

УДК 004.75

Д.О. Бобынцев, Юго-Западный государственный университет (Курск)
(e-mail daniel8728@yandex.ru)

Э.И. Ватутин, канд. техн. наук, доцент, Юго-Западный государственный университет (Курск) (e-mail evatutin@rambler.ru)

В.С. Титов, д-р техн. наук, профессор, Юго-Западный государственный университет (Курск) (e-mail titov-kstu@rambler.ru)

ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ МАТРИЧНОГО МУЛЬТИПРОЦЕССОРА ПРИ ВЫПОЛНЕНИИ ПАРАЛЛЕЛЬНОГО АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ ГРАВИТАЦИОННОГО ВЗАИМОДЕЙСТВИЯ N ТЕЛ

Рассмотрено разбиение задачи гравитационного взаимодействия N тел на параллельные фрагменты, произведена оценка пиковой производительности параллельной вычислительной системы при решении задачи в матричном мультипроцессоре в конфигурации 8×8, исследованы результаты планирования размещения параллельных фрагментов алгоритма с применением двух критериев минимизации коммуникационной задержки, оценена получаемая реальная производительность вычислительной системы для каждого из них.

Ключевые слова: размещение параллельных подпрограмм, матричные мультипроцессоры, задача N тел, коммуникационная задержка в мультипроцессорах, минимаксный критерий, минимаксиминный критерий.

Создание высокопроизводительных вычислительных систем (ВС) является одним из наиболее важных направлений развития вычислительной техники в свете стремления к получению большей производительности в соответствии с законом Мура. Данные системы находят применение при решении различных вычислительных задач, которые характеризуются либо большим объемом вычислений, либо имеют ограничения на время получения результата. Известные частные случаи таких систем – матричные мультипроцессоры (ММП) и суперкомпьютеры с развитой коммуникационной подсистемой, топология межсоединений в которых представляет собой матрицу, тор либо их обобщения на случаи большей размерности (например, подсистема суперкомпьютеров IBM BlueGene представляет собой 5-мерный тор). При интегральном исполнении МПП требование планарности связей и ограничение на число слоев металлизации на кристалле не позволяют реализовать большое число связей, поэтому обычно для них применяется матричная или реже тороидальная топология межпроцессорных коммуникаций. Одной из важных задач в МПП является планирование размещения подпрограмм по множеству обрабатывающих процессоров, целью которого является минимизация величин коммуникационных задержек при передаче данных между процессорами, что особенно важно при решении задач, требующих частых обменов данными между отдельными подзадачами (т.н. сильносвязные задачи). Существует целый ряд задач, в которых интегральное время выполнения целиком и полностью лимитируется временем обмена данными (например, моделирование климата, решение систем дифференциальных уравнений с

использованием сеточных методов и др.) в соответствии с законом Амдала, что делает актуальной задачу отыскания соответствия между подзадачами и процессорами, которое характеризуется минимальным временем обмена данными и позволяет достичь лучшего масштабирования ВС с ростом числа процессоров. Длинные составные и перекрывающиеся маршруты транзитной передачи данных приводят к возрастанию коммуникационных задержек, что существенно снижает реальную производительность ВС (до 50% от пиковой и более) и приводит к простоям процессоров, ожидающих окончания обмена данными. Для выхода из создавшегося положения известны и с успехом развиваются различные подходы для эвристического планирования размещения подпрограмм, сводящие задачу достижения максимальной производительности ВС к задаче минимизации величины коммуникационной задержки, при этом различные подходы характеризуются как применением различных эвристик, так и различной степенью близости достигаемой реальной производительности по отношению к пиковой. В данной статье показано, что применение различных подходов к оценке коммуникационных задержек в сильно-связных задачах на примере задачи гравитационного взаимодействия N тел характеризуется достижением различной реальной производительности ВС.

