

ПЕРСПЕКТИВНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

УДК 519.68 : 681.324

Э. А. Монахова, О. Г. Монахов

(Новосибирск)

КОЛЛЕКТИВНЫЕ ОБМЕНИ В ЦИРКУЛЯНТНЫХ СЕТЯХ
ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ*

Рассматриваются проблемы организации трансляционных и полных обменов в циркулянтных сетях, используемых в качестве сетей связи параллельных вычислительных систем. Исследуются оптимальные циркулянтные сети, имеющие минимум диаметра и минимум среднего расстояния и соответственно максимум показателей надежности, связности и производительности при заданных стоимостных ограничениях. Представлен обзор ранее полученных результатов, а также новые результаты. Рассматривается полный обмен в модели передачи сообщений для случая, когда взаимодействующие процессоры могут обмениваться вплоть до фиксированного числа p пакетов на каждом шаге (p -полный обмен). Устанавливается общий метод вычисления нижних границ времени выполнения p -полного обмена в циркулянтных сетях. Предложены параллельные децентрализованные вершинно-инвариантные алгоритмы трансляционного и p -полного обменов для двумерных оптимальных циркулянтных сетей, обеспечивающие минимум времени выполнения и минимум нагрузки сообщений в сети при коллективных обменах.

Введение. При реализации большинства параллельных алгоритмов используются обмены данными между процессорами вычислительных систем, которые существенно влияют на эффективность выполнения параллельных программ. Основные схемы коллективных межпроцессорных коммуникаций — трансляционный и полный обмены — используются при 90 % параллельных алгоритмов. Трансляционный обмен (one-to-all broadcast) — это операция, где единственный процессор (источник) должен послать идентичные данные в сообщении (пакете) всем другим процессорам системы. Полный (или трансляционно-циклический) обмен (all-to-all broadcast) является обобщением трансляционного обмена, в котором все процессоры одновременно иницируют трансляционный обмен. Рассматриваемые типы обменов изучаются для графов Кэли при различных коммуникационных моделях (см. [1—4]). Мы рассматриваем полнодуплексную модель коммутации сообщений (пакетов). В такой модели протокол состоит из последовательности шагов, и в течение каждого шага каждый процессор может посылать (и получать) пакеты от всех своих соседей. Исследуем случай, когда на каждом шаге взаимодействующие процессоры могут обмениваться вплоть до установленного числа p пакетов. Полный обмен в графах Кэли при $p = 1$ приведен в [4]. Рассмотрим случай $p > 1$. Будем называть его p -полным обменом. В [4] авторы указали нижнюю границу времени p -полного обмена (число шагов), равную $\lceil (N - 1) / \delta p \rceil$, N — число процессоров, δ — минимальная степень вершины графа. Аналогичная проблема рассматривается в [3] для тороидальных решеток при ограниченном размере буферов. Ряд статей исследует проблему оценки минимального вре-

* Работа частично поддержана РФФИ (проекты № 97-01-00884; 96-01-01632).

мени p -полного обмена для других типов графов и моделей связи. Рассмотрим проблему установления нижних границ времени p -полного обмена в циркулянтных сетях, которые являются симметричными графами Кэли. Представим общий метод вычисления нижних границ для циркулянтных графов. Эффективность алгоритмов основных схем обменов непосредственно зависит от топологии связей между процессорами, а именно определяется диаметром графа межпроцессорных связей (максимум минимальных расстояний между каждой парой процессоров). Поэтому применение оптимальных графов в качестве сетей связи параллельных вычислительных систем позволяет реализовать коллективные обмены за минимальное время (для данного числа процессоров) без передачи лишних копий сообщений. Представлены эффективные алгоритмы трансляционного и p -полного обменов для двумерных оптимальных циркулянтных сетей. Циркулянтные сети широко изучаются [2, 5—15]. Частные случаи этих графов реализованы в параллельных системах Iliac IV, MPP, Intel Paragon XP/S, CRAY T3D.

1. Определения. Циркулянтная сеть (circulant network) определяется как граф $G(N; s_1, s_2, \dots, s_n)$ с N вершинами, помеченными целыми числами по модулю N , имеющими $i \pm s_1, i \pm s_2, \dots, i \pm s_n \pmod{N}$ вершины, смежные с каждой вершиной i . Числа s_1, s_2, \dots, s_n называются образующими. Степень $\delta = 2n$ (n — размерность графа) вершины в ненаправленном графе G — число ребер, инцидентных ей. Заметим, что, наряду с торами (toroidal mesh) и гиперкубами, циркулянтные графы принадлежат к классу КАИС-структур [12], которые являются неориентированными групп-графами конечных абелевых групп автоморфизмов. Синтез оптимальных циркулянтных графов — основная проблема графотеоретической оптимизации, представляющая обобщение проблемы (d, k) -графов [16]. Эта проблема заключается в поиске графа с минимумом диаметра и минимумом среднего расстояния среди всех циркулянтных графов, имеющих N вершин и размерность n . Обозначим это множество как $C(N, n)$. Диаметр G определяется как $D = \max_{i,j} d_{ij}$ (d_{ij} — длина кратчайшего пути из вершины i в вершину j), среднее расстояние G — как $\bar{D} = \sum_{ij} d_{ij} / N^2$.

