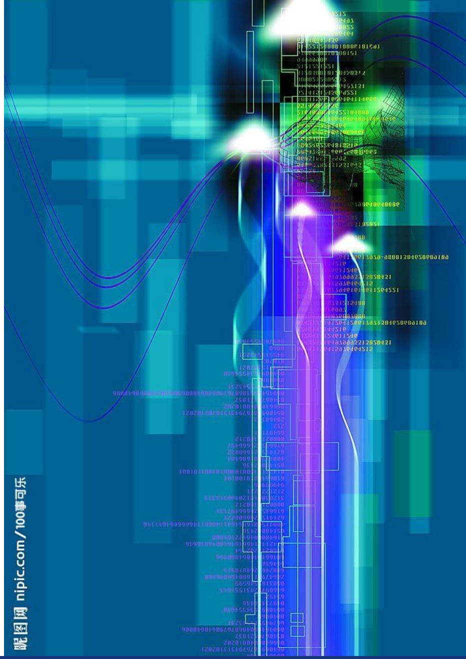
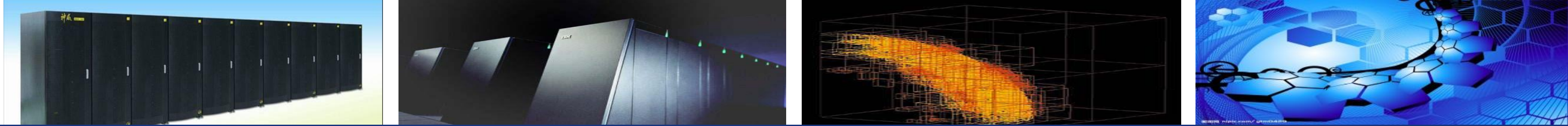


第四章 比较器网络的排序与选择算法

Sorting and Selection on Comparison Network



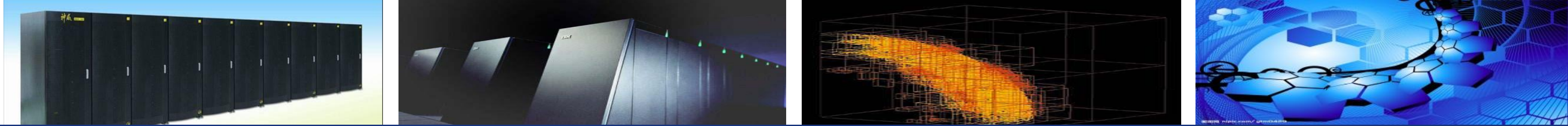
内容

Batcher归并、排序

- 比较操作和[0, 1]原理
- 奇偶归并网络
- 双调归并网络
- Batcher排序网络

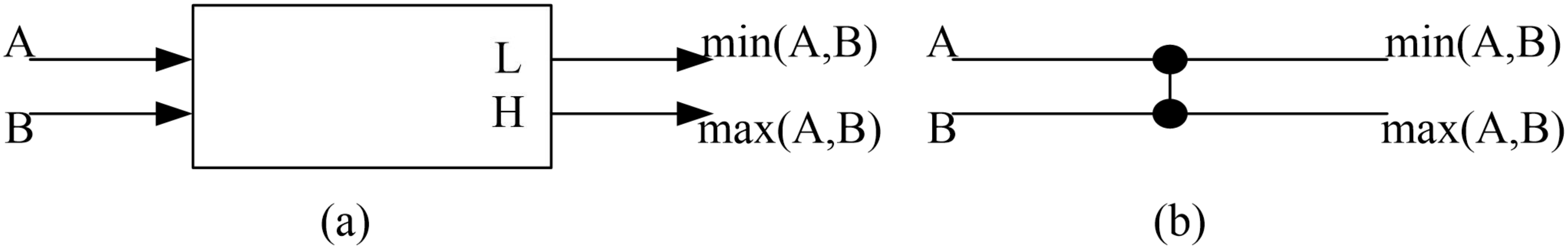
(m, n)-选择网络

- 分组选择网络
- 平衡分组选择网络



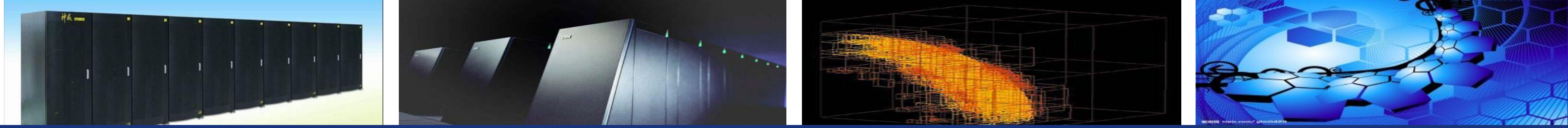
1 Batcher归并和排序－比较操作和[0, 1]原理

1. Batcher比较器



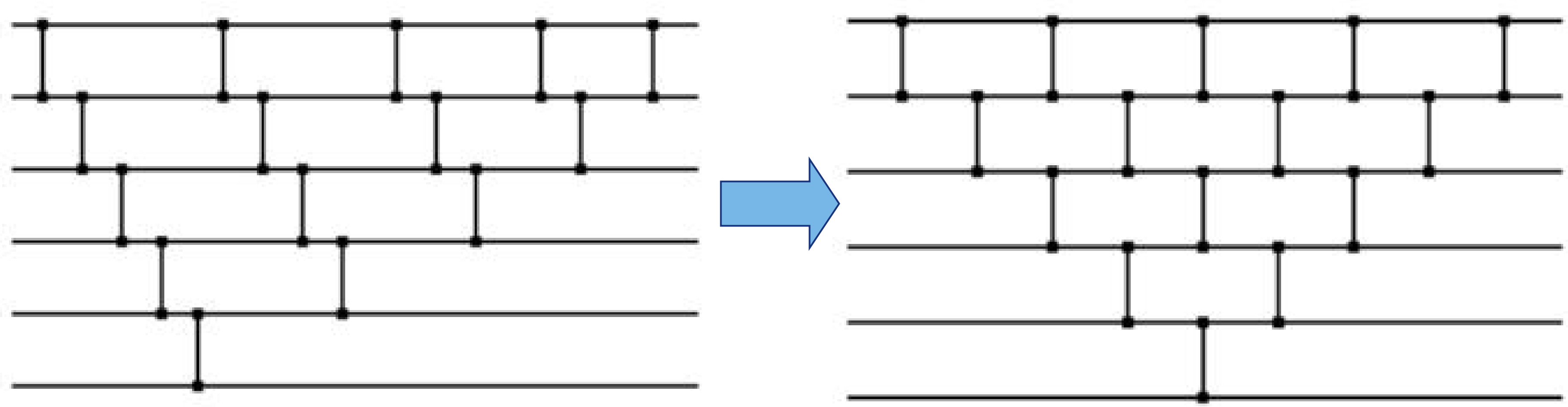
Batcher比较器

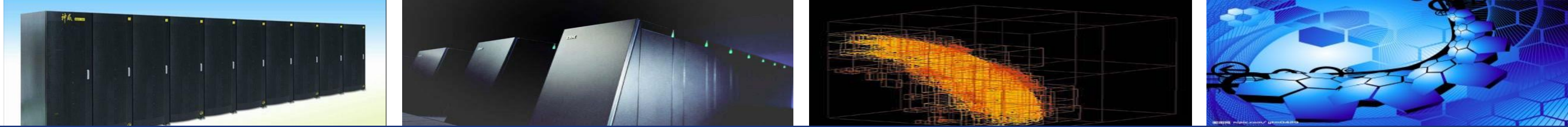
- 比较和条件交换操作: **CCI**
- 比较器网络：用**Batcher**比较器连成的，完成某功能的网络
- 假定：每次每个元素只能与另一个元素比较
- 比较器网络的参数：比较器数目、延迟级数



