

МЕТОДИКА ОЦЕНКИ ПОТЕНЦИАЛЬНОЙ РАСПАРАЛЛЕЛИВАЕМОСТИ ЦИКЛИЧЕСКИХ ЗАДАЧ

И.Г. Кириллов, Е.Г. Толстолужская
(Харьковский университет Воздушных Сил им. И. Кожедуба)

Предлагается методика оценки потенциальных возможностей распараллеливания циклических задач. Проводится сравнительный анализ результатов ее практического использования на примерах однотипных функциональных задач на основе циклических итерационных процедур.

потенциальная распараллеливаемость, циклические задачи

Постановка проблемы. При разработке параллельных аппаратно-программных средств АСУ различного назначения, работающих в режиме реального времени, возникает необходимость обоснования выбора численных методов и реализующих их алгоритмов решения требуемых прикладных задач. Одним из основных показателей, характеризующих практическую пригодность этих методов, является время их реализации при обеспечении требуемой точности.

В условиях разнообразия существующих методов решения функциональных задач, которые, как правило, являются циклическими, особую актуальность приобретает повышение качества оценивания вычислительной сложности и потенциальных возможностей распараллеливания соответствующих алгоритмов с целью их обоснованного выбора по критерию минимума (или не превышения допустимого) времени выполнения [1, 2].

Анализ литературы. Вопросы построения временных моделей работы алгоритмов, конкретных устройств на этапе их проектирования рассмотрены в целом ряде работ [1 – 9]. Как показывает анализ, известные подходы реализации «параллельных» алгоритмов базируются на разномножении «тел» циклов без учета возможностей по распараллеливанию операций внутри этих тел. Отсутствие такого учета не позволяет получить оценки потенциальной «распараллеливаемости» циклических алгоритмов и, как следствие, предельного (минимального) времени их выполнения, что затрудняет решение задачи обоснованного выбора численного метода решения той или иной функциональной задачи, реализуемой в АСУ реального времени.

Цель статьи. В статье предлагается методика оценки потенциальных возможностей параллельной реализации циклических задач на основе размножения тел циклов при их максимальном распараллеливании. Приводится описание основных этапов ее работы с указанием особенностей их выполнения, обсуждаются результаты практического применения разработанной методики на примере решения системы линейных уравнений различными численными методами. Такие задачи, относящиеся к циклическим итерационным задачам, нашли широкое практическое применение в системах обработки сигналов на фоне помех, спектрального анализа процессов различной природы, обработки и сжатия изображений и т.д. Они эквивалентны произведению матрицы, обратной заданной (в частности корреляционной матрицы случайного процесса), на столбец свободных членов. Приводятся результаты сравнительного анализа ряда численных методов решения систем уравнений по времени их реализации в зависимости от числа процессоров в составе многопроцессорного вычислительного комплекса средств автоматизации АСУ. Иллюстрируются возможности сокращения этого времени за счет обоснованного выбора метода с меньшей вычислительной сложностью на основе учета специфики решения задачи.

Изложение основного материала. Методика оценки потенциальной распараллеливаемости циклических задач представлена на рис.1 и включает следующие основные этапы.

На *первом этапе* (блок 1) подготавливаются исходные данные, в качестве которых выступают:

– формализованное представление задачи, например, ее программа на языке СИ (СИ-программа);

– диапазон $n \in n_{\min}, n_{\max}$ изменения ее размерности (как правило, задает «внешний» цикл), например, размерности задачи решения системы уравнений;

– длительность операций, например, в процессорных тактах.

Этап 2 (блок 3) предназначен для синтеза «ациклического аналога» исходной циклической программы путем раскрытия тел циклов. Осуществляется подготовка и формализация данных, необходимых для синтеза параллельной модели выполнения алгоритма для заданного значения n . (в блоке 2 задается начальное значение, например, n_{\max} – значение, обеспечивающее требуемую точность решения функциональной задачи). Далее формируются структуры данных, описывающие набор выполняемых в задаче элементарных операций, их типы, взаимосвязи (входы и выходы), и однозначно задающие граф исходной задачи, вершинами которого являются операции алгоритма.