Пусть для любого графа $G \in C(N, n)$ $K_{n,m}$ обозначает число вершин, которые могут быть достигнуты из вершины 0 с использованием самое большее m образующих, и пусть $K_{n,m}^*$ — верхняя граница для $K_{n,m}$. Определим $L_{n,m} = K_{n,m} - K_{n,m-1}$, $L_{n,m}^*$ будет верхней границей для $L_{n,m}$ на множестве $C(N, n)$. Значения $K_{n,m}^*, L_{n,m}^*$ для любых n, m получены в [12, 17, 7]. Для $n = 2$ $K_{2,m}^* = 2m^2 + 2m + 1$, $L_{2,m}^* = 4m$. Для $n = 3$ $K_{3,m}^* = (4m^3 + 6m^2 + 8m)/3 + 1$, $L_{3,m}^* = 4m^2 + 2$.

Графы, достигающие верхних границ, называются оптимальными, а именно граф $G \in C(N, n)$ оптимален, если $L_{n,m} = L_{n,m}^*$ для любого $0 \leq m \leq D^* - 1$ и $L_{n,D^*} = N - K_{n,D^*-1}^*$, где диаметр D^* получается из неравенства $K_{n,D^*-1}^* < N \leq K_{n,D^*}^*$. В [12] показано, что оптимальный граф имеет минимум D и \bar{D} и максимум надежности и связности среди всех графов из $C(N, n)$. В [18, 19] доказано, что сети с минимальным средним расстоянием являются оптимальными для проведения децентрализованных алгоритмов самодиагностики в вычислительной системе.

Известно, что полный обмен в графе G в рассматриваемой коммуникационной модели и при отсутствии ограничений на размер передаваемых сообщений может быть завершён за число шагов, равное диаметру графа. Проведем сравнение возможностей реализации при таких условиях схемы полного обмена в циркулянтных и гиперкубических сетях связи вычислительных систем. Гиперкуб позволяет реализовать схему полного обмена для $N = 2^3$ процессоров за δ шагов, где δ — диаметр гиперкуба и число линий связи у каждого процессора. В таблице приведены результаты сравнения гиперкубов и оптимальных циркулянтных графов по числу шагов (d) и числу линий связи у каждого процессора (δ), требуемых для реализации схемы полного обмена при одном и том же числе процессоров N . Представленные в таблице значения

| N | Гиперкуб | | Циркулянтный граф | | | | | | | | {s ₁ , s ₂ , ..., s _n } |
|------|----------|----|-------------------|---|---|---|---|----|----|----|--|
| | δ | d | δ | d | δ | d | δ | d | δ | d | |
| 8 | 3 | 3 | 4 | 2 | | | | | | | {1, 2} |
| 16 | 4 | 4 | 4 | 3 | | | | | | | {1, 6} |
| 32 | 5 | 5 | 4 | 4 | | | | | | | {1, 7} |
| 64 | 6 | 6 | 4 | 6 | 6 | 4 | | | | | {1, 4, 25}* |
| 128 | 7 | 7 | 4 | 8 | 6 | 4 | | | | | {1, 8, 54}* |
| 256 | 8 | 8 | | | 6 | 6 | 8 | 5* | | | {1, 9, 74, 103}* |
| 512 | 9 | 9 | | | 6 | 7 | 8 | 5 | | | {1, 15, 56, 149}* |
| 1024 | 10 | 10 | | | 6 | 9 | 8 | 6 | 10 | 6* | {1, 6, 33, 122, 347}* |

образующих оптимальных (или субоптимальных, помеченных звездочкой) циркулянтных графов получены с помощью параллельного алгоритма синтеза, разработанного авторами. Результаты сравнения показывают, что реализация схемы полного обмена в циркулянтных графах требует меньшего числа шагов обмена и меньшего числа линий связи у каждого процессора при одинаковом числе процессоров в системе. Таким образом, циркулянтные графы оказываются эффективнее гиперкубов при решении задач, использующих схему полного обмена, а именно: систем линейных алгебраических уравнений методом простой итерации, Жордана, Зейделя; систем нелинейных алгебраических уравнений; транспортной задачи; обращения матриц методом пополнения; систем обыкновенных дифференциальных уравнений методом Рунге — Кутты; дифференциальных уравнений в частных производных; граничных задач для нелинейных систем; случайного поиска с адаптацией; упорядочения массивов и др.

