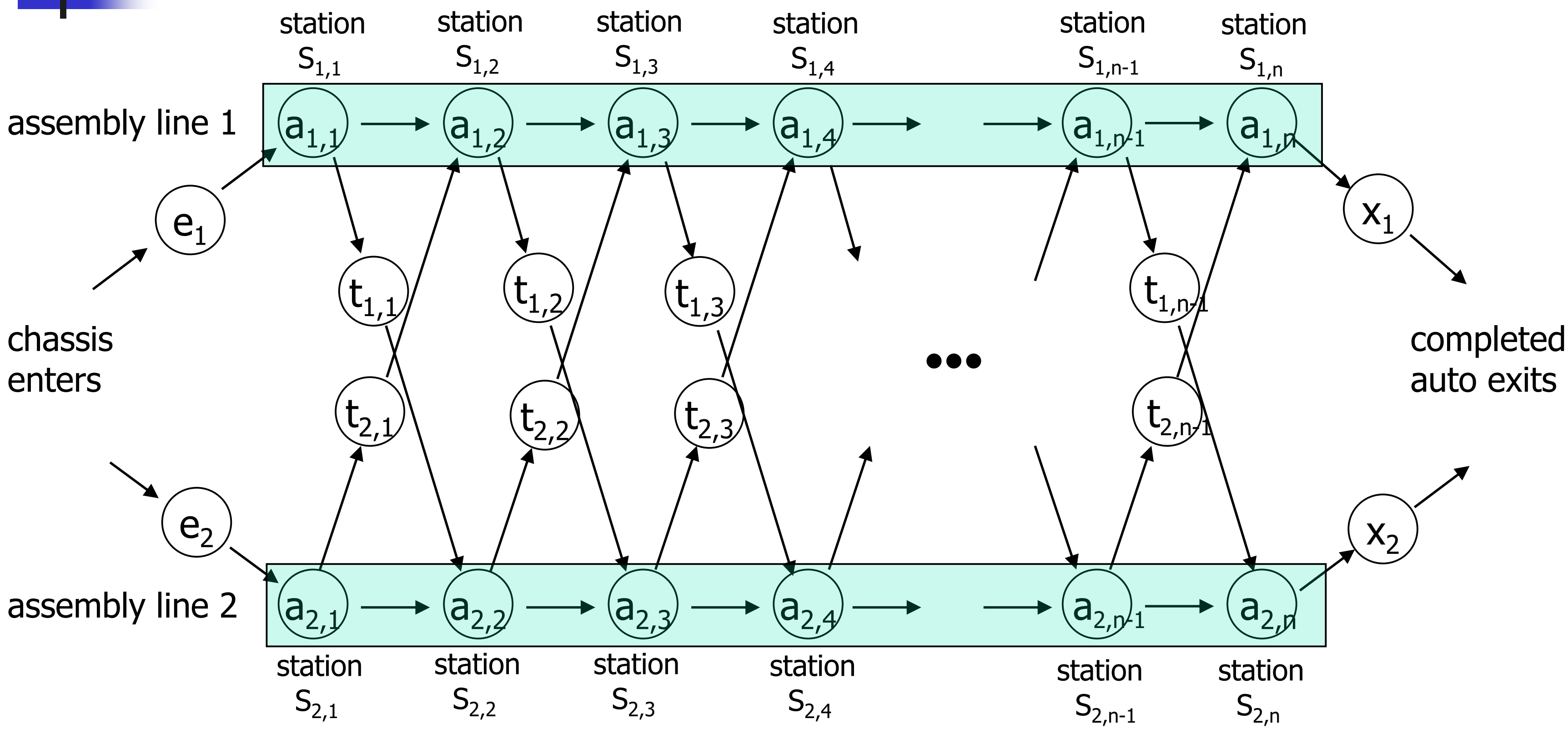
Assembly-line Scheduling

Assembly-line



Assembly-line

- $S_{i,j}$: station j on line i
- $a_{i,j}$: the assembly time required at station $S_{i,j}$
- $t_{i,j}$: the time to transfer a chassis from assembly line i, after having gone through station $S_{i,j}$
- e_i, x_i : entry time on line i, exit time on line i