Для оценки величины коммуникационной задержки при планировании размещения подпрограмм в работе [1] предложен минимаксиминный показатель, заключающийся в вычислении для каждой пары процессоров минимальной суммарной задержки, что соответствует выбору наименее загруженного пути между двумя процессорами при последующей маршрутизации. После определения минимальных суммарных задержек для всех пар процессоров из них выбирается максимальная, которая лимитирует время обменов данными между процессорами и подлежит минимизации. Так как данный метод вычисления величины задержки оценивает худший случай возможных перекрытий каналов передачи данных, то после маршрутизации задержка не увеличивается больше значения, полученного данным методом. Известным конкурирующим подходом [2] является минимаксный показатель коммуникационной задержки, который характеризует время обмена данными между выбранной парой процессоров без учета перекрытий (наложений) каналов передачи данных, возникающих при одновременном обмене данными для группы пар процессоров. Наложение маршрутов передачи данных увеличивает время межпроцессорного обмена и снижает реальную производительность мультипроцессора при решении сильно-связных задач, что не учитывается при минимизации минимаксного показателя.

В качестве объекта для исследования выбрана сильно-связная задача гравитационного взаимодействия N тел, так как минимизация коммуникационной задержки наиболее актуальна для данного класса задач. Данная задача является актуальной и активно исследуется для проверки различных космогонических гипотез (например, образования протопланетных дисков по Канту-Лапласу), моделирования динамики планетных систем и звезд в шаровых

скоплениях, моделирования распределения и динамики объектов в поясе астероидов, кольцах Сатурна, поясе Койпера и облаке Оорта, моделирования распределения темной материи вокруг Млечного пути и в масштабах Вселенной и др.

В классической постановке (Ньютон) данная задача формулируется следующим образом. В трехмерном пространстве находится N материальных точек, для которых известны массы m_i , начальные на момент времени $t=0$ положения $r_{i|t=0}=r_{i0}$ и скорости $v_{i|t=0}=v_{i0}$. Попарное взаимодействие материальных точек подчинено закону всемирного тяготения, а сила гравитации, действующая на каждую из материальных точек, представляет собой векторную сумму сил со стороны остальных тел. Необходимо найти положения точек для всех последующих моментов времени. Задача в данной постановке является наиболее простой в своем классе и не учитывает ряд эффектов теории относительности (Эйнштейн), при этом требуется найти ее точное решение (без применения эвристик, введения иррегулярных сил и т.п.). Аналитическое решение задачи затруднительно при $N > 2$, поэтому на практике применяется численное моделирование. Рассмотрим процесс ее решения с использованием постоянного шага по времени Δt без его адаптивной корректировки, при этом время одной итерации моделирования характеризуется асимптотикой $O(N^2)$, что является серьезным ограничением при моделировании динамики систем из сотен тысяч тел и более.

Численное решение данной задачи выполняется посредством расчёта приращений для скоростей и координат с использованием шага по времени Δt :

$$\begin{aligned}\vec{v}^{(t)} &= \vec{v}^{(t-1)} + \vec{a}^{(t)} \Delta t, \\ \vec{x}^{(t)} &= \vec{x}^{(t-1)} + \vec{v}^{(t)} \Delta t,\end{aligned}$$

где $\vec{v}^{(t)}$ – скорость тела в момент времени t , $\vec{v}^{(t-1)}$ – скорость тела в предшествующий момент времени $t-1$, $\vec{x}^{(t)}$ – координата тела в момент времени t , $\vec{a}^{(t)}$ – ускорение тела в момент времени t , рассчитываемое по второму закону Ньютона:

$$\vec{a}^{(t)} = \frac{\vec{F}}{m},$$

где \vec{F} – равнодействующая сил, действующих на тело, рассчитываемая как векторная сумма сил притяжения между данным телом и остальными телами. При этом сила притяжения между i -м и j -м телом рассчитывается по закону всемирного тяготения:

$$\vec{F}_{ij} = G \frac{m_i m_j}{r^3} \vec{r},$$

где G – гравитационная постоянная, r – расстояние между i -м и j -м телом, рассчитываемое по формуле:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2},$$

где $\vec{x}_i = (x_i, y_i, z_i)$ – вектор координат i -го тела в 3-мерном пространстве.

На основании описанного выше алгоритм расчёта очередного вектора координат i -го тела с указанием требуемого числа операций с плавающей точкой (FLOP) состоит из следующих действий:

- 1) расчёт расстояния между парой тел (выполняется параллельно для текущего i -го тела и всех остальных тел) – 9 FLOP;
- 2) расчёт сил притяжения между данным телом и остальными телами (аналогично п. 1 выполняется параллельно) – 4 FLOP;
- 3) расчёт равнодействующей сил, действующих на тело (совмещено по времени с п. 2);
- 4) расчёт ускорения тела – 1 FLOP;
- 5) расчёт скорости тела в момент времени t – 2 FLOP;
- 6) расчёт координат тела в момент времени t – 6 FLOP.