В качестве модели сетей связи параллельных систем, наряду с циркулянтными графами произвольных размерностей, в данной работе будут рассмотрены оптимальные двумерные циркулянтные графы с аналитически вычисляемым описанием $G(N; s, s+1)$, где $s = \lfloor (\sqrt{2N-1} - 1)/2 \rfloor$. Эти графы были получены в [6, 7, 9, 14]. Оптимальные графы $G(N; s, s+1)$ обладают следующими свойствами: а) существование для любого натурального $N > 4$; б) минимум D и минимум \bar{D} , равные нижним границам, и максимум надежности и связности среди всех двумерных циркулянтных графов с N вершинами; в) одинаковый вид описания $(N; s, s+1)$ для данного s для всех $2s^2 - 1 \leq N \leq 2s^2 + 4s + 3$.

Геометрическое представление рассматриваемого графа показано на рис. 1, а. Числа в квадратах — это номера вершин. Две вершины i и j связаны ребром, если $(i - j) \bmod N \in \{s, s+1, -s, -(s+1)\}$. Для упрощения не все ребра показаны. При таком представлении все вершины графа лежат внутри ромба с диагональю $2D^*$. Число D^* есть диаметр рассматриваемого графа: $D^* = \lfloor (\sqrt{2N-1} - 1)/2 \rfloor$ [6]. Рис. 1, а иллюстрирует граф $G(35; 4, 5)$. Если N увеличивается от $2s^2 - 1$ до $2s^2 + 4s + 3$, ромб последовательно заполняется новыми вершинами (квадраты, обозначенные пунктиром) в порядке, указанном стрелками.

Пусть с вершинами графа будут сопоставлены процессоры, а с ребрами — линии связи между смежными процессорами. Входные (выходные) полюсы каждого процессора пронумерованы числами 1—4 (рис. 1, б) и соответствуют образующим $s+1, s, -(s+1), -s$.

2. Нижние границы. В принятой модели связи протоколы для трансляционного и полного обменов состоят из последовательности шагов, и в течение

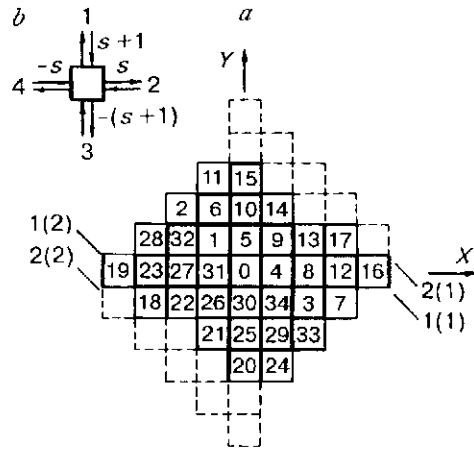


Рис. 1. Геометрическое представление оптимального циркулянтного графа $G(35; 4, 5)$ (a) и нумерация выходных (входных) полюсов процессоров (b)

каждого шага каждый процессор может посылать и получать различные сообщения от всех соседей. Каждый процессор может посылать (и получать) до установленного числа p пакетов через каждую линию связи на каждом шаге. Обозначим $T_b(p, G)$ и $T_g(p, G)$ — соответственно время трансляционного и p -полного обменов, которые определяются числом шагов, необходимых для завершения протоколов в сети G при вышеупомянутых условиях, $T_b^*(p, G)$ и $T_g^*(p, G)$ — минимум времени трансляционного и p -полного обменов. Определим теперь нижние границы числа шагов, требуемых для завершения p -полного обмена для любого циркулянтного графа произвольной степени при рассматриваемой коммуникационной модели.

Утверждение 1. Минимальное число шагов, необходимых для завершения процесса p -полного обмена в циркулянтном графе $G \in C(N, n)$, не менее чем

$$T_g^*(p, G) \geq \frac{N - K_{n,k}^*}{\delta p} + k,$$

где k получается из неравенства $L_{n,k}^* \leq \delta p < L_{n,k+1}^*$.

