение аналитических и полученных экспериментально характеристик, а также определена эффективность параллельной реализации.

Ключевые слова: задача Коши, система обыкновенных дифференциальных уравнений, полностью неявные методы типа Рунге-Кутты.

Постановка проблемы.

Применение параллельных вычислительных систем является стратегическим направлением развития современной компьютерной индустрии. Это обстоятельство вызвано не столько принципиальным ограничением максимально возможного быстродействия последовательных машин, сколько наличием вычислительных задач, для решения которых возможностей существующих средств вычислительной техники является недостаточным. К таким задачам относится решение систем обыкновенных дифференциальных уравнений большой размерности, используемое при моделировании различных динамических процессов с сосредоточенными параметрами.

В докладе рассматривается реализация полностью неявных методов типа Рунге-Кутты на кластере.

Анализ реализации ПНМРК

Для системы из m обыкновенных дифференциальных уравнений:

$$\begin{cases} \bar{y}' = F(x, \bar{y}); \\ \bar{y}(x_0) = \bar{y}_0. \end{cases} \quad (1)$$

где $\bar{y} = (y_1, y_2, \dots, y_m)^T$, $F = (f_1, f_2, \dots, f_m)^T$ - правая часть есть в общем случае нелинейная функция $F = \bar{f}$, задающая отображение: $F : R \times R^m \rightarrow R^m$.

Одношаговый многостадийный метод для решения нелинейной задачи Коши, описываемой СОДУ (1), имеет вид:

$$\begin{cases} \bar{y}_{n+1} = \bar{y}_n + h \cdot \sum_{i=1}^s b_i \cdot \bar{k}_i, \\ \bar{k}_i = F(x_n + c_i h, \bar{y}_n + h \sum_{j=1}^s a_{ij} \bar{k}_j), \quad i = \overline{1, s} \end{cases}, \quad (2)$$

где s -размерные вектора $c = (c_1, c_2, \dots, c_s)^T$, $b = (b_1, b_2, \dots, b_s)^T$ и $s \times s$ -матрица $A = (a_{ij})$, $i, j = \overline{1, s}$, $a_{ij} \neq 0$ описывают уникальный вариант метода и выбираются из соображений точности.

Применение полностью неявных многостадийных методов (ПНММ) требует для определения коэффициентов $\bar{k}_i, i = \overline{1, s}$ решения системы $s \times m$, в общем случае нелинейных, уравнений:

$$\begin{cases} \bar{k}_1 = \bar{f}[x_n + c_1 h; \bar{y}_n + h(a_{11}\bar{k}_1 + a_{12}\bar{k}_2 + \dots + a_{1s}\bar{k}_s)] \\ \bar{k}_2 = \bar{f}[x_n + c_2 h; \bar{y}_n + h(a_{21}\bar{k}_1 + a_{22}\bar{k}_2 + \dots + a_{2s}\bar{k}_s)] \\ \dots \\ \bar{k}_s = \bar{f}[x_n + c_s h; \bar{y}_n + h(a_{s1}\bar{k}_1 + a_{s2}\bar{k}_2 + \dots + a_{ss}\bar{k}_s)]. \end{cases} \quad (3)$$

Решение системы (3) может быть получено с использованием следующей итерационной схемы:

$$\begin{cases} \bar{k}_i^{(0)} = \bar{f}[x_n + c_i h; \bar{g}_i^{(0)}], \\ \bar{k}_i^{(l+1)} = \bar{f}[x_n + c_i h; \bar{g}_i^{(l)}], \end{cases}$$

$$\bar{g}_i^{(0)} = \bar{y}_n, \quad \bar{g}_i^{(l)} = \bar{y}_n + h \sum_{j=1}^s a_{ij} \bar{k}_j^{(l)}, \quad i = \overline{1, s}, \quad l = \overline{1, N}.$$

где

Здесь итерационный процесс, повторенный l -раз представляет $\bar{k}_i^{(l)}$, $i = \overline{1, s}$, как l -тую аппроксимацию для шагового коэффициента \bar{k}_i .

Решение уравнения

Для определения коэффициентов $k_i, i = \overline{1, s}$ необходимо решить систему из s , в общем случае нелинейных, уравнений:

$$\begin{cases} k_1 = f[x_n + c_1 h; y_n + h(a_{11}k_1 + a_{12}k_2 + \dots + a_{1s}k_s)] \\ k_2 = f[x_n + c_2 h; y_n + h(a_{21}k_1 + a_{22}k_2 + \dots + a_{2s}k_s)] \\ \dots \\ k_s = f[x_n + c_s h; y_n + h(a_{s1}k_1 + a_{s2}k_2 + \dots + a_{ss}k_s)]. \end{cases} \quad (4)$$

Для решения системы выбран метод простой итерации с итерационной схемой, описанной (3).