Схема последовательно-параллельного решения задачи с разбиением на подзадачи приведена на рис. 1, при этом в составе каждой подзадачи производится вычисление физических характеристик одного тела, а общее число подзадач составляет N .

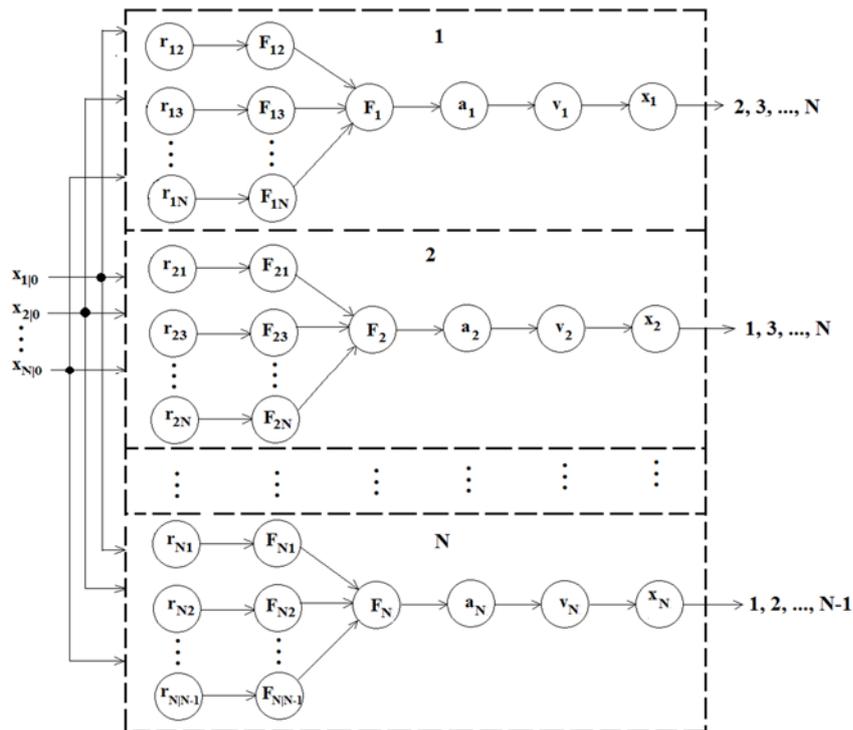


Рис. 1. Схема параллельного решения задачи N тел

Представляется возможным параллельное решение данной задачи в матричном мультипроцессоре, при котором каждый процессор выполняет цикл расчёта координат одного тела и передаёт результат всем остальным процессорам, выполняющим расчёты для других тел, после чего производится переход к следующей итерации. В соответствии с изложенным выше граф взаимодействия подзадач для рассмотренного параллельного алгоритма решения задачи является полностью связным, а весами дуг являются объёмы дан-

ных, необходимые для следующей итерации. С учетом специфики разбиения на подзадачи между итерациями моделирования требуется обмен векторами координат ($3 \times 8 = 24$ байт для вещественных чисел двойной точности), в то время как вектора скоростей тел хранятся в памяти процессоров и не передаются между процессорами, снижая нагрузку на коммуникационную подсистему.

На примере описанной выше задачи произведена оценка реальной производительности матричного мультипроцессора Tile64Pro [3] в конфигурации 8×8 процессоров с применением средств планирования размещения, позволяющих оптимизировать задержку в линиях связи между процессорами. Для этого будем производить оценку реальной производительности мультипроцессора как

$$P = \frac{V_{\text{выч}}}{T_{\text{выч}} + T_{\text{ком}}}$$

где $V_{\text{выч}}$ – объём вычислений (необходимое число операций с плавающей точкой, FLOP), $T_{\text{выч}}$ – время вычислений в секундах, $T_{\text{ком}}$ – время передачи данных, которое оценивается как максимальное время передачи данных в любой паре процессоров и зависит от результата планирования размещения подпрограмм.