Доказательство. Заметим, что одинаковый вид процесса p -полного обмена из любой вершины следует из свойств циркулянтных графов. Имеем для любого графа $G \in C(N, n)$ $L_{n,i} \leq L_{n,i}^*$. Не более чем один пакет ($L_{n,1}^*/\delta$) может быть передан, используя данную линию связи, на первом шаге p -полного обмена, не более чем $L_{n,2}^*/\delta$ пакетов — на втором шаге, и т. д.; на шаге k и далее самое большее p пакетов может быть передано, где k — минимальное число, которое удовлетворяет неравенствам $L_{n,k}^* \leq \delta p < L_{n,k+1}^*$. Для завершения алгоритма общее количество $(N - 1)/\delta$ пакетов должно пройти через линию связи. Минимальное число шагов (обозначим его через t) должно удовлетворять следующему неравенству:

$$\sum_{i=1}^k L_{n,i}^* + (t - k)\delta p \geq N - 1.$$

Поскольку $K_{n,k}^* = \sum_{i=0}^k L_{n,i}^* = 1 + \sum_{i=1}^k L_{n,i}^*$, получаем

$$T_g^*(p, G) \geq \frac{N - K_{n,k}^*}{\delta p} + k.$$

Примечание. Полученная нижняя граница $T_g^*(p, G)$ улучшает нижнюю границу, данную в [4] для графов Кэли, где оптимальное время p -полного обмена — $\lceil (N-1)/\delta p \rceil$ шагов; δ — минимальная степень графа.

Утверждение 2. Минимальное число шагов, необходимое для завершения процесса p -полного обмена в циркулянтных графах со степенью 4, не менее чем

$$T_g^*(p, G) \geq \frac{N-1}{4p} + \frac{p-1}{2}.$$

Доказательство. При $\delta = 4$ из $L_{n,k}^* \leq \delta p < L_{n,k+1}^*$ и выражений для $K_{2,m}^*$ и $L_{2,m}^*$ следует, что $k = p$ и

$$T_g^*(p, G) \geq \frac{N-1}{4p} + p - \frac{2p^2+2p}{4p} = \frac{N-1}{4p} + \frac{p-1}{2}.$$

Утверждение 3. Оптимальное число p , необходимое для достижения минимального числа шагов завершения процесса p -полного обмена в циркулянтных графах со степенью 4, не менее чем

$$\frac{2D+1}{2} \pm \left(\frac{(2D+1)^2}{4} - \frac{N-1}{2} \right)^{1/2}.$$

Доказательство. Чтобы достичь минимального числа шагов, t должно равняться диаметру. Поэтому ищем оптимальное значение p из следующего равенства:

$$D = (N-1)/4p + (p-1)/2.$$

Это соответствует следующему оптимальному числу пакетов:

$$p^* \geq D + \frac{1}{2} \pm \left(\frac{(2D+1)^2}{4} - \frac{N-1}{2} \right)^{1/2}.$$

Примечание. Для оптимальных циркулянтных графов с $N = 2D^2 + 2D + 1$ имеем $p^* = D$.

Утверждение 4. Минимальное число шагов, необходимое для завершения процесса p -полного обмена в циркулянтных графах со степенью 6, не менее чем

$$T_g^*(p, G) \geq \frac{N-1}{6p} + k \left(1 - \frac{2k^2+3k+4}{9p} \right),$$

где $k = \left\lceil \left(\frac{3p-1}{2} \right)^{1/2} \right\rceil$.

Доказательство. Значение k должно удовлетворять неравенствам $4k^2 + 2 \leq 6p < 4k^2 + 8k + 6$. Решая квадратное уравнение $2k^2 + 4k + 3 - 3p = 0$, получим $k_1 = \left(\frac{3p-1}{2} \right)^{1/2} - 1$. Выражение для k следует из неравенств

$$\left(\frac{3p-1}{2} \right)^{1/2} - 1 < k \leq \left(\frac{3p-1}{2} \right)^{1/2}.$$

3. Децентрализованные алгоритмы трансляционного и полного обменов. Рассмотрим проблему организации трансляционного обмена для двумерных оптимальных циркулянтных сетей и затем два различных оптимальных алгоритма p -полного обмена для них.

Постановка проблемы: организовать передачу сообщений из любого процессора во все другие (трансляционный обмен) и из каждого процессора систе-

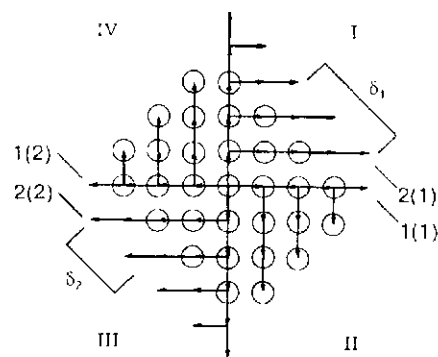


Рис. 2. Построение минимального покрывающего дерева для графа $G(N; 4,5)$

мы всем другим (полный обмен) за минимальное время и без дублирования сообщений. Последнее требование вызвано необходимостью минимизировать нагрузку в сети при коллективных обменах.