Problem Definition

- There are two assembly lines each with n stations
- The j-th station on line 1, 2 performs the same function
- Determine which stations to choose in order to minimize the total time through the factory

Brute Force Way

- There are 2^n possible ways to choose stations
- So brute force way take $\Omega(2^n)$ time complexity
- It is infeasible when n is large

The Structure of the Fastest Way through the Factory (1)

- The fastest way through station $S_{1,j}$ is either
  - the fastest way through station $S_{1,j-1}$ and then directly through station $S_{1,j}$ or
  - the fastest way through station $S_{2,j-1}$, a transfer from line 2 to line 1, and then through station $S_{1,j}$

The Structure of the Fastest Way through the Factory (2)

- The fastest way through station $S_{2,j}$ is either
  - the fastest way through station $S_{2,j-1}$ and then directly through station $S_{2,j}$ or
  - the fastest way through station $S_{1,j-1}$, a transfer from line 1 to line 2, and then through station $S_{2,j}$

The Structure of the Fastest Way through the Factory (3)

- Solving the problem of finding the fastest way through station j of either line
  - need to solve the sub problems of finding the fastest ways through station j-1 on both lines

Recursive Solution

- $f_i[j]$: the fastest time to get a chassis from starting point through station $S_{i,j}$

$$f_1[j] = \begin{cases} e_1 + a_{1,1} & \text{if } j = 1 \\ \min(f_1[j-1] + a_{1,j},\ f_2[j-1] + t_{2,j-1} + a_{1,j}) & \text{if } j \ge 2 \end{cases}$$

$$f_2[j] = \begin{cases} e_2 + a_{2,1} & \text{if } j = 1 \\ \min(f_2[j-1] + a_{2,j},\ f_1[j-1] + t_{1,j-1} + a_{2,j}) & \text{if } j \ge 2 \end{cases}$$

- The fastest way through the entire factory

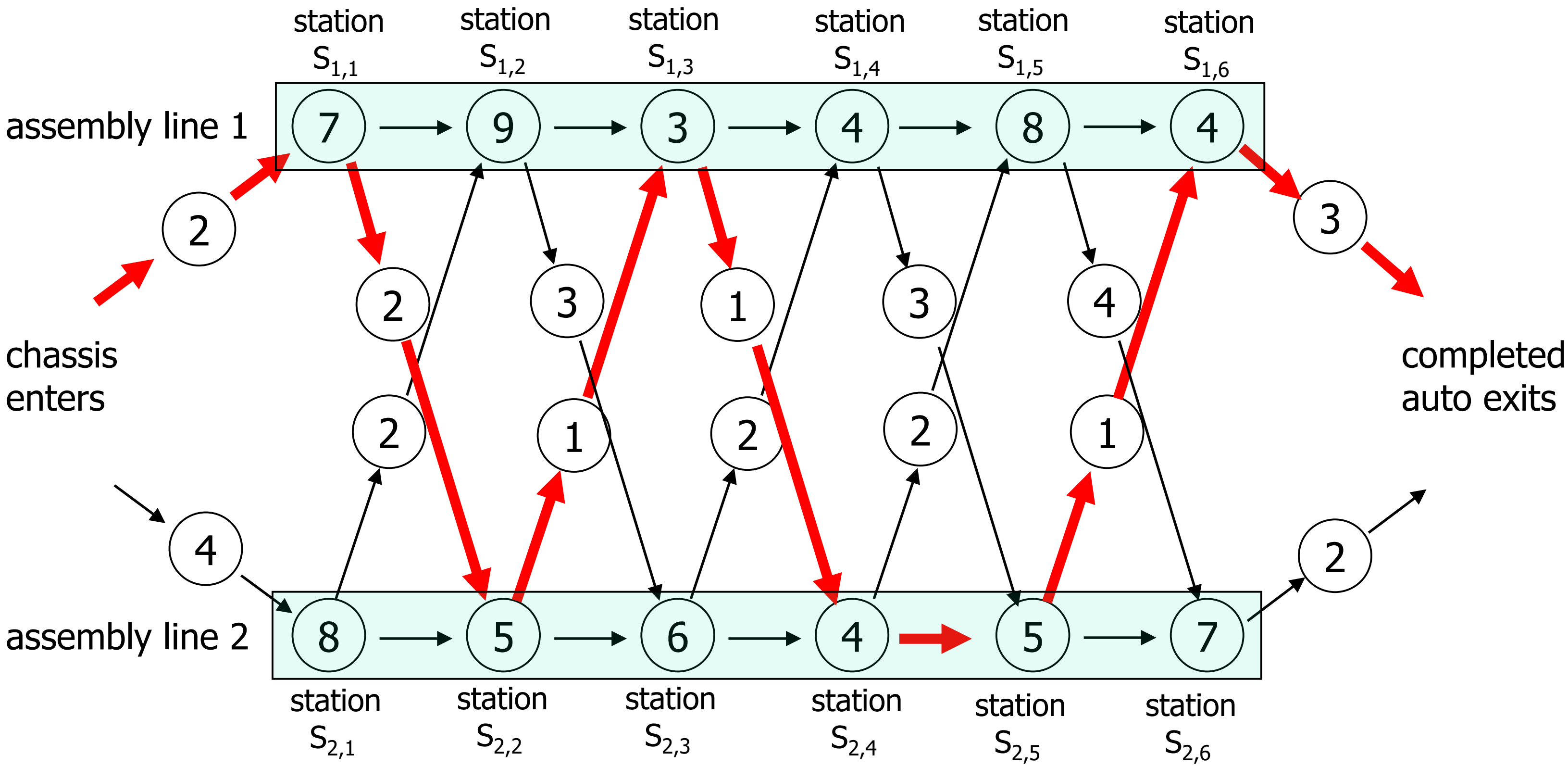
$$f^* = \min(f_1[n] + x_1,\ f_2[n] + x_2)$$