1 Batcher归并和排序—比较操作和[0, 1]原理

冒泡排序网络示意



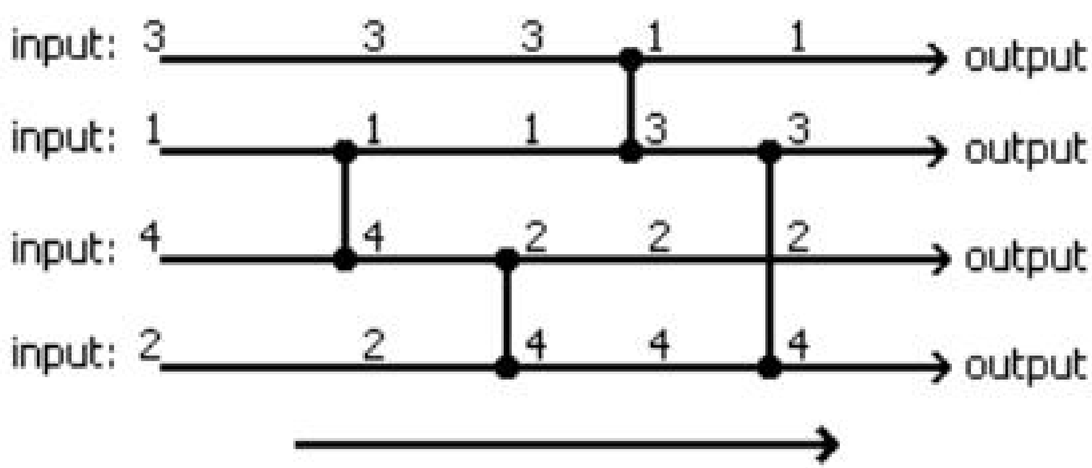


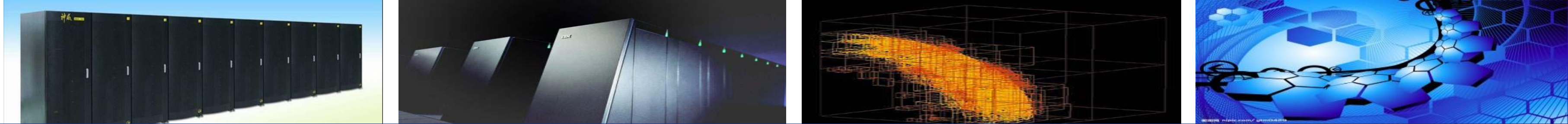
1 Batcher归并和排序—比较操作和[0, 1]原理

2. [0, 1]原理(定理4.1):

如果一个n输入的网络能排序所有2^n种0,1序列，那么它也能排序n个数的任意序列。

反证法！一个说明性的例子。





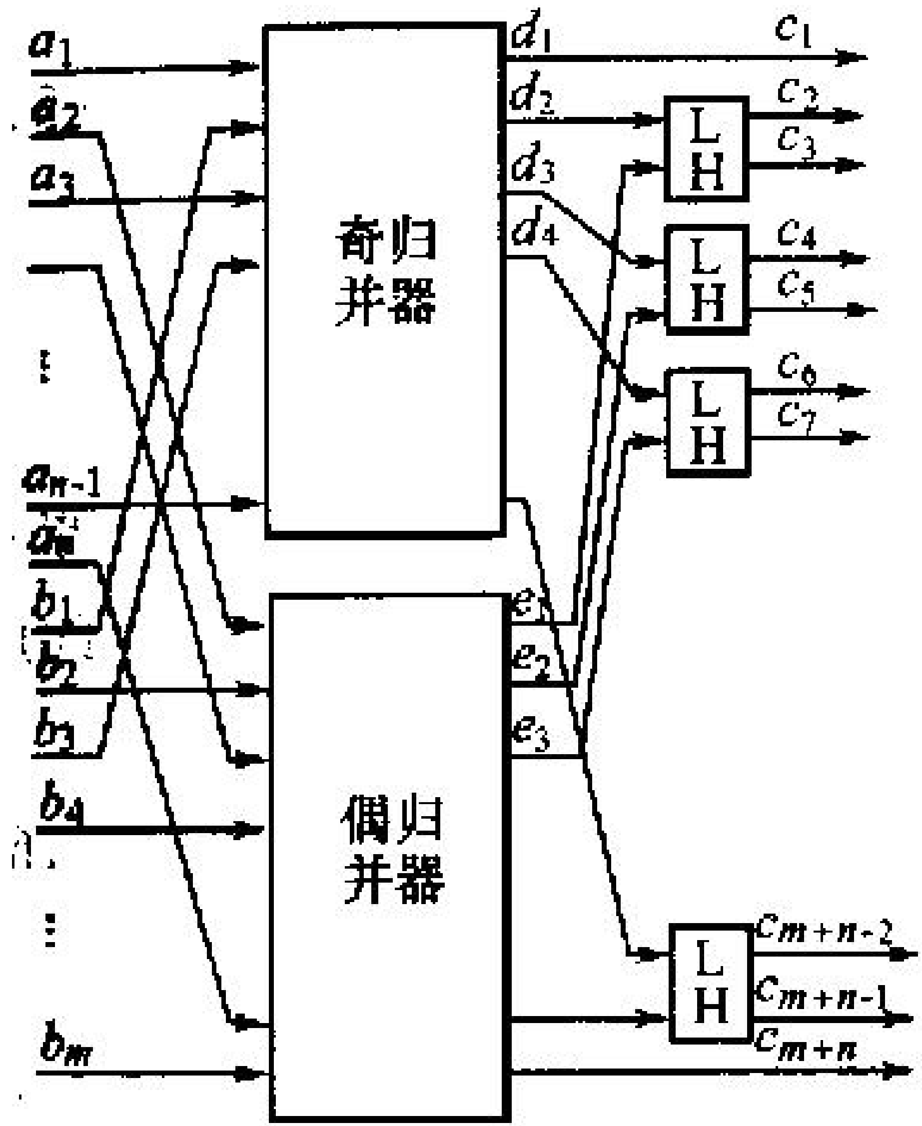
1 Batcher归并和排序—奇偶归并网络(O-E)

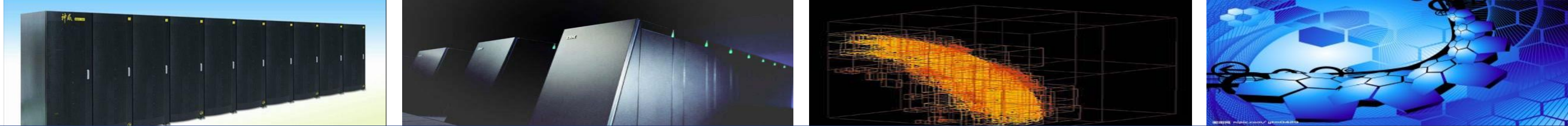
1. 网络构造

- 有序序列A: $a_1,a_2,\cdots,a_n$
  B: $b_1,b_2,\cdots,b_m$
- 归并思想：
  - A, B中奇数号元素进入奇归并器；
  - A, B中偶数号元素进入偶归并器；
  - 再将奇归并器与偶归并器的输出进行交叉比较

注: (m,n)规模划分为：

$$\begin{cases}(\lceil m/2\rceil,\lceil n/2\rceil)\text{奇}\\(\lfloor m/2\rfloor,\lfloor n/2\rfloor)\text{偶}\end{cases}$$