В соответствии с алгоритмом расчёта координат тел объём вычислений определяется как

$$\begin{aligned} V_{\text{выч}} &= 9(N-1)N + 4(N-1)N + (N-1)N + N + 2N + 6N = \\ &= 14N^2 - 5N. \end{aligned} \quad (1)$$

Так как мультипроцессор является однородной вычислительной системой, в которой каждое процессорное ядро выполняет цикл расчёта координат для одного тела, интегральное время моделирования определяется произведением времени выполнения одного цикла расчёта координат на число итераций моделирования. В соответствии с характеристиками процессора Tile64Pro максимальная частота ядер составляет 1 ГГц, поэтому время выполнения цикла расчёта координат в наносекундах равно количеству тактов на выполнение операций с плавающей точкой, составляющих цикл:

$$T_{\text{выч}} = \frac{(N-1)c_1 + (N-1)(c_2 + c_3) + c_4 + c_5 + 3c_6}{10^9} \quad (2)$$

где $c_1=54$ такта – вычисление расстояния между телами, $c_2+c_3=116$ тактов – вычисление силы притяжения между парой тел и модификация равнодействующей, $c_4=80$ тактов – вычисление ускорения тела, $c_5=14$ тактов – вычисление скорости тела, $c_6=14$ тактов – вычисление координаты тела.

Время передачи данных между процессорами при условии, что все обмены данными можно выполнять одновременно, лимитируется худшей попарной задержкой между процессорами, вычисляемой в результате выполнения алгоритма планирования размещения. При последовательном побайтном обмене время передачи данных между парой процессоров без учёта пе-

рекрытий маршрутов передачи данных определяется как произведение $m \cdot d \cdot c$, где m – количество передаваемых байт, d – количество последовательно соединённых линков (англ. hop) между данной парой процессоров, c – постоянный множитель, равный времени передачи одного байта по одному линку. Возможные перекрытия маршрутов, требующие суммирования времён передачи данных, учитываются в минимаксиминном показателе как описано выше. При этом время $T_{\text{ком}}$ определяется как максимальное время передачи данных в любой паре процессоров, вычисленное по минимаксиминному показателю с использованием оценочного значения $c=10$ нс, учитывающего возможности коммуникационной подсистемы мультипроцессора Tile64Pro.

Так как при полной загрузке мультипроцессора в выбранной конфигурации 8×8 (задача 64 тел) и одинаковых объёмах передаваемых данных оптимизация размещения подпрограмм с целью уменьшения задержки в линиях связи между процессорами не представляется возможной ввиду равноправности различных вариантов размещения, в качестве тестового примера выбрано заполнение $\frac{1}{4}$ мультипроцессора, что соответствует решению задачи 16 тел.

Представляется возможным теоретически оценить пиковую производительность мультипроцессора при решении поставленной задачи, если принять $T_{\text{ком}} \rightarrow 0$ (коммуникационная подсистема работает мгновенно и не оказывает влияния на интегральное время обработки данных, лимитируемое только временем вычислений). В данном случае с использованием формул (1) и (2) получаем, что $V_{\text{выч}}=3504$ FLOP, а $T_{\text{выч}}=2686$ нс. Тогда пиковая производительность мультипроцессора в указанной задаче равна $P_{\text{peak}} = \frac{V_{\text{выч}}}{T_{\text{выч}}} = 1,3$

GFLOP/s.

Алгоритм планирования размещения подпрограмм выполнен с применением минимаксиминного [1] и минимаксного показателей [2] с целью сравнения эффективности их применения. На рис. 2а показано начальное размещение 16 подпрограмм расчёта координат тел, при котором максимальное расстояние между процессорами, обрабатывающими данные подпрограммы, совпадает с Манхэттенской нормой и составляет 6 линков (выделены жирным). Данное размещение является оптимальным с точки зрения максимального расстояния между процессорами, однако приводит к резкому возрастанию коммуникационной задержки при маршрутизации по кратчайшим путям, вызванному полностью графа обмена данными между подпрограммами и значительными перекрытиями каналов передачи данных. При этом все кратчайшие пути между занятыми процессорами лежат в пределах выделенного на рис. 2а блока.

