并行排序作业

一、 问题描述

1. 考虑 n 个处理器的环，设计将 n 个元素的双调排序映射到此环的算法，要求最小化通信开销。
2. 分析算法的并行时间，它是代价最优的吗？
3. 若处理器数目 p 小于元素数 n，是否能达到代价最优？

二、 算法设计

思路：以 16 个元素为例进行分析，编号从 0000 到 1111。通信性能正比于节点距离。
每个比较器比较的都是编号相差 1 位的元素，从最低位开始比较，假设最低位为第 0 位，最高位为 d-1 位，共 $\log n$ 个阶段，每个阶段最后一个步骤比较最后 1 位不同的元素，倒数第二个步骤比较倒数第二位不同的元素……

首先考虑相邻之间节点两两进行通信，如图 1。以贪心算法的的思路考虑，0000 下一步要通信的是 0010，将前两部分连接的方式有两种，如图 1(a)和图 1 (b)所示，两种方式通信代价之和都是 4，对于其他节点同理。所以可以考虑图 1 (a)的连接方式，此步连接效果如图 3 所示。下一步仍然有两种连接方式，如图 1(a)和图 1(b)所示，两种通信代价之和都是 16。所以从整体来看，通信代价最小的方式就是将处理器依次在环上进行映射，如图 1。