3.1. *Алгоритм трансляционного обмена.* Решение проблемы организации трансляционного обмена заключается в построении минимальных покрывающих деревьев с корнем в источнике и передаче копий сообщений только по направленным ребрам этих деревьев. Данный метод был предложен для циркулянтных графов в [2] и для произвольных графов в [20]. Позднее покрывающие деревья применялись для организации трансляционного обмена в гиперкубах [21]. Вид этих деревьев одинаков из любой вершины-источника вследствие регулярности связей. С увеличением N особенности рассматриваемых оптимальных графов делают возможным сохранение вида построенного дерева при добавлении необходимых ребер на последнем ярусе. На рис. 2 и 3 из возможных минимальных покрывающих деревьев показаны два для графов $G(N; 4,5)$, $31 \leq N \leq 51$. Вершина-источник расположена в центре, стрелки на ребрах указывают направления передач копий сообщений между смежными процессорами. Покрывающее дерево для $N = 31$ (минимальное число процессоров, при котором описание $(N; 4,5)$ оптимально) обозначено стрелками и кружками. Цифры указывают, в каком порядке новые ребра покрывающего дерева появляются при увеличении N от $2s^2 - 1$ до $2s^2 + 4s + 3$. Для более однородного заполнения квадрантов новые вершины появляются поочередно в противоположных квадрантах. В дальнейшем будем рассматривать алгоритмы трансляционного и p -полного обменов для дерева, представленного на рис. 2.

Рассмотрим параллельный децентрализованный алгоритм трансляционного обмена, который является вершинно-инвариантным и инвариантным к числу N процессоров в системе. Структура сообщения для трансляционного обмена следующая: $msg := \{T, R, U, V\}$, где T — текст сообщения; $R \in \{0, 1, \dots, D^*\}$ — счетчик расстояния между процессором, выполняющим алгоритм, и процессором-источником; $U \in \{0, 1\}$, $U + 1$ — число выходных полюсов, по которым процессор, выполняющий алгоритм, передаст копии сообщения; $V \in \{0, 1, \dots, D^* - 1\}$ определяет конец работы алгоритма при $R = D^* - 1$ (следует послать копию сообщения только для $V > 0$).

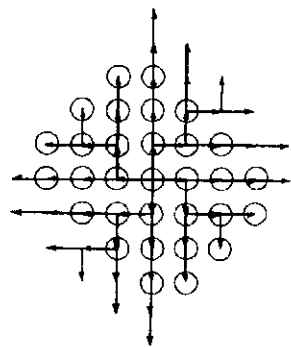


Рис. 3. Вариант минимального покрывающего дерева для графа $G(N; 4,5)$

Если необходимо передать копии сообщения соседям, алгоритм трансляционного обмена формирует параметры R, U, V , определяет номера выходных полюсов, а затем устанавливает сообщение в выходную очередь.

При изменении N в диапазоне оптимальности будут выделены два интервала (где $s > 1$ — любое положительное целое):

$$2s^2 - 1 \leq N \leq 2s^2 + 2s + 1, \quad (1)$$

$$2s^2 + 2s + 2 \leq N \leq 2s^2 + 4s + 3. \quad (2)$$

Пусть $\delta = N - (2s^2 - 1)$ для интервала (1) и $\delta = N - (2s^2 + 2s + 1)$ для интервала (2). Вычислим δ_1, δ_2 как число ребер покрывающего дерева, расположенных на последнем ярусе от вершины-источника соответственно в квадрате I (III) (см. рис. 2): $\delta_1 = \lceil \delta/2 \rceil, \delta_2 = \delta - \delta_1$. Используя эти значения, алгоритм трансляционного обмена в вершине-источнике формирует параметр V .

Предложенный алгоритм представлен в программах 3.1 и 3.2. Алгоритм начинает работу в транзитном процессоре (или источнике) после получения

Программа 3.1. Алгоритм трансляционного обмена в процессоре-источнике

```

begin
  R := 0;
  U := 1;
  if inequality (1) and  $\delta_1(\delta_2) > 0$  then  $V_{1(3)} := \delta_1(\delta_2) - 1$ ;
  else  $V_{1(3)} := \delta_1(\delta_2)$ ;
  send msg from input 0 to output 1 (3);
  if inequality (2) then  $V_{2(4)} := 0$ ;
  else
    if  $\delta_1(\delta_2) > 0$  then  $V_{2(4)} := s$ ;
    else  $V_{2(4)} := s - 1$ ;
  send msg from input 0 to output 2 (4);
end