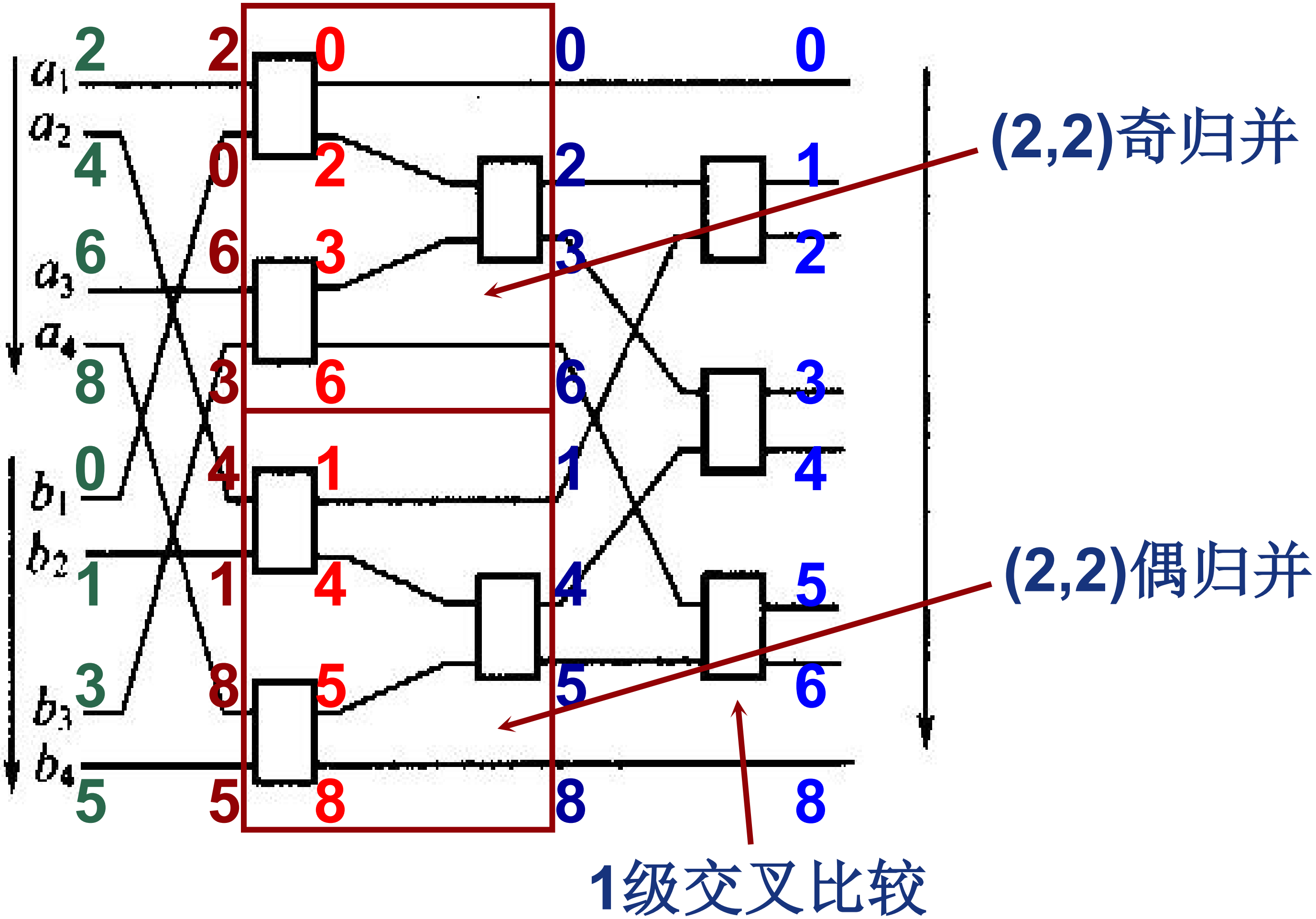


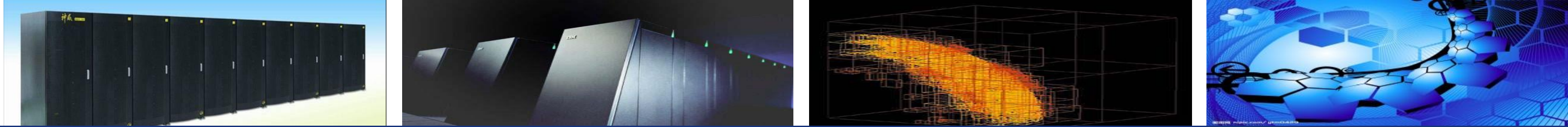


1 Batcher归并和排序—奇偶归并网络

2. 例：m=n=4　A=(2,4,6,8)　B=(0,1,3,5)

(4, 4)奇偶归并→2×(2, 2)奇偶归并＋1级交叉比较





1 Batcher归并和排序—奇偶归并网络

3. 复杂性分析

- 比较器个数

$$C_{OE}^{M}(m,n)=\begin{cases} mn & mn\le 1 \\ C_{OE}^{M}\left(\left\lceil \frac{m}{2}\right\rceil,\left\lceil \frac{n}{2}\right\rceil\right)+C_{OE}^{M}\left(\left\lfloor \frac{m}{2}\right\rfloor,\left\lfloor \frac{n}{2}\right\rfloor\right)+\left\lfloor \frac{m+n-1}{2}\right\rfloor & mn>1 \end{cases}$$

  - **Knuth** ==> $C_{OE}^{M}(n,n)=O(n\log n)$
  - 当m=n=2ᵗ时，不难推得

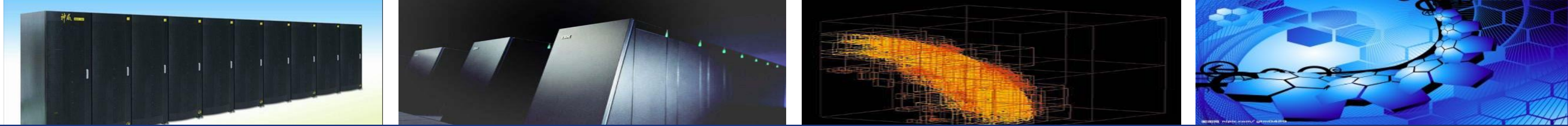
$$C_{OE}^{M}(n,n)=2C_{OE}^{M}(n/2,n/2)+\left\lfloor n-\tfrac{1}{2}\right\rfloor=2C_{OE}^{M}(n/2,n/2)+\left\lfloor (n-1)+\tfrac{1}{2}\right\rfloor$$

$$=2C_{OE}^{M}(n/2,n/2)+(n-1)=2(2C_{OE}^{M}(n/4,n/4)+n/2-1)+(n-1)$$

$$=2^{2}C_{OE}^{M}(n/2^{2},n/2^{2})+(n-2)+(n-1)=\ldots$$

$$=2^{t}C_{OE}^{M}(n/2^{t},n/2^{t})+\sum_{i=0}^{t-1}(n-2^{i})=nC_{OE}^{M}(1,1)+\sum_{i=0}^{t-1}n-\sum_{i=0}^{t-1}2^{i}$$

$$=n+tn-(n-1)=n\log n+1$$



1 Batcher归并和排序—奇偶归并网络

3. 复杂性分析

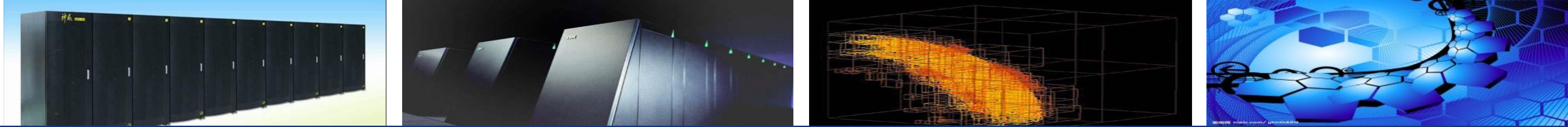
- 延迟级数：穿过网络任一路线上的最多比较器数目

$$D_{OE}^{M}(m,n)=\begin{cases}0 & m=0\text{或}n=0\\ 1 & m=n=1\\ 1+\max\left(D_{OE}^{M}(\lceil m/2\rceil,\lceil n/2\rceil),D_{OE}^{M}(\lfloor m/2\rfloor,\lfloor n/2\rfloor)\right) & \text{其他}\end{cases}$$