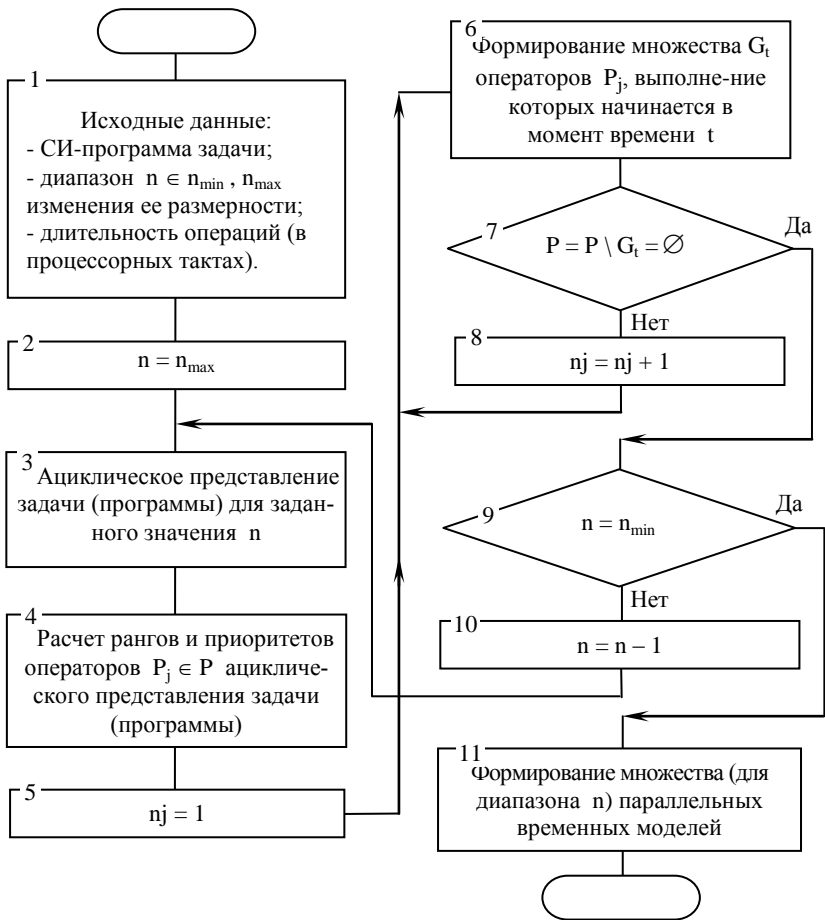


Рис. 1. Структурная схема методики оценки потенциальной распараллеливаемости циклических задач

Дальнейшие этапы методики, в отличие от существующих подходов [3, 5, 7], направлены на максимальное распараллеливание тел циклов.

На *третьем этапе* (блок 4) решаются две основные задачи.

Во-первых, для каждой вершины $P_j \in P$, где P – множество всех вершин графа задачи, осуществляется расчет значения максимального времени выполнения j -й операции с учетом всех зависящих от него операций алгоритма, задаваемого рангом r_j по соотношениям [1, 2]:

$$r_j = t_j^0 \text{ при } W(P_j) = \emptyset; \quad (1)$$

$$r_j = \max_{P_\xi \in W(P_j)} (r_\xi + t_j^0) \text{ при } W(P_j) \neq \emptyset, \quad (2)$$

где t_j^0 – время выполнения j -й операции; $W(P_j)$ – внешнее множество операторов P_j , условие $W(P_j) = \emptyset$ означает, что оператор P_j является выходным оператором задачи. В случае непустого внешнего множества – $P_\xi \in W(P_j)$, оператор P_ξ является внешним для оператора P_j ; r_ξ – ранг (временная характеристика) оператора P_ξ .

Во-вторых, для оператора $P_j \in P$ производится расчет значения приоритета b_j , определяющего его важность по отношению к другим операторам P_k , и тем самым очередность его рассмотрения при решении задачи распараллеливания. Соотношения, обеспечивающие расчет приоритетов, имеют следующий вид:

$$b_j = 1 \text{ при } r_j = \max_{P_j \in P} r_j \text{ и } r_i < r_j \text{ при } i \neq j; \quad (3)$$

$$b_j < b_i \text{ при } r_i = r_j \text{ и } t_j^0 > t_i^0; \quad (4)$$

$$b_j < b_i \text{ при } r_j > r_i; \quad (5)$$

$$b_j < b_i \text{ при } r_i = r_j; t_j^0 = t_i^0 \text{ и } |W_j| > |W_i|; \quad (6)$$

$$b_j < b_i \text{ при } r_i = r_j; t_j^0 = t_i^0 \text{ и } |W_j| = |W_i|, \text{ но } j < i. \quad (7)$$

Здесь r_i , r_j – ранги операторов P_i , P_j , определяемые по соотношениям (1), (2).