```

Программа 3.2. Алгоритм трансляционного обмена в транзитном процессоре

```

begin
  send msg from input  $i$  to output 0;
  R :=  $R^m + 1$ ;
  if  $(R = D^*)$  or  $((R = D^* - 1) \text{ and } (V^m = 0))$  then stop;
  else  $(V := V^m)$ ;
  if  $(U^m \neq 0)$  then
    begin
      U := 0;
      send msg from input  $i$  to output  $(i + 2) \bmod 4 + 1$ ;
      U :=  $U^m$ ;
      if  $(R \neq D^* - 1)$  and  $(V^m \leq R)$  and  $(V^m \neq 0)$  then  $V := 0$ ;
      if  $(R = D^* - 1)$  then stop;
    end;
  send msg from input  $i$  to output  $(i + 1) \bmod 4 + 1$ ;
  stop;
end

```

сообщения с входа 1—4 (или 0). В программе 3.2 параметры R^m, U^m, V^m поступившего сообщения с входа i преобразуются в параметры R, U, V для сообщения в выходной очереди.

Использование параметра V позволяет избежать передачи лишних копий сообщений на расстоянии $R = D^* - 1$ шагов от источника. Если сообщение с $R = D^*$ прибывает в процессор, тогда оно должно быть принято, потому что пришло первый раз, после этого процессор заканчивает алгоритм трансляционного обмена. Принимая в рассмотрение вышеупомянутое, можно прийти к следующему.

Утверждение 5. Для циркулянтного графа $G(N; s, s + 1)$ с любым $N > 4$ при использовании предложенного алгоритма трансляционного обмена

$$T_b(1, G) = T_b^*(1, G) = D^*.$$

3.2. Алгоритмы p -полного обмена. Алгоритмы p -полного обмена основаны на рассмотренном трансляционном обмене как суперпозиции этих алгоритмов, выполненных для всех процессоров системы. Выход на p -полный обмен реализуется каждым процессором, если появляется сообщение. В результате каждый процессор передаст данные всем другим без лишних копий и получает данные от всех процессоров.

Первый представленный алгоритм p -полного обмена [2] вводит упорядочивание в обработку поступающих пакетов. Как известно, номера выходных полюсов для поступающих пакетов определяются алгоритмом трансляционного обмена. Последовательность принятия с входных полюсов и обработки поступающих пакетов для p -полного обмена будет определяться согласно правилам, которые следуют из принятой коммуникационной модели (число p пакетов в этом случае ограничено диаметром графа в отличие от большинства рассматриваемых в литературе алгоритмов, где на последнем шаге выполняемого протокола длина сообщения линейно зависит от N):

- 1) первым обработать пакет, пришедший на вход 0;
- 2) $i := 1$;
- 3) обработать $F_R(i)$ пакетов, пришедших на вход i ;
- 4) $i := i + 1$, если $i > 4$, тогда $i := 1$; идти на 3.

Функция $F_R(i)$ устанавливает число пакетов, прошедших расстояние R от источника до текущего процессора и прибывших на вход i . Значения $F_R(i)$ определяются следующим образом.

Если $R \in \{1, 2, \dots, D^* - 1\}$ и $i \in \{1, 2, 3, 4\}$, тогда $F_R(i) = R$. Если $R = D^*$ и $\delta_2(\delta_1) = s + 1$, тогда

$$F_R(2)(F_R(4)) = \delta_2(\delta_1) - 1,$$

$$F_R(1)(F_R(3)) = \begin{cases} s, & \text{если (1),} \\ 1, & \text{если (2).} \end{cases}$$

Если $R = D^*$ и $\delta_2(\delta_1) < s + 1$, тогда

$$F_R(2)(F_R(4)) = \delta_2(\delta_1),$$

$$F_R(1)(F_R(3)) = \begin{cases} s - 1, & \text{если (1),} \\ 0, & \text{если (2).} \end{cases}$$

Значения функции $F_R(i)$ становятся понятны, если рассмотреть дерево стока или дерево путей, которое порождается передачей пакетов в произвольный процессор системы, соответствующий центральному кружку ромба, от других процессоров. Это дерево представлено на рис. 4 и сформировано как

