به نام خدا

پردازش موازی جلسه اول: معرفی درس

مقدمه

مرتب‌سازی آرایه خطی

(با فرض اینکه داده‌ها درون پردازنده‌ها باشند و از بیرون وارد نشوند.)

نوبت‌های زوج و فرد را جدا می‌کنیم. در نوبت‌های فرد پردازنده‌های فرد min می‌گیرند و در نوبت‌های زوج، پردازنده‌های زوج.

ce.sharif.edu/~ghodsi/PP/Leighton%20figures/Chapter%201-all.pdf

78 / 118

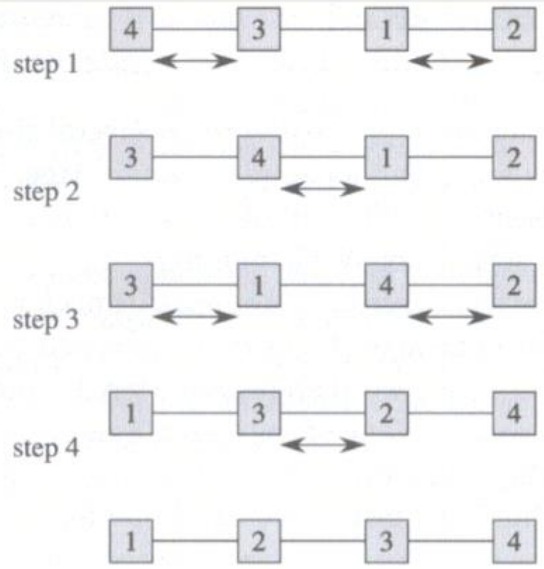


Figure 1-78 *Sorting N numbers on an N-cell linear array using odd-even transposition sort. The values in cell i and $i+1$ are compared and possibly exchanged during steps t for which $i+t$ is even ($1 \le i < N$ and $1 \le t \le N$). Smaller values are moved leftward during each exchange.*

ghodsi/PP/Leighton%20figures/Chapter%201-all.pdf

79 / 118

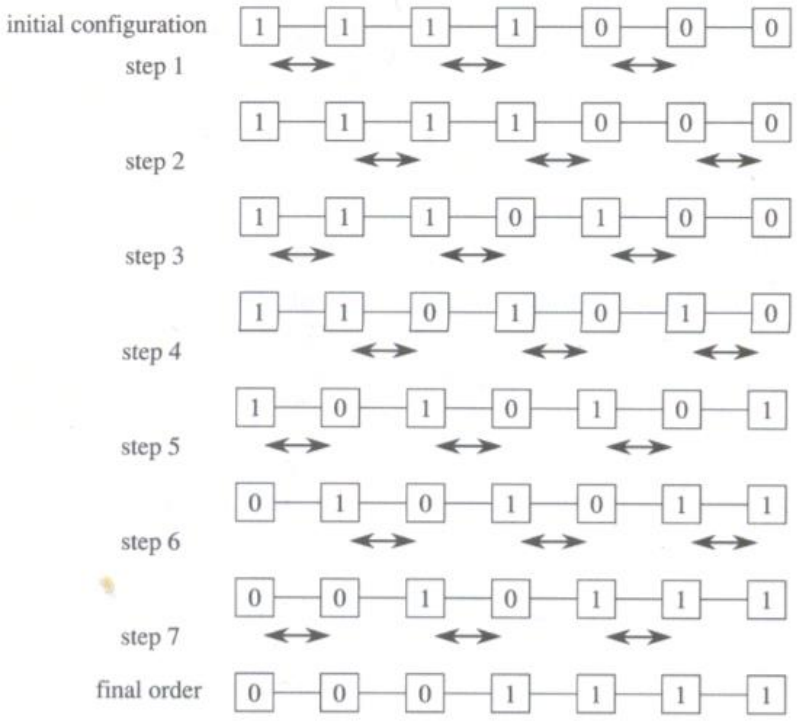


Figure 1-79 *The action of odd-even transposition sort on a set of 0s and 1s in an N-cell linear array for N = 7. The rightward movement of the ith rightmost 1 is never blocked after step i until the ith rightmost 1 reaches cell N − i + 1. Hence, the 0s and 1s are sorted within N steps.*

طراحی الگوریتم موازی

طراحی یک الگوریتم موازی معادل با طراحی IC سخت‌افزاری معادل آن است. در الگوریتم موازی روش کار مثل این است که یک کلاک داریم و با هر کلاک پردازنده‌ها یک عمل را انجام می‌دهند. به چنین مداری، مدار سیستولیک (systolic) می‌گویند.

چند مرحله لازم است تا مرتب‌سازی مثال قبل انجام شود؟

جواب: n-1 مرحله

چرا این الگوریتم درست کار می‌کند؟

اثبات: بزرگترین عدد هر بار یک قدم به سمت راست حرکت می‌کند، دورترین جایی که می‌تواند باشد سمت چپ‌ترین مکان است.