Этап 4 (блок 6). Формируется множество $G_t(nj)$ операторов P_j графа задачи, являющихся кандидатами на начало выполнения на временном ярусе с номером nj в текущий момент времени t в предположении отсутствия ограничений на максимальное количество функциональных модулей различных типов (или универсальных процессоров, реализующих элементарные операции) применительно к случаю использования в модели метода совмещения операций. Начальное значение количества функциональных модулей (процессоров) принято равным 1 (последовательный вариант выполнения алгоритма).

$$G_t(nj) = R_{nj-1} \cup K_{nj}, \quad (8)$$

где R_{nj-1} – множество операторов P_j , реализация которых не была начата на предыдущем $(nj - 1)$ -м ярусе в связи с отсутствием необходимых свободных модулей (или процессоров); K_{nj} – множество операторов,

выполнение которых может быть начато на n_j -м ярусе в связи с наличием всех необходимых для них данных, обусловливаемых учетом ранее выполненных операторов и информационно-управляющих связей между операторами $P_j \in P$:

$$K_{n_j} = \begin{cases} \bigcup_j P_j \text{ для } S(P_j) = \emptyset \text{ (при } n_j = 0); \\ \bigcup_j P_j \text{ при выполнении для } P_j \text{ следующих условий:} \\ \quad P_j \in \bigcup_{P_\xi \in G_{n_j-1}} W(P_\xi) \text{ при } S(P_j) \subseteq G_{n_j-1}; \\ \quad t_\xi^H + t_\xi^0 \leq t(n_j) \text{ для всех } P_\xi \in S(P_j), \end{cases} \quad (9)$$

где $S(P_j)$ – сопряженное множество оператора P_j , условие $S(P_j) = \emptyset$ означает, что оператор P_j является входным оператором задачи; $G_t(n_j-1)$ – множество операторов P_j , реализация которых была начата на предыдущих ярусах $n_j \in 1, n_j-1$.

Этап 5 (блок 11). После выполнения условий (блоки 7, 9) для множества синтезированных параллельных временных моделей исходной задачи формируется множество рассчитанных временных характеристик $T(P(n))$. В качестве временного показателя потенциальной распараллеливаемости циклической задачи используется среднее время реализации множества P операторов реализующего ее алгоритма, определяемое выражением

$$T(P) = \sum_{\xi=1}^W p_\xi T_\xi, \quad (10)$$

где W – число ветвей в алгоритме; p_ξ – вероятность реализации ξ -й ветви; T_ξ – время реализации ξ -й ветви параллельного алгоритма, определяемое по формуле

$$T_\xi = \max_{P_j \in P(\xi)} (t_j^H + t_j^0), \quad (11)$$

где $P(\xi)$ – множество операторов ξ -й ветви; t_j^H – момент начала выполнения оператора $P_j \in P(\xi)$; t_j^0 – время выполнения оператора $P_j \in P(\xi)$.

Таким образом, результатом выполнения методики рис. 1 являются зависимости времени $T(P(n))$ от числа NM процессоров для различных значений n размерности анализируемой циклической задачи из выбранного диапазона $n \in n_{\min}, n_{\max}$.

Рассмотрим результаты практического применения разработанной методики на примере решения системы линейных уравнений различными численными методами: методом Гаусса без выбора ведущего элемента [6] (для случая матрицы системы общего вида, специфика отсутствует) и методом обобщенной факторизации Левинсона (для случаев эрмитовой и эрмитовой теплоцевой матрицы системы) [10, 11].

Время выполнения элементарных операций в процессорных тактах задано для микропроцессора MIPS R10000 (табл.1) [12].

Таблица 1

Время выполнения элементарных операций
в микропроцессоре MIPS R10000

Процессорная операция	Время выполнения (такты)
Целочисленное сложение, вычитание, логические операции и переходы	1
Целочисленные команды загрузки/записи	2
Целочисленное умножение	10
Целочисленное деление	35
Вычисление квадратного корня	10

Результаты применения методики оценки потенциальной распараллеливаемости рассматриваемой задачи различными численными методами представлены на рис. 1 – 3. Параметром семейств кривых, приведенных на рисунках, является размерность задачи, т.е. размерность n исходной квадратной матрицы системы.