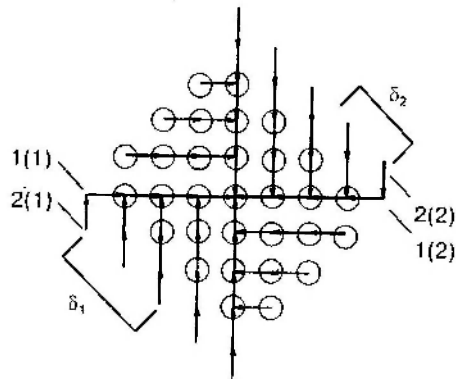


Рис. 4. Дерево стока для графа $G(N; 4,5)$

суперпозиция путей, ведущих в центральный процессор из каждого процессора по соответствующему ему дереву, представленному на рис. 2. Цифры задают последовательность появления новых ребер при изменении числа процессоров в системе. В этом случае все процессоры системы разделены на четыре области, соответствующие полюсам, по которым пакеты из этих областей прибывают в центральный процессор. Числа на рис. 5 указывают на порядок прибытия пакетов, находящихся на равных расстояниях от принимающего процессора. Алгоритм p -полного обмена заканчивает работу после обработки $F_R(i)$ пакетов, пришедших на полюс i для $i = 4$ и $R = D^*$.

Заметим, что

$$\sum_{i=1}^4 \sum_{R=1}^{D^*} F_R(i) = N - 1.$$

На основе анализа алгоритма p -полного обмена доказано следующее.

Утверждение 6. Для циркулянтного графа $G(N; s, s + 1)$ с любым $N > 4$ при предложенном алгоритме p -полного обмена

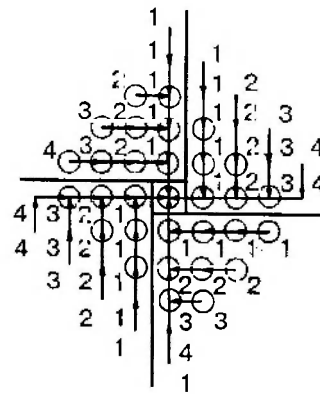
$$T_g(D, G) = T_g^*(D, G) = D^*.$$

Рассмотрим p -полный обмен в оптимальных циркулянтных графах с $N = 2D^2 + 2D + 1$ для любого фиксированного p . Пусть дерево трансляционного обмена — это дерево на рис. 2. Однако в отличие от рассмотренного выше алгоритма p -полного обмена на каждом шаге этого обмена взаимодействующие вершины могут обмениваться не более чем p пакетами.

Номера выходных полюсов для поступающих пакетов будут определяться предложенным алгоритмом трансляционного обмена, а последовательность принятия с входных полюсов и обработки поступающих пакетов для p -полного обмена — согласно правилам, которые следуют из принятой модели:

- 1) первым обработать пакет, поступивший на вход 0;
- 2) $i := 1$;

Рис. 5. Порядок прибытия пакетов, расположенных на равных расстояниях от источника



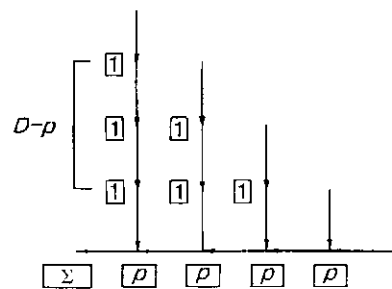


Рис. 6. Состояния вершин после $p - 1$ шагов алгоритма p -полного обмена

- 3) обработать все пакеты вплоть до p , прибывшие на вход i ;
- 4) $i := i + 1$, если $i > 4$, тогда $i := 1$; идти на 3.

В этом случае можно установить следующие результаты.

Утверждение 7. Для циркулянтного графа $G(N; D, D + 1)$ с $N = 2D^2 + 2D + 1$ минимальное время предложенного алгоритма p -полного обмена

$$T_g^*(p, G) = \left\lceil \frac{D(D+1)}{2p} - \frac{p-1}{2} \right\rceil + p - 1.$$

Доказательство. Для каждой вершины циркулянтного графа $G(N; D, D + 1)$ рассмотрим дерево стока. Все вершины графа G разделены на четыре области, соответствующие полюсам, по которым пакеты из этих областей прибывают в центральную вершину. Рассмотрим область, соответствующую одному из полюсов. Число вершин области — $\frac{N-1}{4} = \frac{D(D+1)}{2}$. После $p - 1$ шагов алгоритма состояния вершин в рассматриваемой области показаны на рис. 6.

Когда все оставшиеся пакеты придут в центральную вершину (самое большее p пакетов через одну линию связи в течение шага), алгоритм p -полного обмена будет завершен. Необходимо минимальное число шагов x должно удовлетворять неравенству

$$xp \geq \sum_{i=p}^D i = \frac{(D+p)(D+1-p)}{2}.$$

Отсюда $x = \left\lceil \frac{(D+p)(D+1-p)}{2p} \right\rceil$ и $T_g^*(p, G) = \left\lceil \frac{D(D+1)}{2p} - \frac{p-1}{2} \right\rceil + p - 1$.

Таким образом, получен оптимальный алгоритм для двумерных циркулянтных графов, и его время совпадает с нижней границей из утверждения 2.