  - 一般地有

$$D_{OE}^{M}(m,n)=1+D_{OE}^{M}(\lceil m/2\rceil,\lceil n/2\rceil)$$

  - 当m=n=2ᵗ时，不难推得 $D_{OE}^{M}(n,n)=\log n+1$



1 Batcher归并和排序—双调归并网络

1. 定义及定理

- 定义4.5: 一个序列$a_1,a_2,\cdots,a_n$是双调序列(Bitonic Sequence)，如果：

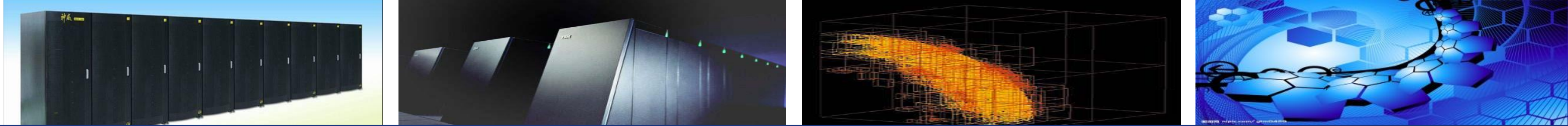
  (1)存在一个$a_k(1\leqslant k\leqslant n)$, 使得$a_1\geq\cdots\geq a_k\leq\cdots\leq a_n$成立；或者

  (2)序列能够循环移位满足条件(1)

- 示例：

  序列(1,3,5,7,8,6,4,2,0), (7,8,6,4,2,0,1,3,5)和(1,2,3,4,5,6,7,8)都是双调序列。

a_k



1 Batcher归并和排序—双调归并网络

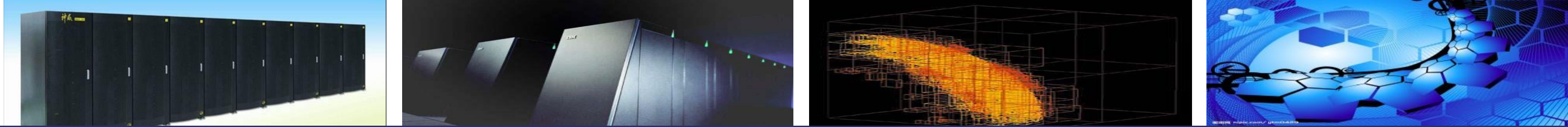
- 定理4.3(Batcher定理)：

设序列$a_1,\cdots,a_n,a_{n+1},\cdots, a_{2n}$是一个双调序列, 记

$b_i=\min\{a_i, a_{i+n}\}$ ==> $MIN=\{b_1,\cdots,b_n\}$,

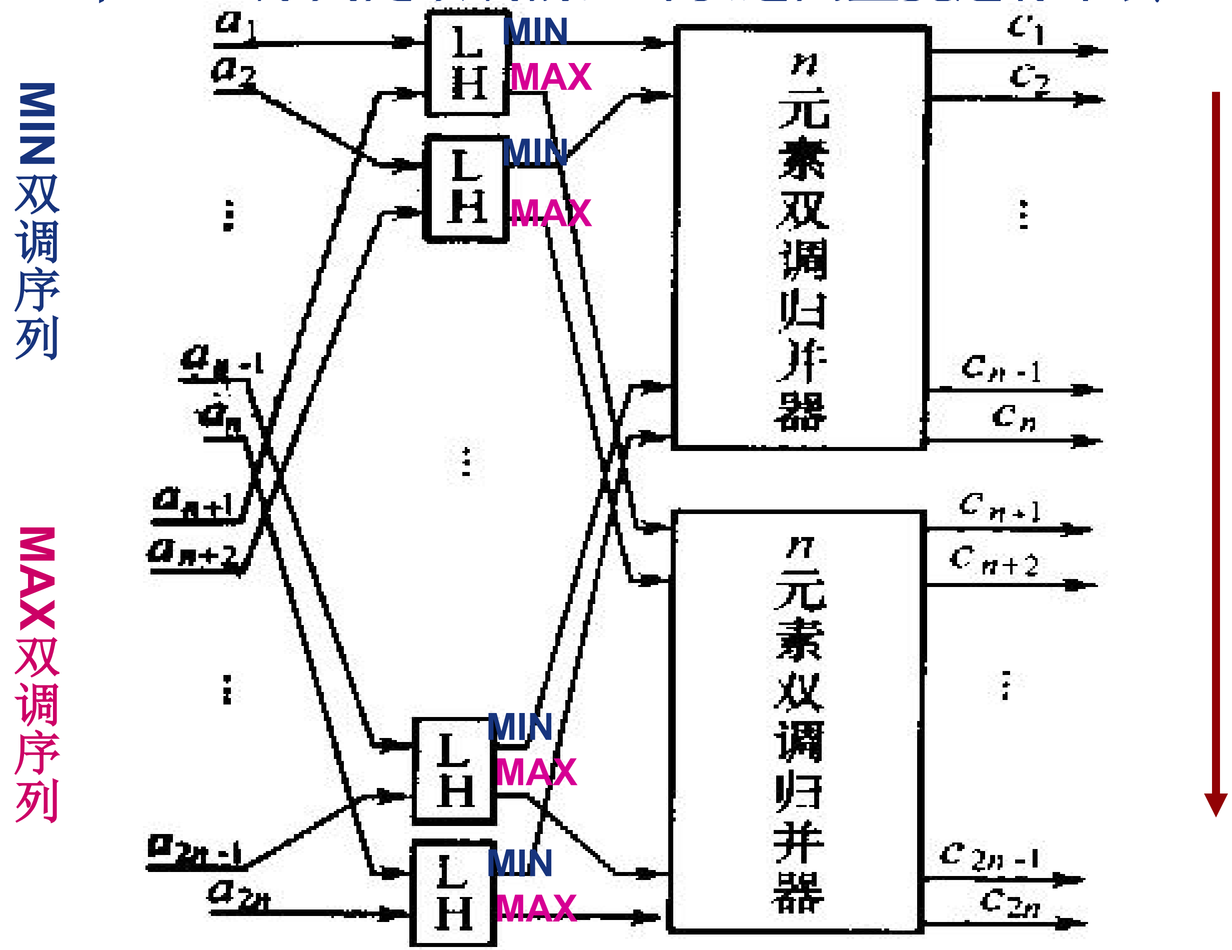
$c_i=\max\{a_i, a_{i+n}\}$ ==> $MAX=\{c_1,\cdots,c_n\}$, 则

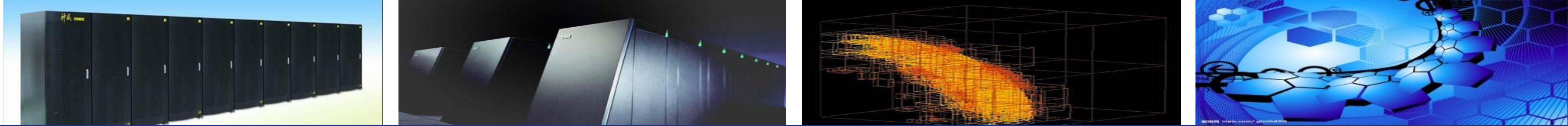
(1) $b_i \leqslant c_j$ $(1\leq i, j\leq n)$ (2) MIN和MAX序列仍是双调的