مرتب‌سازی توری n*n: می‌خواهیم مارپیچی مرتب شده باشد.

hton%20figures/Chapter%201-all.pdf

80 / 118

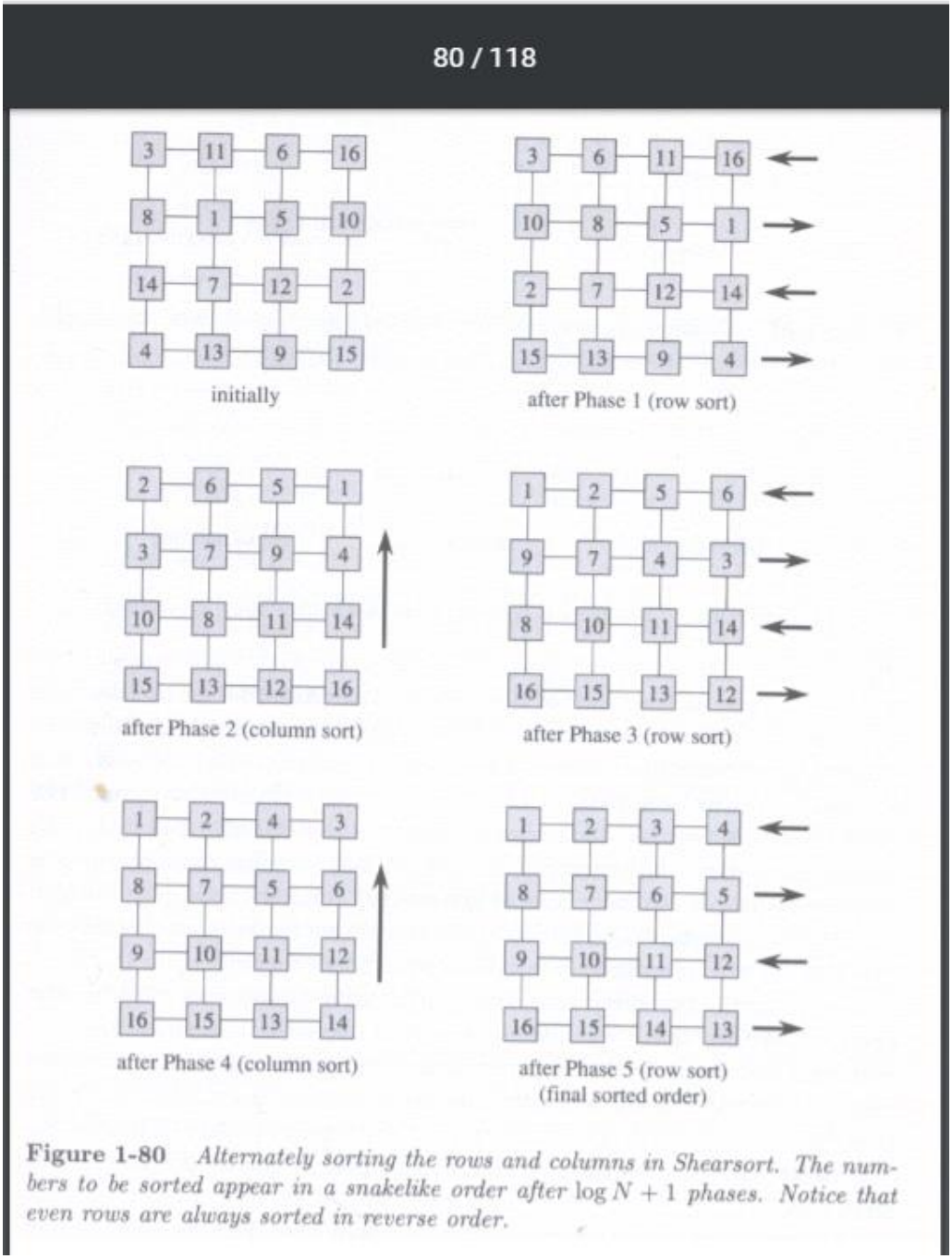


Figure 1-80 *Alternately sorting the rows and columns in Shearsort. The numbers to be sorted appear in a snakelike order after* $\log N + 1$ *phases. Notice that even rows are always sorted in reverse order.*

با تکرار 3n مرحله می‌توانیم این کار را انجام بدهیم. زمان n log n بهترین زمانی است که می‌توانیم هر سطر/ستون را مرتب کنیم.

بدون موازی‌سازی (در حالت ترتیبی)، زمان مرتب‌سازی n^2 عدد $O(n^2 \log n)$ می‌شود.