С целью улучшения исходного решения было проведено моделирование работы алгоритмов с использованием указанных выше минимаксного и минимаксиминного показателей оценки коммуникационной задержки и были получены следующие результаты. Максимальное время обмена данными в

любой паре процессоров для начального размещения на рис. 2а по минимаксному показателю и в соответствии с описанным выше способом вычисления времени передачи данных между двумя процессорами без учета перекрытия каналов составляет 1920 нс, тогда как в действительности имеют место множественные перекрытия, и худшее время обмена составляет 13440 нс. Данное время соответствует обмену между парой процессоров 1-28.

Между двумя процессорами, расположенными на одной горизонтали или одной вертикали процессорной матрицы, существует только один кратчайший путь, что позволяет оценить минимальное количество перекрытий на любом кратчайшем пути при описанном выше условии маршрутизации.

Рассмотрим вариант кратчайшего пути между 1 и 28 процессорами: 1-2-3-4-12-20-28. В приведенном выше варианте кратчайшего пути можно выделить следующие кратчайшие каналы меньшей длины, имеющие только один вариант кратчайшего пути (таблица 1).

Таблица 1. Перечень кратчайших каналов, имеющих только один вариант кратчайшего пути

Пара процессоров одной горизонтали/ вертикали	Длина канала	Пары процессоров на разных горизонталях и вертикалях	Длина канала
1-2	1	1-12	4
1-3	2	1-20	5
1-4	3	2-12	3
2-3	1	2-20	4
2-4	2	2-28	5
3-4	1	3-12	2
4-12	1	3-20	3
4-20	2	3-28	4
4-28	3		
12-20	1		
12-28	2		
20-28	1		

В соответствии с таблицей 1 рассчитаем минимально возможную суммарную задержку данного кратчайшего пути с учётом однонаправленного обмена данными между 1 и 28 процессорами, кратчайшее расстояние между которыми равно 6:

$$T_{\text{SUM}}=(24 \cdot 1 \cdot 6+24 \cdot 2 \cdot 4+24 \cdot 3 \cdot 2+24 \cdot 6 \cdot 1) \cdot 10=24 \cdot 10 \cdot (6+8+6+6)=6240 \text{ нс.}$$

На рис. 2а выделен описанный кратчайший путь между процессорами 1 и 28 с указанием количества накладывающихся на каждый межпроцессорный канал обменов данными.

Для получения худшего случая коммуникационной задержки допустим, что приведенные в таблице 1 однонаправленные обмены данными между процессорами, расположенными на разных горизонталях и вертика-

лях, будут проложены по указанному выше кратчайшему пути, и прибавим суммарную задержку для них к минимально возможной

$$T_{\text{SUM}}=6240+(24\cdot 2\cdot 1+24\cdot 3\cdot 2+24\cdot 4\cdot 3+24\cdot 5\cdot 2)\cdot 10=6240+240\cdot(2+6+12+10)=13440 \text{ нс.}$$

На рис. 2б показан вариант субоптимального размещения, полученный с применением минимаксного показателя для оценки качества размещения, на рис. 2в – с применением минимаксиминного показателя. Так как ни один из полученных вариантов не гарантирует наличия в любой паре взаимодействующих процессоров хотя бы одного кратчайшего пути, освобождённого от перекрытий, целесообразно оценивать коммуникационную задержку в полученных субоптимальных вариантах размещения по минимаксиминному показателю.