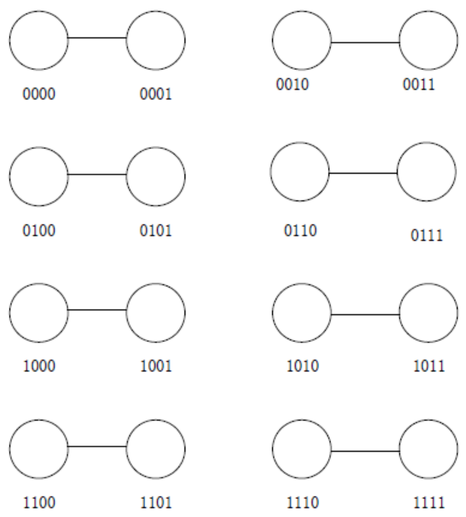

图 1 相邻节点之间两两通信

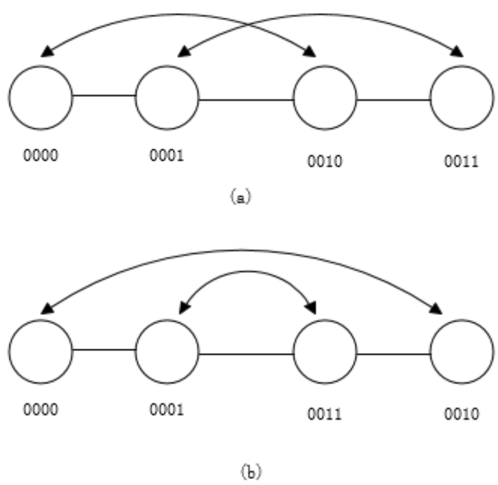

图 2 四个节点的两种通信连接方式

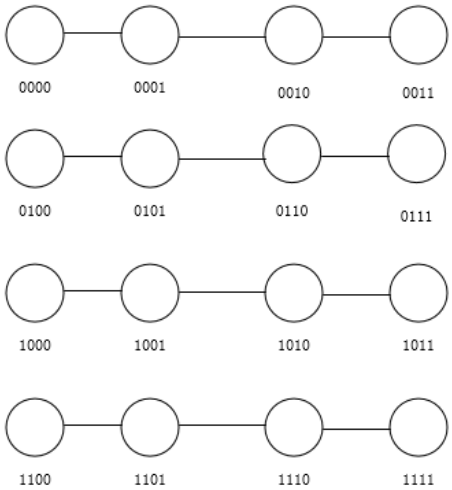

图 3 四组通信节点

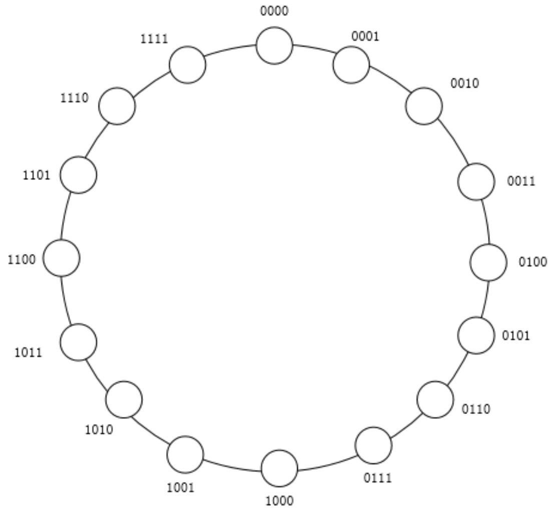

图 5 映射到环

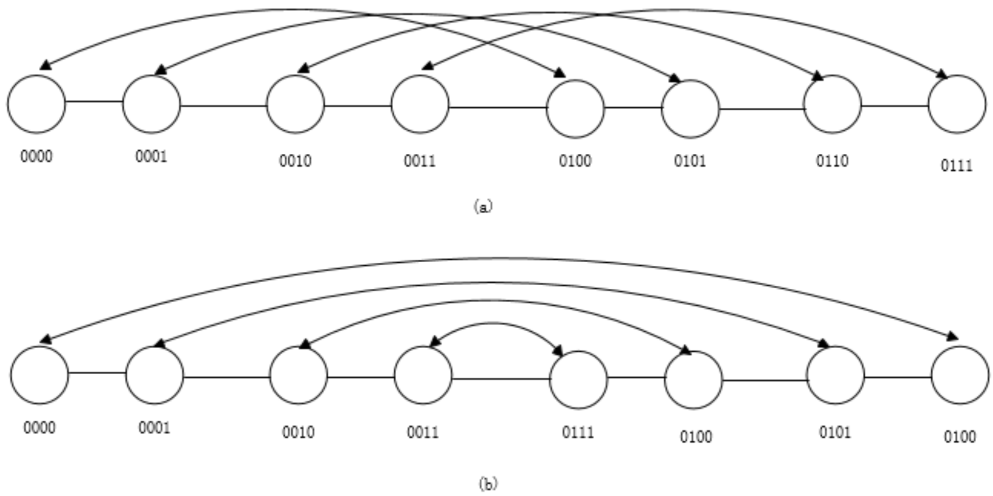

图 4 八个节点的两种通信连接方式

所以对于任意一个节点 label，在第 i 步，要进行 i 个阶段，每个阶段需要找到与自己通信的节点 label+2^i，如果 label 的第 j 位（当前交换维度）为 1 ，我需要保持最大，如果为 0，需要保持最小。假设通信时候不发生冲突，所有的通信都可以在理想时间内完成。

每个比较器比较的都是编号相差 1 位的元素，通信距离为 2^d 。

Algorithm 1 n 个元素的双调排序映射到环

```
Input: d, label
Output: sorted sequence
Procedure BITONIC_SORT(label, d)
Begin
  for i=0 to d-1 do
     for j:=i downto 0 do
          find the communication node label+2^i
```

```
        if jst bit of label =1
            comp_exchange_max(j);
        else
            comp_exchange_min(j);
    end for
end for
```

三、 算法分析

1. 在 n 个处理器，n 个元素的情况下，每个节点映射到一个处理器，通信性能正比于节点距离，一共有 $\log n$ 个阶段，总的通信代价为

$$\sum_{i=1}^{\log n}\sum_{j=1}^{i}2^j = 2n - \log n - 2 \approx 2n$$

   计算代价为 $\frac{\log n(\log n+1)}{2}$，即 $\Theta(\log^2 n)$， 所以 $T_p = \Theta(2n) + \Theta(\log^2 n)$

   $\cos t = nT_p = \Theta(n^2) + \Theta(n\log^2 n) = \Theta(n^2)$

   $T_s = \Theta(n\log n)$

   所以不是代价最优
2. 当处理器数目 p 小于元素数 n，每个处理器上要存 n/p 个数据，使用原算法，虚处理器的任务映射到对应物理处理器，将原来的比较-交换操作换成比较-分裂。

   初始化对每个进程 n/p 个元素进行排序，代价为 $\Theta(\frac{n}{p}\log(\frac{n}{p}))$；

   每一次比较分裂操作的代价为 $\Theta(\frac{n}{p})$；

   双调排序的步骤为 $\frac{\log p(\log p+1)}{2}$，即 $\Theta(\log^2 p)$

   通信代价为 $\Theta(n)$

   $T_p = \Theta(\frac{n}{p}\log\frac{n}{p}) + \Theta(\frac{n}{p}\log^2 p) + \Theta(n)$

   $T_s = \Theta(n\log n)$

$$S=\frac{T_s}{T_p}=\frac{\Theta(n\log n)}{\Theta(\frac{n}{p}\log\frac{n}{p})+\Theta(\frac{n}{p}\log^2 p)+\Theta(n)}$$

$$E=\frac{S}{p}=\frac{\Theta(\frac{n}{p}\log n)}{\Theta(\frac{n}{p}\log\frac{n}{p})+\Theta(\frac{n}{p}\log^2 p)+\Theta(n)}$$

$$E=\frac{1}{1-\Theta(\frac{\log p}{\log n})+\Theta(\frac{\log^2 p}{\log n})+\Theta(\frac{p}{\log n})}$$

如果想要达到代价最优，则 $E=\Theta(1)$，那么 $\dfrac{p}{\log n}=\Theta(1)$，可以达到代价最优

也就是当 $p=\Theta(\log n)$ 时，可以达到代价最优。

等效率函数 $W=\dfrac{E}{1-E}T_o(W,p)$，将 $p=\Theta(\log n)$ 代入

$$\begin{aligned}W&=T_o(W,p)\\&=pT_p-T_s\\&=\Theta(n\log\frac{n}{p})+\Theta(n\log^2 p)+\Theta(np)-\Theta(n\log n)\\&=\Theta(2^p(p-\log p))+\Theta(2^p\log^2 p)\\&=\Theta\left(2^p p\right)\end{aligned}$$

综上所述：

网络拓扑	等效率条件最大处理器数	对应的并行运行时间	等效率函数
环	$\Theta(\log n)$	$\Theta(n)$	$\Theta\left(2^p p\right)$