1 Batcher归并和排序—双调归并网络

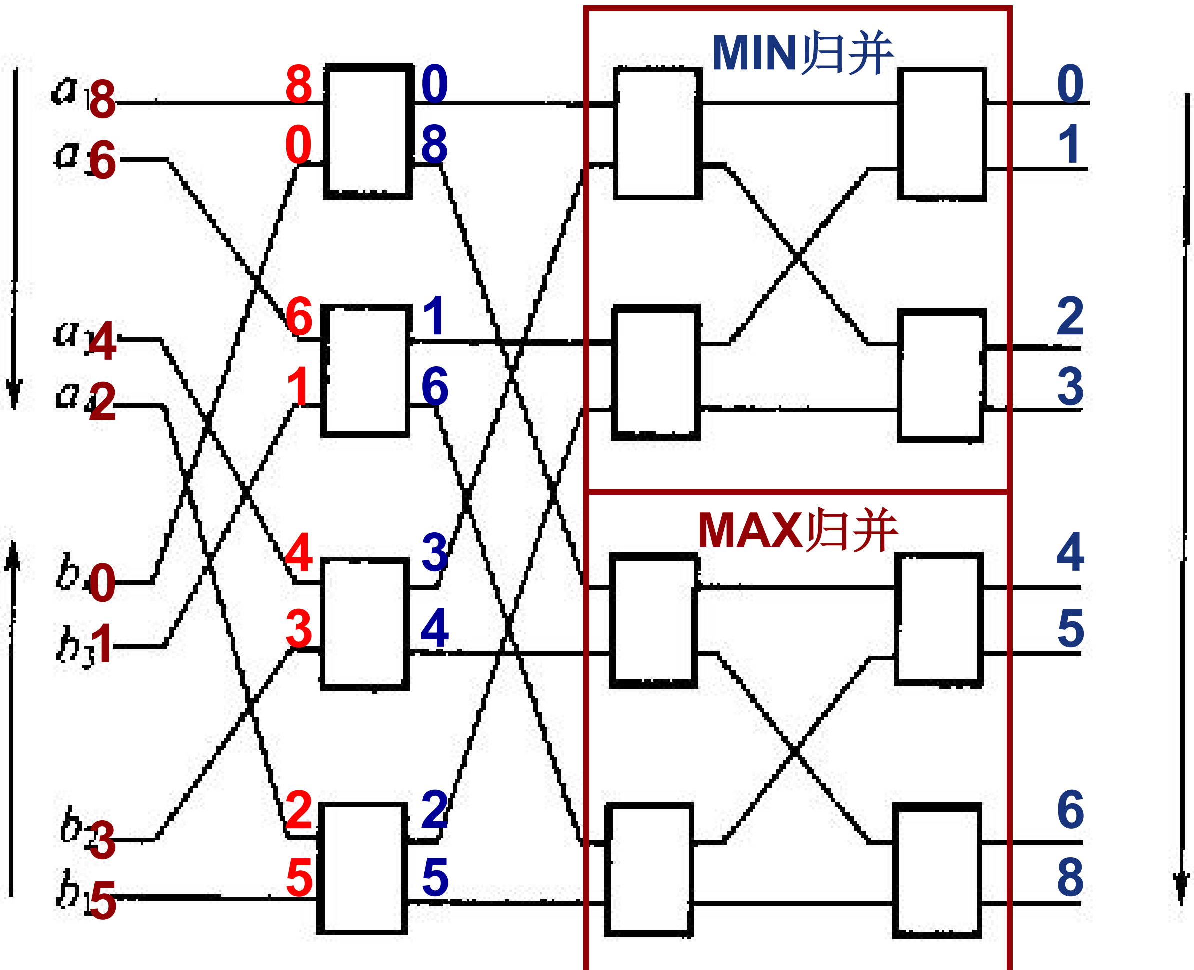
- 2. 网络构造(依据Batcher定理)
- 2n个输入的双调序列两两比较形成2个大小为n的MIN和MAX序列
- MIN和MAX序列是双调的，可以递归重复进行下去





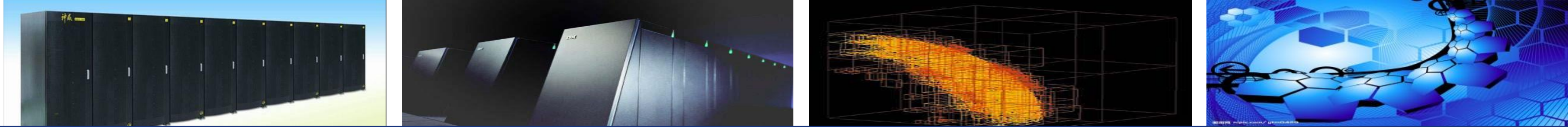
1 Batcher归并和排序－双调归并网络

3. 例:双调序列(8,6,4,2,0,1,3,5)的(4,4)双调归并网络



两两比较

2个(2,2)双调归并网络



1 Batcher归并和排序—双调归并网络

4. 复杂性分析

❖ 比较器数目

$$C_{BIT}^{M}(n) = C_{BIT}^{M}(\lceil n/2 \rceil) + C_{BIT}^{M}(\lfloor n/2 \rfloor) + \lfloor n/2 \rfloor, \quad n \geqslant 2$$

MIN比较器数　　MAX比较器数　　本级两两比较器数

当$n=2^t$时

$$C_{BIT}^{M}(2^t) = 2C_{BIT}^{M}(n/2) + n/2 = (n/2)\log n$$

1 Batcher归并和排序—双调归并网络

4. 复杂性分析

- 延迟级数

$$D_{BIT}^{M}(n)=\lceil \log n \rceil,\qquad n \geqslant 2$$

注：如何推导？

$$D_{BIT}^{M}(n)=\begin{cases} 0 & n=1 \\ 1+\max\left\{D_{BIT}^{M}(\lceil n/2 \rceil),\, D_{BIT}^{M}(\lfloor n/2 \rfloor)\right\} & n>1 \end{cases}$$

$$=1+D_{BIT}^{M}(\lceil n/2 \rceil)=1+(1+D_{BIT}^{M}(\lceil n/2^{2} \rceil))$$

$$=\sum_{i=1}^{\lceil \log n \rceil} 1+D_{BIT}^{M}\left(\left\lceil \frac{n}{2^{\lceil \log n \rceil}} \right\rceil\right)=\lceil \log n \rceil$$

1 Batcher归并和排序－ Batcher排序网络

1. 排序网络原理

(1)对输入数进行两两比较，形成长度为2的有序序列组；

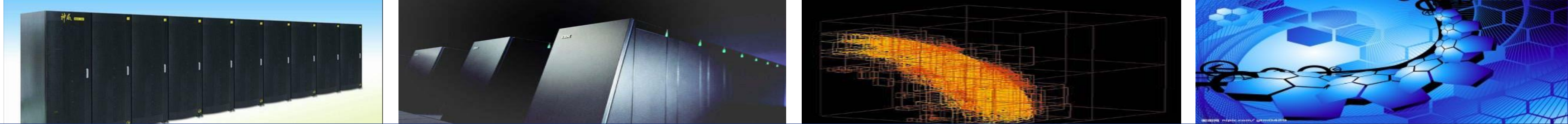
(2)对长度为2的有序序列组进行两两归并，形成长度为4的有序序列组；

(3)重复上述步骤，直至形成两个长度为n/2的有序序列；

(4)最后对长度为n/2的有序序列归并为一个完整的有序序列。

注：记n元输入的Batcher排序网络为B(n)

记(m,n)元输入的Batcher归并网络为B(m,n)