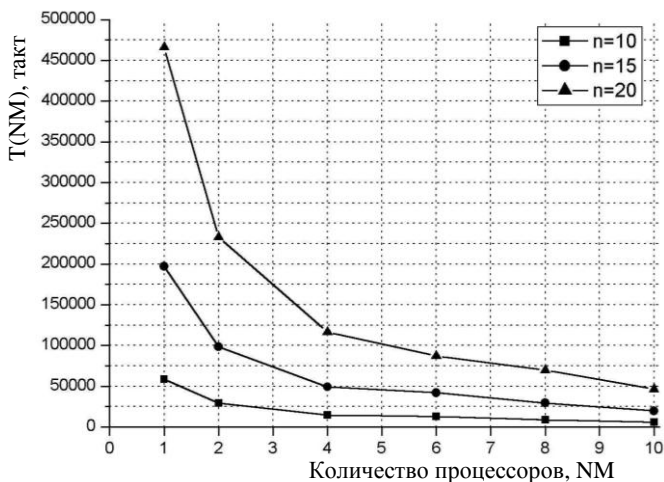


Рис. 1. Время решения системы линейных уравнений методом Гаусса

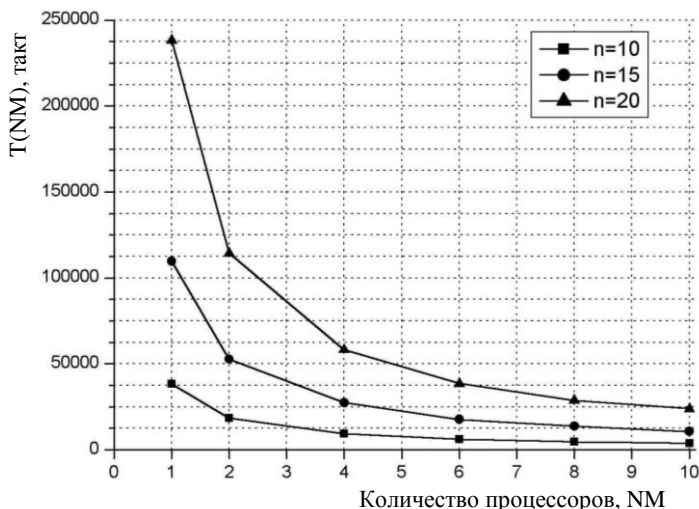


Рис. 2. Время решения системы линейных уравнений методом обобщенной факторизации Левинсона (эрмитова матрица)

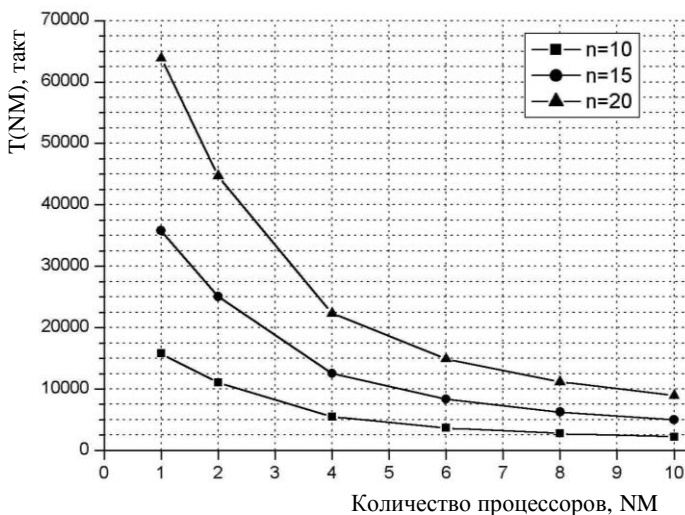


Рис. 3. Зависимость времени решения системы линейных уравнений методом обобщенной факторизации Левинсона (эрмитова тривицева матрица)

Сравнительная характеристика зависимости времени решения системы линейных уравнений от количества процессоров различными численными методами для размерности матрицы $n = 20 \times 20$ представлена на рис. 4.

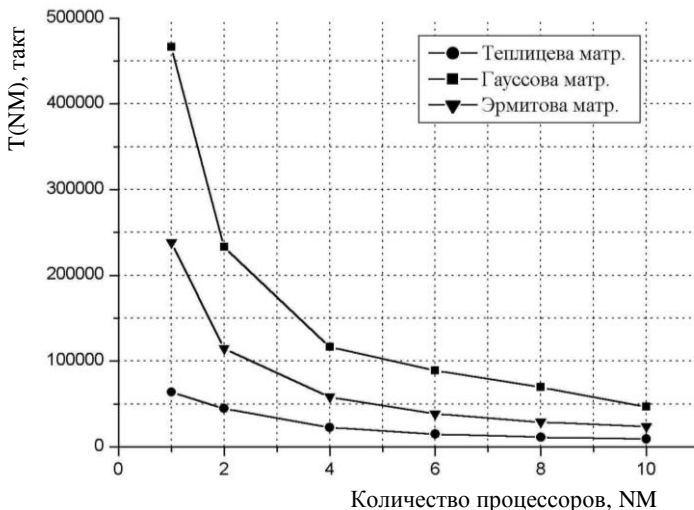


Рис. 4. Зависимость времени решения системы линейных уравнений различными методами для размерности матрицы $n = 20 \times 20$