Вычисление 1-й итерации отображено на схеме:

$$\begin{cases} k_1^{(l)} = f[x_n + c_1 h; y_n + h(a_{11}k_1^{(l-1)} + a_{12}k_2^{(l-1)} + \dots + a_{1s}k_s^{(l-1)})] \\ k_2^{(l)} = f[x_n + c_2 h; y_n + h(a_{21}k_1^{(l-1)} + a_{22}k_2^{(l-1)} + \dots + a_{2s}k_s^{(l-1)})] \\ \dots \\ k_s^{(l)} = f[x_n + c_s h; y_n + h(a_{s1}k_1^{(l-1)} + a_{s2}k_2^{(l-1)} + \dots + a_{ss}k_s^{(l-1)})]. \end{cases} \quad (5)$$

Количество итераций l равняется порядку метода g .

Выполнение нулевой итерации, как и последующих, разбивается на количество процессоров. При этом на каждом из процессоров выполняется расчет s/p коэффициентов k . Где s – порядок метода, а p – количество процессоров.

В параллельном алгоритме, после каждой итерации (как нулевой, так и каждой из l последующих) выполняется множественная пересылка типа «всем» между процессорами. Каждый процессор передает коэффициенты, хранящиеся в переменной kr_{goc} всем процессорам, кроме себя. После решения системы (4), на каждом процессоре выполняется вычисление y_{n+1} по формуле:

$$y_{n+1} = y_n + h \cdot (b_1 \cdot k_1 + b_2 \cdot k_2 + \dots + b_s \cdot k_s).$$

Аналитические динамические характеристики метода:

$$T_1 = ((t_+ + t_* + T_F) + l * (t_+ (2 + (s-1)) + t_* (2 + s) + T_F)) * s + t_+ (1 + (s-1)) + t_* (1 + s)$$

$$T_{p,comp} = ((t_+ + t_* + T_F) + l * (t_+ (2 + (s-1)) + t_* (2 + s) + T_F)) * \left[\frac{s}{p} \right] + (t_+ (1 + (s-1)) + t_* (1 + s))$$

Данные формулы были подтверждены проведенным экспериментом на MPI-кластере. При проведении эксперимента используется топология однонаправленное кольцо.

Экспериментальные значения динамических характеристик последовательного и параллельного алгоритмов:

Порядок метода	Послед. алгоритм	2 процессора	
		Время	Ускорение
2	0,0000164	0,0000302	0,543046
4	0,0000329	0,0000516	0,637597
6	0,000052	0,0000698	0,744986
8	0,00037	0,0000812	4,55665
10	0,000193	0,000203	0,950739

Решение системы уравнений

Для определения коэффициентов $\bar{k}_i, i = \overline{1, s}$ необходимо решить систему из $S \times m$, в общем случае нелинейных, уравнений:

$$\begin{cases} \bar{k}_1 = \bar{f}[x_n + c_1 h; \bar{y}_n + h(a_{11}\bar{k}_1 + a_{12}\bar{k}_2 + \dots + a_{1s}\bar{k}_s)] \\ \bar{k}_2 = \bar{f}[x_n + c_2 h; \bar{y}_n + h(a_{21}\bar{k}_1 + a_{22}\bar{k}_2 + \dots + a_{2s}\bar{k}_s)] \\ \dots \\ \bar{k}_s = \bar{f}[x_n + c_s h; \bar{y}_n + h(a_{s1}\bar{k}_1 + a_{s2}\bar{k}_2 + \dots + a_{ss}\bar{k}_s)]. \end{cases}$$

Для решения системы выбран метод простой итерации с итерационной схемой, описанной (3).