1 Batcher归并和排序— Batcher排序网络

2. 奇偶排序网络

- 基于奇偶归并网络
- 示例： **B(8)**

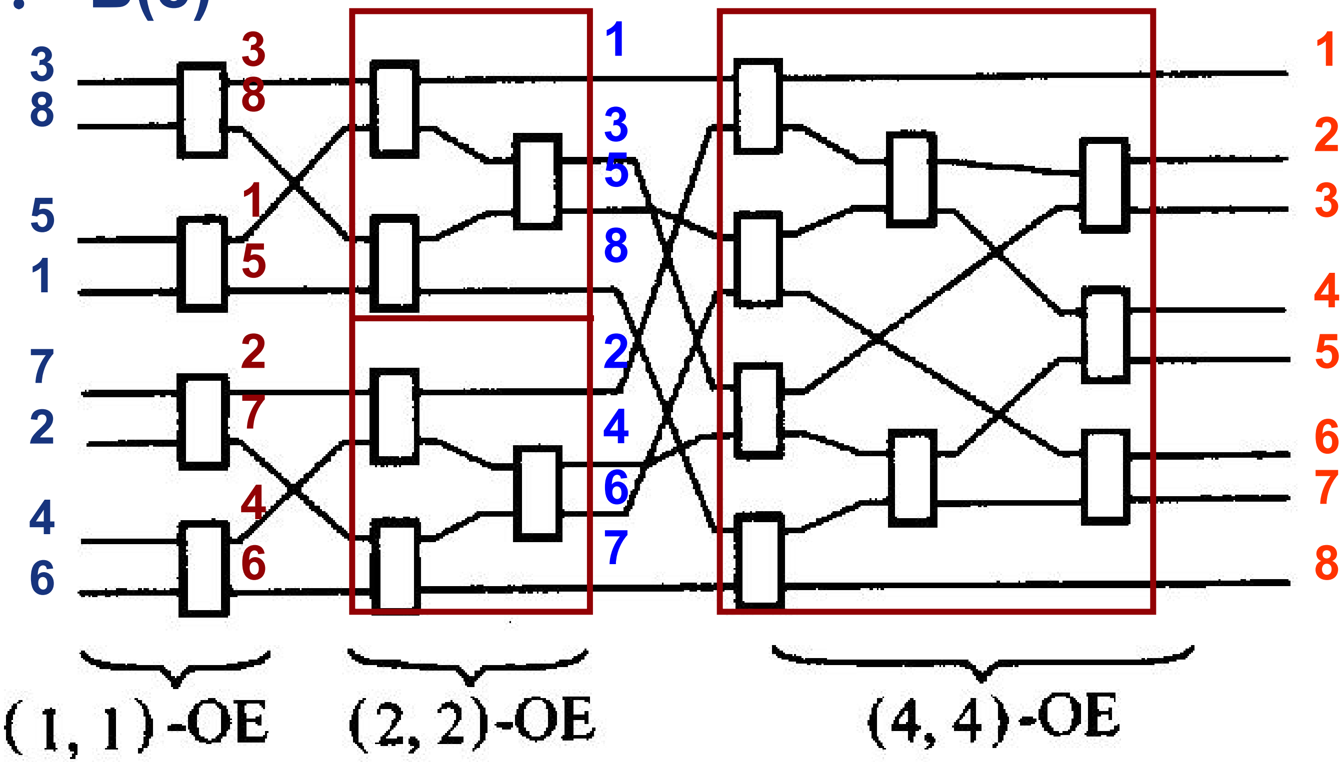
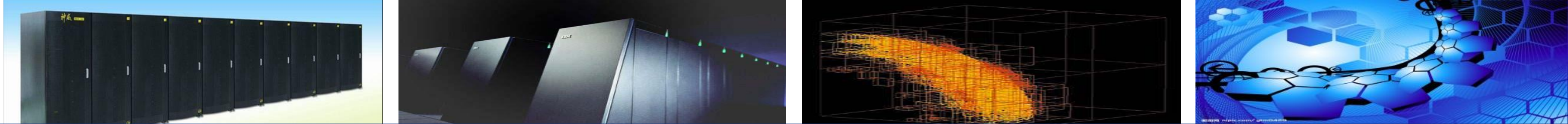


图 3.6　8 输入的奇偶排序网络



1 Batcher归并和排序— Batcher排序网络

3. 双调排序网络

- 基于双调归并网络
- 示例：**B(8)**

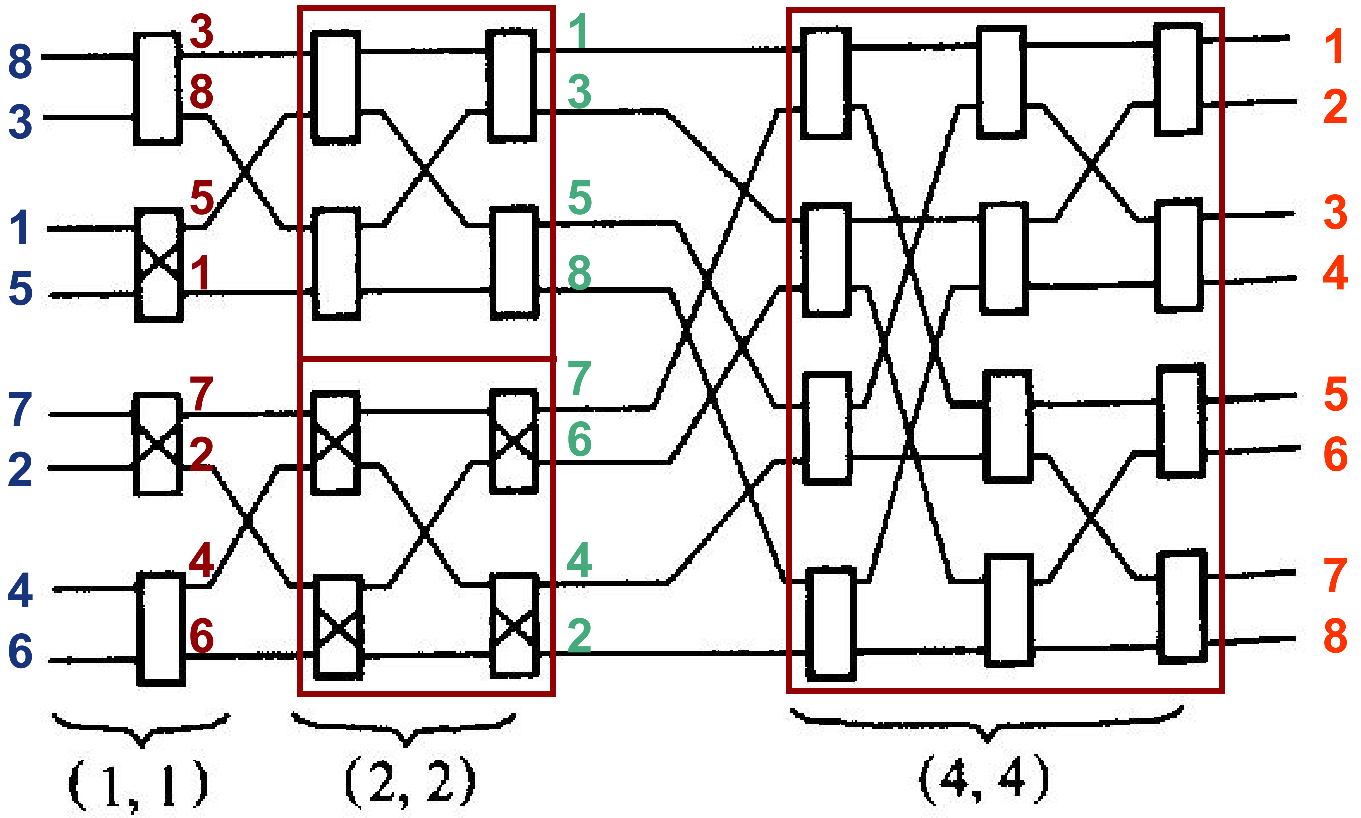
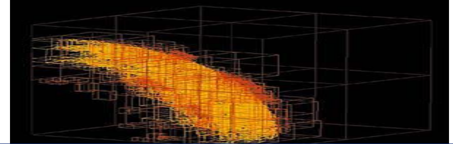


图 3.7　8 输入的双调排序网络






1 Batcher归并和排序— Batcher排序网络

4. 排序网络复杂性

- 奇偶排序网络
  - 比较器数目 $C_{OE}^{S}(n) = (n/4)(\log^2 n - \log n + 4) - 1$
  - 延迟级数 $D_{OE}^{S}(n) = \lceil \log n \rceil (1 + \lceil \log n \rceil)/2, \quad n \geq 2$
- 双调排序网络
  - 比较器数目 $C_{BIT}^{S}(n) = (n/4)(\log^2 n + \log n)$
  - 延迟级数 $D_{BIT}^{S}(n) = \lceil \log n \rceil (1 + \lceil \log n \rceil)/2, \quad n \geq 2$