پس الگوریتم موازی سرعت را چند برابر می‌کند:

$$Speedup: S = \frac{n^2 \log n}{n \log n} = n$$

صورت کسر زمان اجرای سریال و مخرج آن زمان اجرای موازی است.

با توجه به اینکه از n^2 پردازنده استفاده کرده‌ایم، از لحاظ سخت‌افزاری چقدر بهینه است؟

جمع دو عدد بیتی

- Ripple adder زمان O(n)

زمانی که صرف carry می‌شود باعث کند شدن الگوریتم می‌شود.

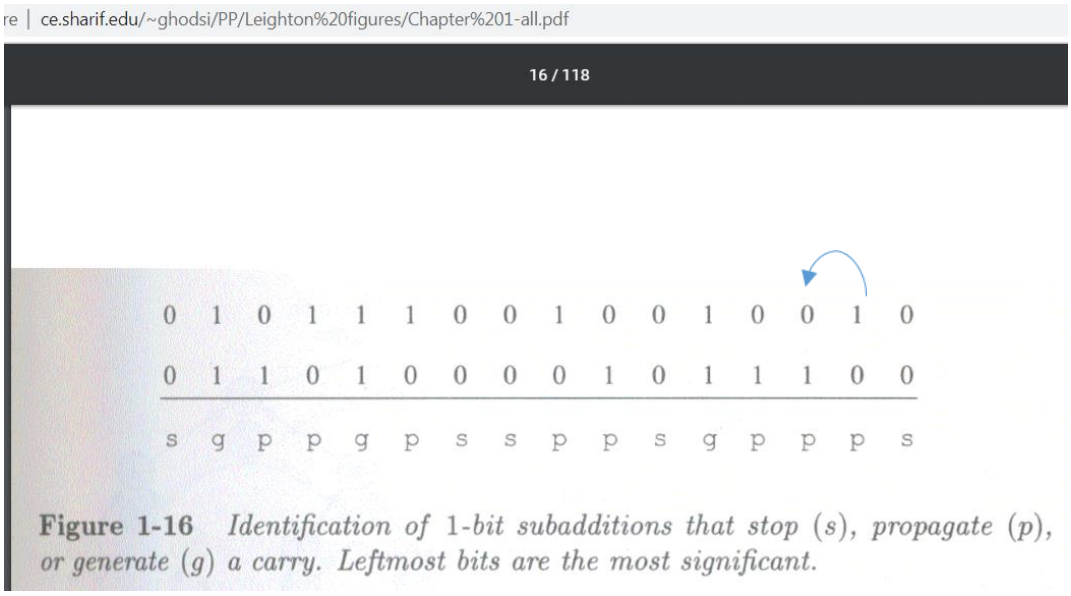
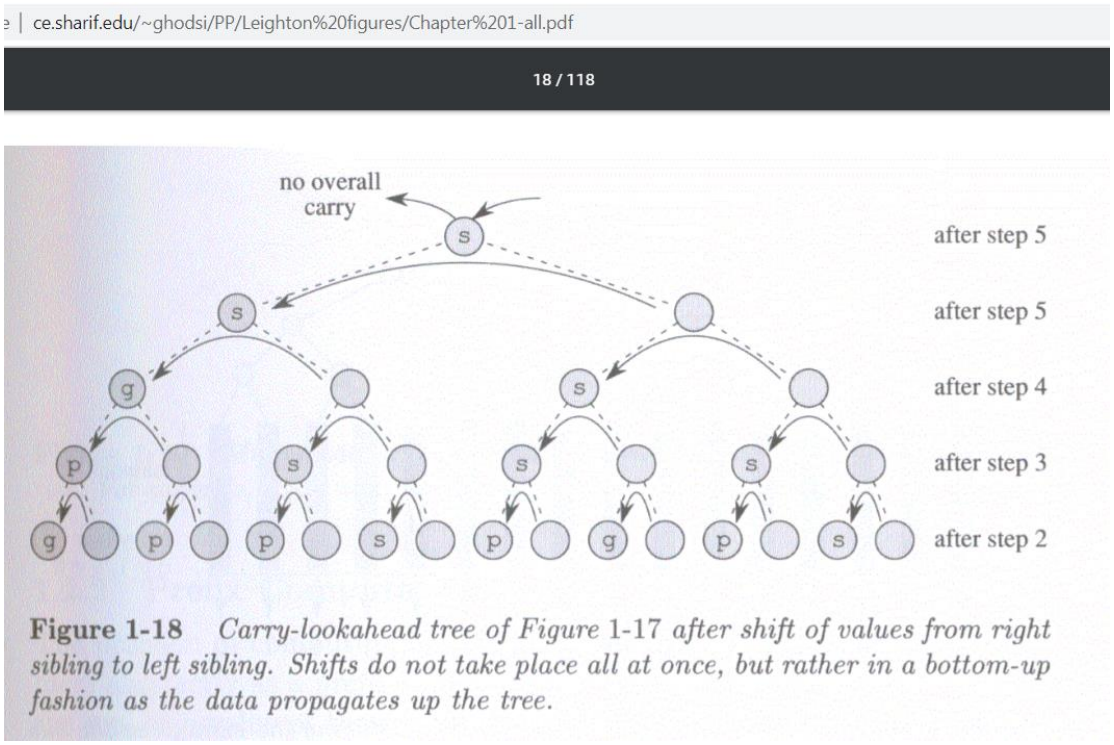