На нулевой итерации выполняется вычисление функций уравнений с аргументами

$$x_n + c_i h; y_{n1}, y_{n2}, \dots, y_{nm}; i = \overline{1, s}.$$

$$\left\{ \begin{array}{l}
 k_{11}^{(0)} = f_1[x_n + c_1 h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 k_{12}^{(0)} = f_2[x_n + c_1 h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 \dots \\
 k_{1m}^{(0)} = f_m[x_n + c_1 h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 k_{21}^{(0)} = f_1[x_n + c_2 h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 k_{22}^{(0)} = f_2[x_n + c_2 h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 \dots \\
 k_{2m}^{(0)} = f_m[x_n + c_2 h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 \dots \\
 k_{s1}^{(0)} = f_1[x_n + c_s h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 k_{s2}^{(0)} = f_2[x_n + c_s h; y_{n1}, y_{n2}, \dots, y_{nm}] \\
 \dots \\
 k_{sm}^{(0)} = f_m[x_n + c_s h; y_{n1}, y_{n2}, \dots, y_{nm}]
 \end{array} \right.$$

На последующих l итерациях выполняется вычисление коэффициентов $\bar{k}_i^{(l)}, i = \overline{1, s}$, используя коэффициенты $\bar{k}_i^{(l-1)}, i = \overline{1, s}$ из предыдущей $(l-1)$ -й итерации.

Количество процессоров равномерно делится на порядок метода, таким образом составляя s групп. Каждая группа вычисляет m уравнений в каждом векторе \bar{k}_i . В случае, если процессоров больше, чем порядок метода, уравнения в каждой группе соответственно делятся между процессорами группы. Таким образом, каждый процессор в группе вычисляет d элементов \bar{k}_i , где $d = \lceil m/c \rceil$, а $c = \lceil p/s \rceil$. Вычисление \bar{y}_{n+1} разбивается на p процессоров. На каждом процессоре вычисляется m/p элементов вектора \bar{y}_{n+1} .

Аналитические динамические характеристики метода:

$$T_1 = ((t_+ + t_* + T_F) + l * (t_+ (1 + s * m)) + t_* (1 + (1 + s) * m) + T_F * m) * s + t_+ (1 + (s - 1)) + t_* (1 + s)$$

$$T_{p,comp} = ((t_+ + t_* + T_F) + l * (t_+ (1 + s * m)) + t_* (1 + (1 + s) * m) + T_F * m) * \left[\frac{m}{\left[\frac{p}{s} \right]} + t_+ (1 + (s - 1)) + t_* (1 + s) \right] * \left[\frac{m}{p} \right]$$

Данные формулы были подтверждены проведенным экспериментом на MPI-кластере. При проведении эксперимента используется топология однонаправленное кольцо.

Экспериментальные значения динамических характеристик последовательного и параллельного алгоритмов для методов второго порядка:

Кол-во уравнений	Послед. алгоритм	2 процессора		4 процессора	
		Время	Ускорение	Время	Ускорение
4	0,0000115	0,0000736	0,15625	0,000333	0,034535
40	0,000624	0,00211	0,295735	0,000706	0,883853
400	0,10277	0,10517	0,97718	0,04925	2,086701
4000	7,90148	3,37831	2,338885	4,52561	1,745948
8000	32,92852	14,69011	2,241543	19,62485	1,677899

Экспериментальные значения динамических характеристик последовательного и параллельного алгоритмов для методов четвертого порядка:

Кол-во уравнений	Послед. алгоритм	4 процессора		8 процессоров	
		Время	Ускорение	Время	Ускорение
4	0,0000334	0,000685	0,048759	0,0054	0,006185
40	0,00353	0,01237	0,285368	0,00436	0,809633
400	0,34222	0,588	0,582007	0,207955	1,645644
4000	25,8134	17,354	1,487461	15,496	1,665811
8000	108,9928	58,7362	1,855632	70,1942	1,552732

Анализирую полученные в результате эксперимента значения, можно сделать вывод о целесообразности применения параллельных алгоритмов при решении жестких задач Коши ПНМРК.

Выводы

Рассмотрены параллельные алгоритмы решения нелинейной задачи Коши. Выполнен анализ решения одного дифференциального уравнения и системы таких уравнений большой размерности.

Оговорены особенности выполнения параллельной реализации алгоритмов. Проанализированы аналитические оценки динамических характеристик.

Проведены эксперименты на MPI-кластере, подтверждающие аналитические оценки и позволяющие наглядно проиллюстрировать эффективность использование параллельных алгоритмов при решении жестких задач Коши.

Список литература

1. Хайрер Э., Нёрсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи: Пер. с англ. – М.: Мир, 1990. – 512с.

2. Хайрер Э., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи. – М.: Мир, 1999. – 685с.

3. Фельдман Л.П., Назарова И.А. Эффективность параллельных алгоритмов оценки локальной апостериорной погрешности для численного решения задачи Коши // Электронное моделирование, т. 29, № 3, 2007. – С. 11-25.