Computing the Fastest Times (1)

- We can write a recursive algorithm based on previous recurrence relation
- Let $r_i(j)$ be # of references made to $f_i[j]$ in a recursive algorithm
  - $r_1(n)=r_2(n)=1$
  - $r_1(j)=r_2(j)=r_1(j+1)+r_2(j+1)$ for $j=1,\ldots,n-1$
  - $r_i(j)=2^{n-j}$ (Exercise 15.1-3)
  - $f_1[1]$ alone is referenced 2^{n-1} times
- Thus, the running time is exponential in n

Computing the Fastest Times (2)

- By computing the $f_i[j]$ values in order of increasing station numbers j
  - Can compute the fastest way through the factory in $\Theta(n)$ time



j	1	2	3	4	5	6
$f_1[j]$	9	18	20	24	32	35
$f_2[j]$	12	16	22	25	30	37

j	2	3	4	5	6
$l_1[j]$	1	2	1	1	2
$l_2[j]$	1	2	1	2	2

$f^* = 38$

$l^*=1$

FASTEST-WAY(a,t,e,x,n)

$f_1[1] \leftarrow e_1+a_{1,1}$
$f_2[1] \leftarrow e_2+a_{2,1}$
for $j \leftarrow 2$ to n
    do if $f_1[j-1]+a_{1,j} \leq f_2[j-1]+t_{2,j-1}+a_{1,j}$
        then $f_1[j] \leftarrow f_1[j-1]+a_{1,j}$
            $l_1[j] \leftarrow 1$
        else $f_1[j] \leftarrow f_2[j-1]+t_{2,j-1}+a_{1,j}$
            $l_1[j] \leftarrow 2$
    if $f_2[j-1]+a_{2,j} \leq f_1[j-1]+t_{1,j-1}+a_{2,j}$
        then $f_2[j] \leftarrow f_2[j-1]+a_{2,j}$
            $l_2[j] \leftarrow 2$
        else $f_2[j] \leftarrow f_1[j-1]+t_{1,j-1}+a_{2,j}$
            $l_2[j] \leftarrow 1$
if $f_1[n]+x_1 \leq f_2[n]+x_2$
    then $f^* = f_1[n]+x_1$
        $l^* = 1$
    else $f^* = f_2[n]+x_2$
        $l^* = 2$