Заключение. Рассмотрены основные схемы межпроцессорных обменов, включая трансляционный и полный обмены в сетях, представленных циркулянтными графами. Получен новый метод оценки нижних границ времени выполнения полного обмена при заданных ограничениях на число обмениваемых за один шаг пакетов в произвольных циркулянтных графах. Данный метод может быть применен для других классов однородных графов Коли. В частности, оценки, найденные для нижних границ числа шагов p -полного обмена, оказываются верными для всех подклассов КАИС-структур [12], включая гиперкубы и торы. Предложены параллельные децентрализованные алгоритмы трансляционного и p -полного обменов для оптимальных двумерных циркулянтных сетей. Получены оценки времени выполнения протоколов, зависящие от диаметра сетей. Показано существование минимального времени выполнения предложенных протоколов (и минимальной нагрузки сообщений) для оптимальных циркулянтных графов при реализации предложенных алгоритмов.

СПИСОК ЛИТЕРАТУРЫ

1. Hedetniemi S. M., Hedetniemi S. T., Liestman A. L. A survey of gossiping and broadcasting in communication networks // Networks. 1988. 18. P. 319.
2. Монахова Э. А. Алгоритмы межмашинных обменов и реконфигурации графов связей в вычислительных системах с программируемой структурой // Вычислительные системы. Новосибирск, 1982. Вып. 94. С. 81.

3. Konig J.-C., Rao P. S., Trystram D. Analysis of gossiping and broadcasting algorithms in torus with restricted bufferization capabilities // Technical Report IMAG. Grenoble, 1994.
4. Bermond J.-C., Kodate T., Perennes S. Gossiping in Cayley graphs by packets // Combinatorics and Computer Science (CCS'95). LNCS. 1996. 1120. P. 301.
5. Du D.-Z., Hsu D. F., Li Q., Xu Y. A combinatorial problem related to distributed loop networks // Networks. 1990. 20. P. 173.
6. Монахова Э. А. Об аналитическом задании оптимальных двумерных диофантовых структур однородных вычислительных систем // Вычислительные системы. Новосибирск, 1981. Вып. 90. С. 81.
7. Boesch F. T., Wang J.-F. Reliable circulant networks with minimum transmission delay // IEEE Trans. Circuits. Syst. 1985. CAS-32. P. 1286.
8. Monakhova E. A. Optimal circulant computer networks // Proc. Intern. Conf. on Parallel Computing Technologies, PaCT-91. Novosibirsk, USSR, 1991. P. 450.
9. Bermond J.-C., Iliades G., Peyrat C. An optimization problem in distributed loop computer networks // Third Intern. Conf. on Combinatorial Math. N. Y., USA, June 1985: Ann. New York Acad. Sci., 1989. 555. P. 45.
10. Bermond J.-C., Tzvieli D. Minimal diameter double-loop networks: Dense optimal families // Networks. 1991. 21. P. 1.
11. Zerovnik J., Pisanski T. Computing the diameter in multiple-loop networks // J. Algorithms. 1993. 14. P. 226.
12. Корнеев В. В. Макроструктура однородных вычислительных систем // Вычислительные системы. Новосибирск, 1974. Вып. 60. С. 17.
13. Воробьев В. А. Простейшие структуры однородных вычислительных систем // Там же. С. 35.
14. Beivide R., Herrada E., Balcazar J. L., Arruabarrena A. Optimal distance networks of low degree for parallel computers // IEEE Trans. Comput. 1991. 40, N 10. P. 1109.
15. Bermond J.-C., Comellas F., Hsu D. F. Distributed loop computer networks: a survey // J. Parallel Distributed Comput. 1995. 24. P. 2.
16. Memmi G., Raiffard Y. Some new results about the (d, k) graph problem // IEEE Trans. Comput. 1982. 31. P. 784.
17. Wong C. K., Maddocks T. W. A generalized Pascal's triangle // Fibonacci Quart. 1975. 13. P. 134.
18. Димитриев Ю. К. Самодиагностика модульных вычислительных систем. Новосибирск, Наука, 1993.
19. Димитриев Ю. К. Анализ самодиагностических свойств структур распределенных живучих вычислительных систем // Автоматика. 1996. № 5. С. 71.
20. Корнеев В. В., Монахов О. Г. О децентрализованном распределении заданий в вычислительных системах с программируемой структурой // Электронное моделирование. 1981. № 6. С. 15.
21. Johnsson S. L., Ho C.-T. Optimum broadcasting and personalized communication in hypercubes // IEEE Trans. Comput. 1989. 38. P. 1249.

Поступила в редакцию 30 июня 1997 г.