Полученные результаты применения предложенной методики оценки потенциальной распараллеливаемости циклической задачи в условиях рассмотренных примеров (рис. 1 – 3) свидетельствуют о возможности сделать обоснованный выбор размерности n в условиях временных и аппаратных ограничений. Так решение системы линейных уравнений с матрицей общего вида (рис. 1) за заданное (в условных единицах) время, например, $T \approx 25000$ тактов для $n = 20$ оказывается невозможным. В условиях наличия достаточного количества оборудования (число процессоров равно $NM = 8$) возможно ее решение для $n = 15$, а в случае отсутствия – только для $n = 15$ ($NM = 2$). Заметим, что снижение размерности задачи в общем случае может приводить, например, к снижению точности получения результата, что в ряде ситуаций не допустимо.

В этом случае требуется провести сравнительный анализ различных численных методов решения требуемой задачи. Так в условиях рассмотренного примера для эрмитовой тривиальной матрицы системы целесообразно использовать соответствующий численный метод (рис. 3, 4), который позволяет решить задачу за требуемое время ($T \approx 25000$ тактов) с необходимой точностью ($n = 20$) при использовании $NM = 4$ процессоров.

Выводы. 1. Предложенная в статье методика позволяет получить оценки потенциально возможного времени параллельного выполнения циклических алгоритмов на основе разложения тел циклов и максимального их распараллеливания.

2. Методика оценки потенциальных возможностей распараллеливания циклических задач может быть использована при проектировании параллельных аппаратных и программных средств с минимальным временем параллельной реализации алгоритмов.

3. Результаты сравнительного анализа исследуемых численных методов решения системы линейных уравнений свидетельствуют о целесообразности применения метода обобщенной факторизации Левинсона для случаев эрмитовой и эрмитовой теплоцевой матрицы системы, обеспечивающего минимальное время решения.

ЛИТЕРАТУРА

1. Поляков Г.А., Умрихин Ю.Д. Автоматизация проектирования сложных цифровых систем коммутации и управления. – М.: Радио и связь, 1988. – 304 с.
2. Толстолужская Е.Г., Толстолужский Д.А., Поляков Г.А. Оценка диапазона изменения эффективности параллельных временных моделей алгоритмов // *Радиоелектронні і комп'ютерні системи*. – 2005. – №3 (11). – С. 106-114.
3. Трахтенгерц Э.А. Введение в теорию анализа и распараллеливания программ ЭВМ в процессе трансляции. – М.: Наука, 1981. – 256 с.
4. Системы параллельной обработки / Под. ред. Д. Ивенса; пер. с англ. – М.: Мир, 1985. – 416 с.
5. Барский А.Б. Параллельные процессы в вычислительных системах. Планирование и организация. – М.: Радио и связь, 1990. – 256 с.
6. Корнеев В.В. Параллельные вычислительные системы. – М.: Нолидж, 1999. – 320 с.
7. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – СПб.: БХВ – Петербург, 2002. – 608 с.
8. Поляков Г.А. Адаптивные самоорганизующиеся системы с мультипараллельной обработкой данных – стратегия развития цифровой вычислительной техники в XXI веке // *Прикладная радиоэлектроника*. – Х.: АНПРЭ, ХНУРЭ, 2002. – Т. 1, № 1. – С. 57-69.
9. Толстолужская Е.Г. Методика формализованного синтеза мультипараллельных архитектурно-ориентированных моделей решения задач // *Зб. наук. пр. інституту проблем моделювання в енергетиці ім. Г.С. Пухова*. – К.: ІПМЕ, 2003. – Вип. 22. – С. 206-215.
10. Леховицкий Д.И. Обобщенный алгоритм Левинсона и универсальные решетчатые фильтры // *Изв. вузов. Радиофизика*. – М., 1992. – Т. 35, № 9-10. – С. 790-808.
11. Леховицкий Д.И., Кириллов И.Г. Систематические структуры спецпроцессоров на основе решетчатых фильтров // *Военно-технические труды*. – Х.: ВИРТА ПВО, 1993. – Вып. 1, Ч. 2. – С. 175-182.
12. Аваков В. Микропроцессор MIPS R10000 // *Открытые системы*. – М., 1995. – № 6(14). – С. 62-69.

Поступила 19.04.2006

Рецензент: доктор технических наук, профессор Г.А. Поляков,
Академия наук прикладной радиоэлектроники, Харьков.