Figure 1-16 *Identification of 1-bit subadditions that stop (s), propagate (p), or generate (g) a carry. Leftmost bits are the most significant.*

(بدترین حالت وقتی است که در هر مرحله carry داشته باشیم).

- Carry-lookahead Adder

یک ساختار درختی داریم که اعداد در برگ‌ها هستند. کاری می‌کنیم که در زمان log n هر carry به مقصد برسد.



چون عملیات پردازنده‌ها ساده است، کلاک آنها را می‌توانیم خیلی سریع کنیم.

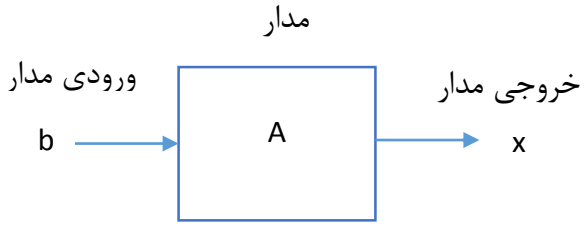
حل معادلات خطی

ماتریس A و بردار b داده شده است. می‌خواهیم بردار x را به دست بیاوریم.

Ax=b

با فرض n-بعدی بودن یعنی A_{n*n} و b_{1*n} و x_{1*n}

الگوریتم معمولی $O(n^3)$ زمان می‌خواهد.



الگوریتم موازی 3n زمان می‌خواهد که ثابت می‌شود بهینه هم هست.

$$S = \frac{O(n^3)}{\theta(n)} = O(n^2)$$

که بهبود سرعت خوبی است.

مثالهای دیگری از الگوریتم‌های موازی این درس: به دست آوردن مولفه‌های همبندی گراف، MST، ...

در طراحی الگوریتم‌های موازی مهم این است که پردازنده‌ها را چطور به هم وصل کنید.

مسئله‌ی Routing

n پردازنده داریم، به صورت توری به هم وصل هستند. هر پردازنده می‌خواهد یک بسته به مقصد پردازنده‌ی دیگری بفرستد.

ton%20figures/Chapter%201-all.pdf

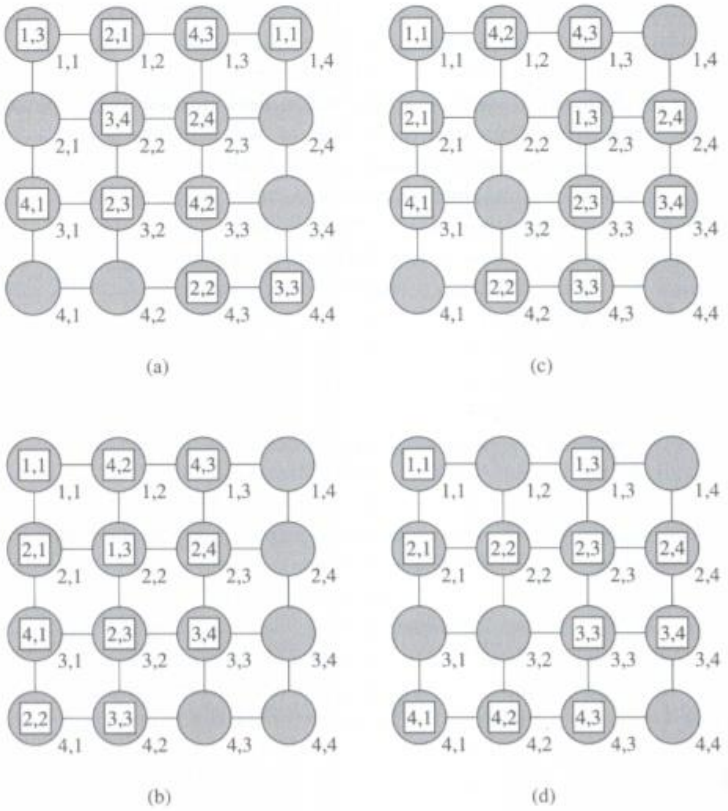


Figure 1-94 *A sorting-based algorithm for packet routing that runs in $O(\sqrt{N})$ steps with queues of size* 1. *(a) Location of packets initially. (b) Location of packets after sorting packets in column-major order based on column destination. (c) Location of packets after routing to the correct column. (d) Final location of packets.*