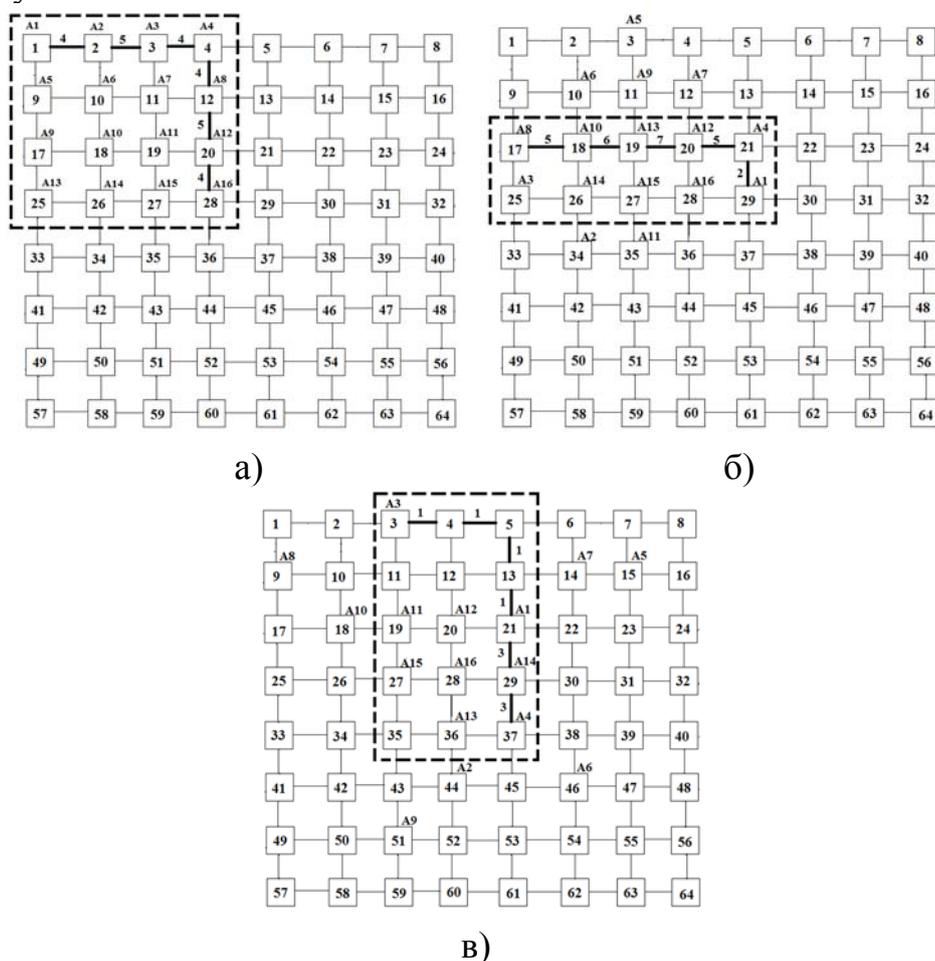


Рис. 2. Начальное размещение задачи 16 тел в матричном мультипроцессоре конфигурации 8×8 (а), вариант субоптимального размещения с использованием минимаксного (б) и минимаксиминного (в) показателей

В соответствии с описанным выше получено максимальное время передачи данных для рис. 2б – 8400 нс для процессоров 17 – 29 и для рис. 2в – 4560 нс для процессоров 3 – 37.

Аналогично первоначальному размещению оценим минимально возможную суммарную задержку для кратчайших путей описанных выше пар процессоров. Кратчайшие пути между процессорами 17 и 29 лежат в преде-

лах выделенного на рис. 2б блока. Из рис. 2б следует, что в полученном с применением минимаксного критерия варианте размещения ни один процессор из выделенного блока не остался свободным, поэтому при полностью обмен данными между подпрограммами любой кратчайший путь между процессорами 17 и 29 может иметь максимальную степень перекрытий. Рассмотрим вариант кратчайшего пути 17-18-19-20-21-29 длиной 5. В нём можно выделить пары процессоров, лежащие на одной горизонтали или одной вертикали: 17-18, 17-19, 17-20, 17-21, 18-19, 18-20, 18-21, 19-20, 19-21, 20-21, 21-29. Рассчитаем минимально возможную суммарную задержку при таком списке фактических перекрытий в соответствии с длинами вышеуказанных каналов передачи данных

$$T_{\text{SUM}}=(24 \cdot 1 \cdot 5+24 \cdot 2 \cdot 3+24 \cdot 3 \cdot 2+24 \cdot 4 \cdot 1+24 \cdot 5 \cdot 1) \cdot 10=6240 \text{ нс.}$$

Аналогично рис. 2а на рис. 2б показано минимальное количество перекрывающихся обменов данными на межпроцессорных каналах выделенного пути.

Полученное минимально возможное значение суммарной задержки совпадает с аналогичным значением для начального размещения, поэтому размещение, полученное с применением минимаксного критерия, гарантирует уменьшение только худшего случая перекрытий кратчайших каналов передачи данных.