2 (m, n)-选择网络 —分组选择网络

1. 基于划分原理的(m,n)-选择过程

①将n个输入数据划分成若干个大小相等的子序列(≥m)；

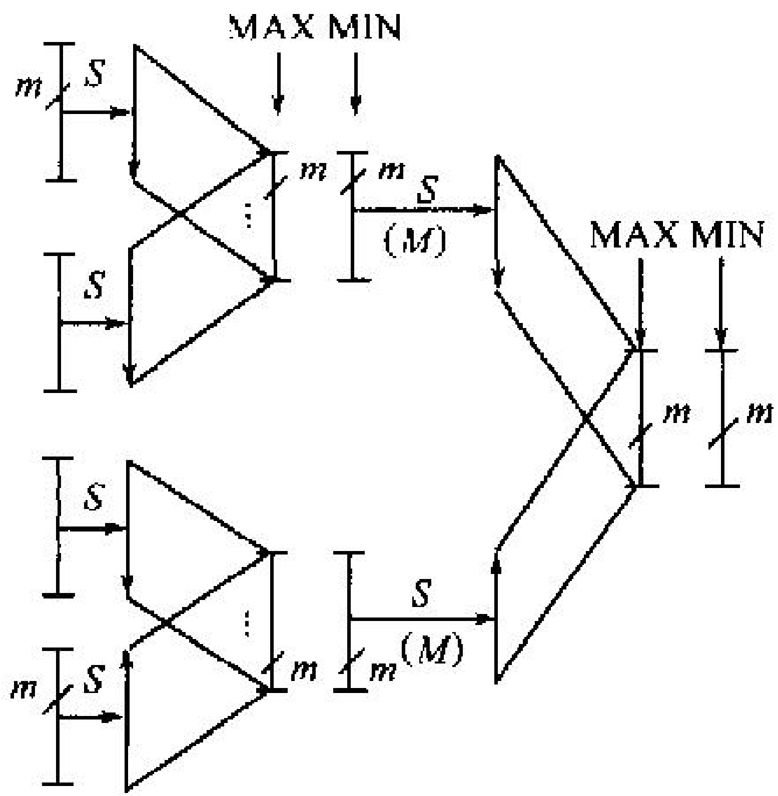
②使用Batcher排序网络对各子序列排序；

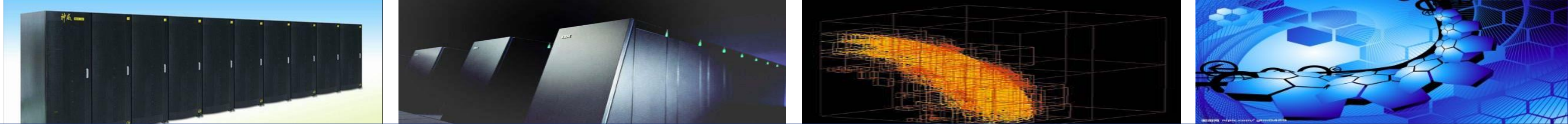
③将有序子序列形成双调序列，进行两两对接；

使用Batcher定理形成MAX,MIN序列，弃去MAX序列；

再使用Batcher排序网络将MIN序列排成有序序列；

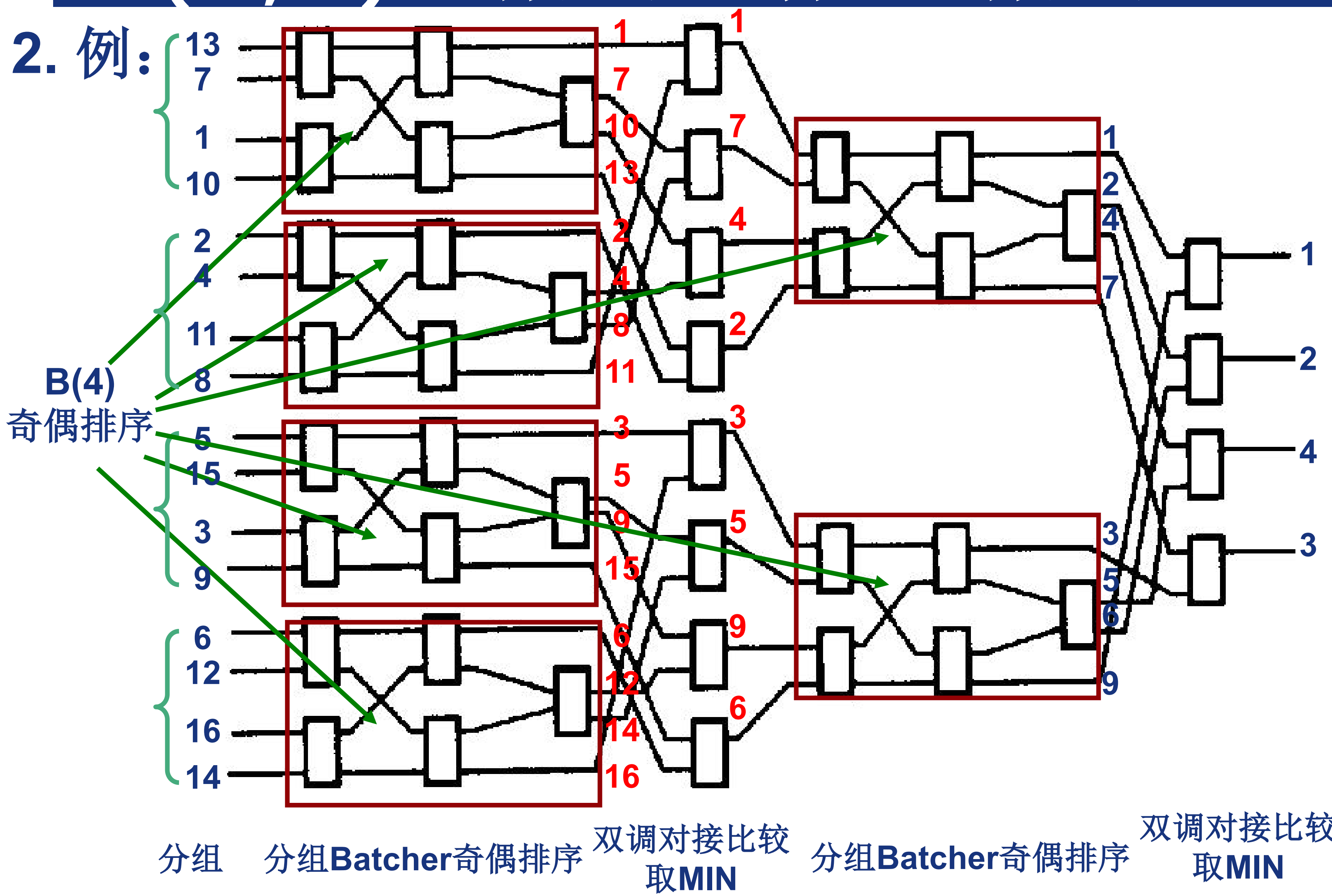
④重复③直至MIN序列恰好包含所需的m个最小元素为止。

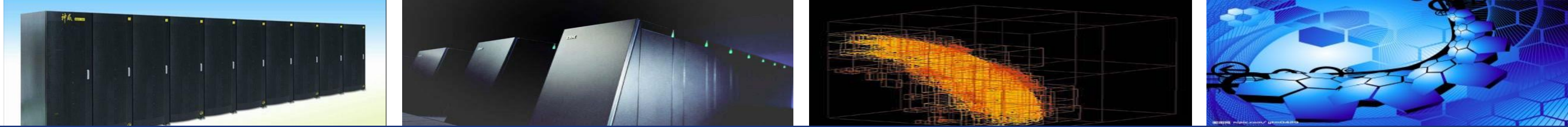




2 (m, n)-选择网络 —分组选择网络

2. 例：





2 (m, n)-选择网络 —分组选择网络

3. 正确性定理

P129定理4.4

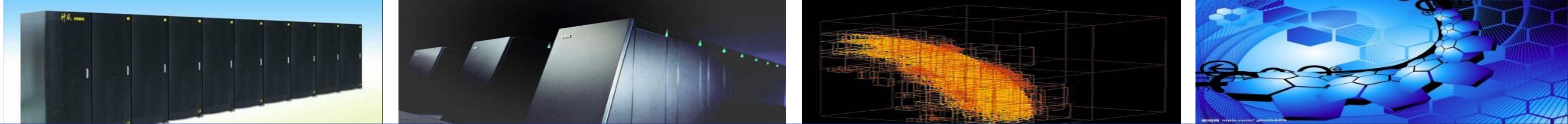
4. 复杂性分析

- 比较器数目

$$C_A^p(m,n) \leqslant (n-m)\left((1/2)\log^2 m-(1/2)\log m+3-2/m\right)$$

- 延迟级数

$$D_A^p(m,n)=\log\frac{n}{m}\left(1+D_{OE}^S(m)\right)=\log\frac{n}{m}\left(1+\frac{1}{2}\log^2 m+\frac{1}{2}\log m\right)$$