Кратчайшие пути между процессорами 3 и 37, для которых при имитационном моделировании получено максимальное время передачи данных, лежат в пределах выделенного на рис. 2в блока. Наименьшее количество загруженных процессоров – 4 – входит в кратчайшие пути 3-4-5-13-21-29-37, 3-4-12-13-21-29-37, 3-11-12-13-21-29-37, поэтому вероятность того, что данные пути окажутся наименее загруженными, наиболее высокая.

В любом из описанных выше вариантов кратчайших маршрутов занятыми являются только процессоры 3, 21, 29 и 37, из которых процессоры 21, 29 и 37 лежат на одной вертикали. Рассчитаем минимально возможную суммарную задержку между 3 и 37 процессорами с учётом длин перекрывающихся каналов 3-37, 21-29, 21-37 и 29-37:

$$T_{\text{SUM}}=(24 \cdot 6 \cdot 1+24 \cdot 1 \cdot 2+24 \cdot 2 \cdot 1) \cdot 10=240 \cdot (6+2+2)=2400 \text{ нс.}$$

Аналогично рис. 2а и 2б, на рис. 2в показано минимальное количество перекрывающихся обменов данными на межпроцессорных каналах выделенного пути.

Рассчитаем худший случай перекрытий пути 3-4-5-13-21-29-37 при допущении, что все однонаправленные обмены между 3, 21, 29 и 37 процессорами будут проложены по данному пути. Тогда фактический список перекрытий данного пути будет следующим: 3-21, 3-29, 3-37, 21-29, 21-37, 29-37 и

$$T_{\text{SUM}}=(24 \cdot 4+24 \cdot 5+24 \cdot 6+24 \cdot 1 \cdot 2+24 \cdot 2) \cdot 10=240 \cdot (4+5+6+2+2)=4560 \text{ нс.}$$

Так как оба полученных результата меньше минимально возможной суммарной задержки, полученной для пары процессоров 17-29 в варианте размещения, полученном с применением минимаксного критерия, можно

сделать вывод о целесообразности применения минимаксиминного критерия при планировании размещения параллельных подпрограмм в матричных мультипроцессорах.

На основании полученных значений коммуникационной задержки для худших случаев перекрытий получена минимальная оценка реальная производительность мультипроцессора: при использовании минимаксного критерия – 0,32 GFLOP/s, при использовании минимаксиминного критерия – 0,48 GFLOP/s. Данные результаты позволяют сделать вывод, что применение минимаксиминного показателя позволяет получить на 50% большую реальную производительность по сравнению с применением минимаксного показателя в рассматриваемой задаче.

Список литературы

1. Бобынцев, Д.О. Минимаксиминный критерий оценки качества размещения параллельных подпрограмм в матричных мультиконтроллерах / Д.О. Бобынцев, Д.Б. Борзов // Известия Юго-Западного государственного университета. Серия. Управление, вычислительная техника, информатика. Медицинское приборостроение. – 2012. – №2. – Ч.1. – С.27-31.
2. Борзов, Д.Б. Методика планирования размещения задач в матрично-торроидальных базовых блоках кластерных мультиконтроллеров. / Д.Б. Борзов, Б.И. Мараят. Деп. в ВИНТИ 18.07.06 г., №961-В 2006.
3. Калачёв, А. Высокопроизводительные многоядерные процессоры для встраиваемых систем / А. Калачёв // Компоненты и технологии. – 2010. – №2. – С.92-102.

D.O. Bobyntsev, Southwest State University (Kursk) (e-mail daniel8728@yandex.ru)

E.I. Vatutin, Candidate of Technical Sciences, Associate Professor, Southwest State University (Kursk) (e-mail evatutin@rambler.ru)

V.S. Titov, Doctor of Technical Sciences, Professor, Southwest State University (Kursk) (e-mail titov-kstu@rambler.ru)

ESTIMATION OF MATRIX MULTIPROCESSOR PERFORMANCE USING PARALLEL ALGORITHM FOR THE N-BODY PROBLEM SOLVING

The partitioning of N-body problem to the group of parallel tasks is showed. The peak and real performances estimation for parallel computing system implementing parallel N-body problem solving algorithm using 8×8 matrix multiprocessor is given. Results of computing experiment with of scheduling parallel tasks using two different criteria of communication delay estimation is obtained.

Keywords: *parallel tasks scheduling, matrix multiprocessors, N-body problem, communication delay, MinMax criterion, MinMaxMin criterion.*