2 (m, n)-选择网络 —平衡分组选择网络

1. 平衡分组选择过程

①将n个输入数据划分成若干个大小相等的子序列;

②使用Batcher排序网络对各子序列排序;

③将有序子序列形成双调序列，进行两两对接;

使用Batcher定理形成MAX,MIN序列，弃去MAX序列;

④对MIN序列进行双调归并形成有序序列;

将有序子序列形成双调序列，进行两两对接;

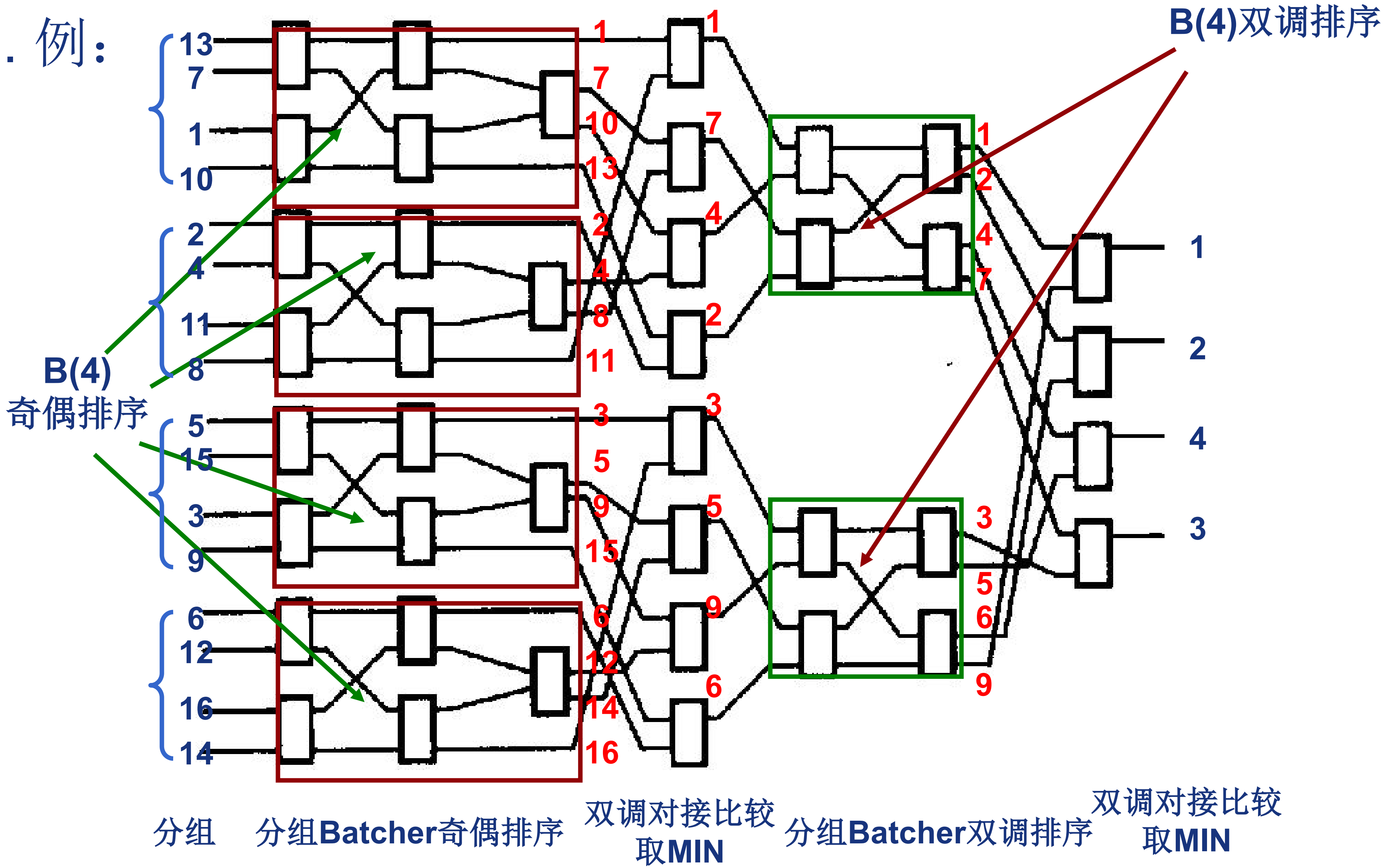
重复，直至恰好包含所需的m个最小元素为止。

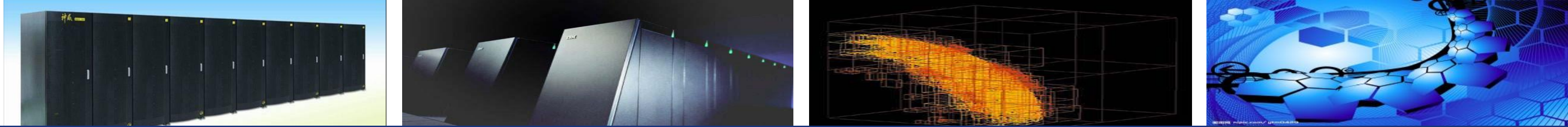
注: (1)用双调排序网络取代奇偶排序网络(第1次除外)

(2)减少了比较器的级数

2 (m, n)-选择网络 —平衡分组选择网络

2. 例：





2 (m, n)-选择网络 —平衡分组选择网络

3. 复杂性分析

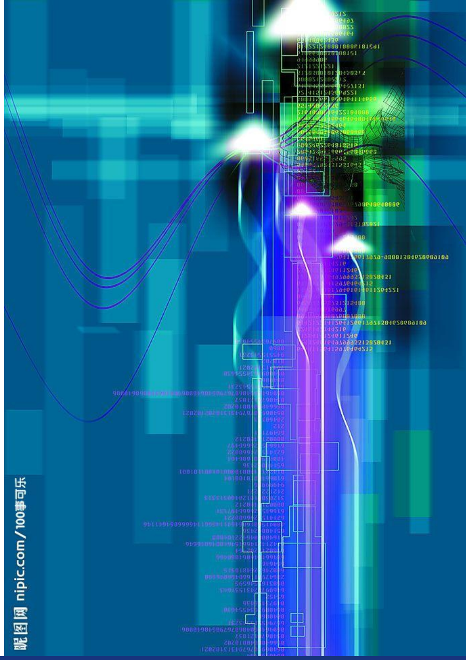
❖ 比较器数目

$$C_C^p(m,n) = (n/4)\log^2 m + (n/4 - m)\log m + 2n - (n/m + m)$$

❖ 延迟级数

$$D_C^p(m,n) = \log\frac{n}{m}(1 + \log m) + \frac{1}{2}\log m(\log m - 1)$$

注：平衡分组选择网络比分组选择网络快了**O(logm)**



The End

Thank You !