ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

TỐI ƯU LẬP KẾ HOẠCH

Bài toán lập lịch

Nội dung

- Mô hình bài toán lập lịch
- Mô hình MIP
- Phương pháp tìm kiếm cục bộ

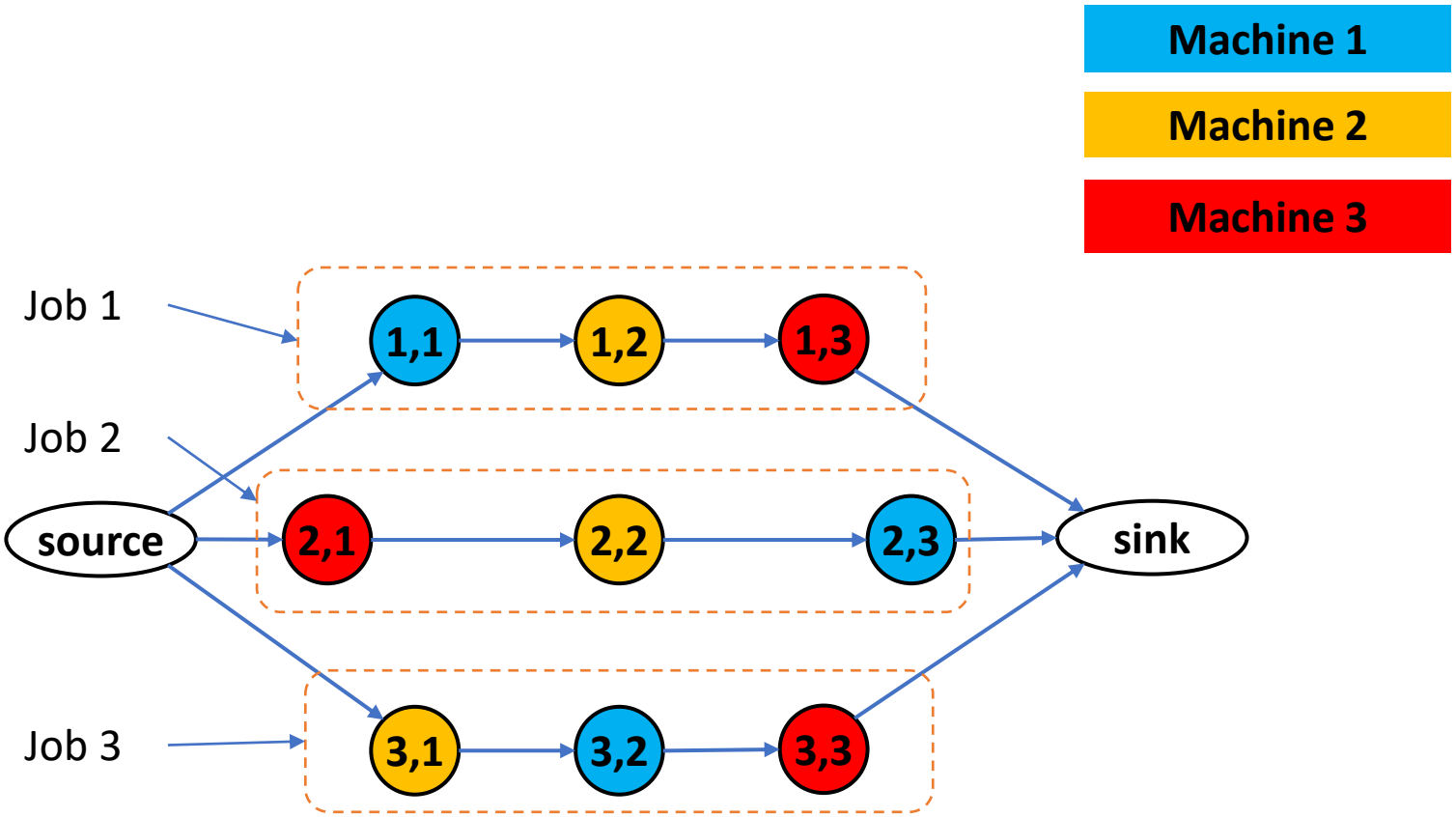
Bài toán lập lịch - JobShop Scheduling

- Mô tả bài toán
  - Một tập hữu hạn *J* gồm *n* jobs 1, 2, …, *n* cần được xử lý trên một tập hữu hạn *M* gồm *m* machines 1, 2, …, *m*
  - Mỗi job *j* bao gồm 1 chuỗi các task *t(j,1)*, . . ., *t*(*j*, λ_j), mỗi task được thực hiện trên 1 machine khác nhau và theo thứ tự đặt ra
  - Mỗi task *t*(*j*, *i*) bao gồm 2 thông tin
    - *r*(*j*, *i*): chỉ số của machine mà task *t*(*j*, *i*) được thực hiện
    - *d*(*j*, *i*): khoảng thời gian task *t*(*j*, *i*) thực hiện trên machine *r*(*j*, *i*)
  - Cần lập lịch thực hiện các tasks của các jobs trên các machines sao cho thời gian hoàn thành là ngắn nhất

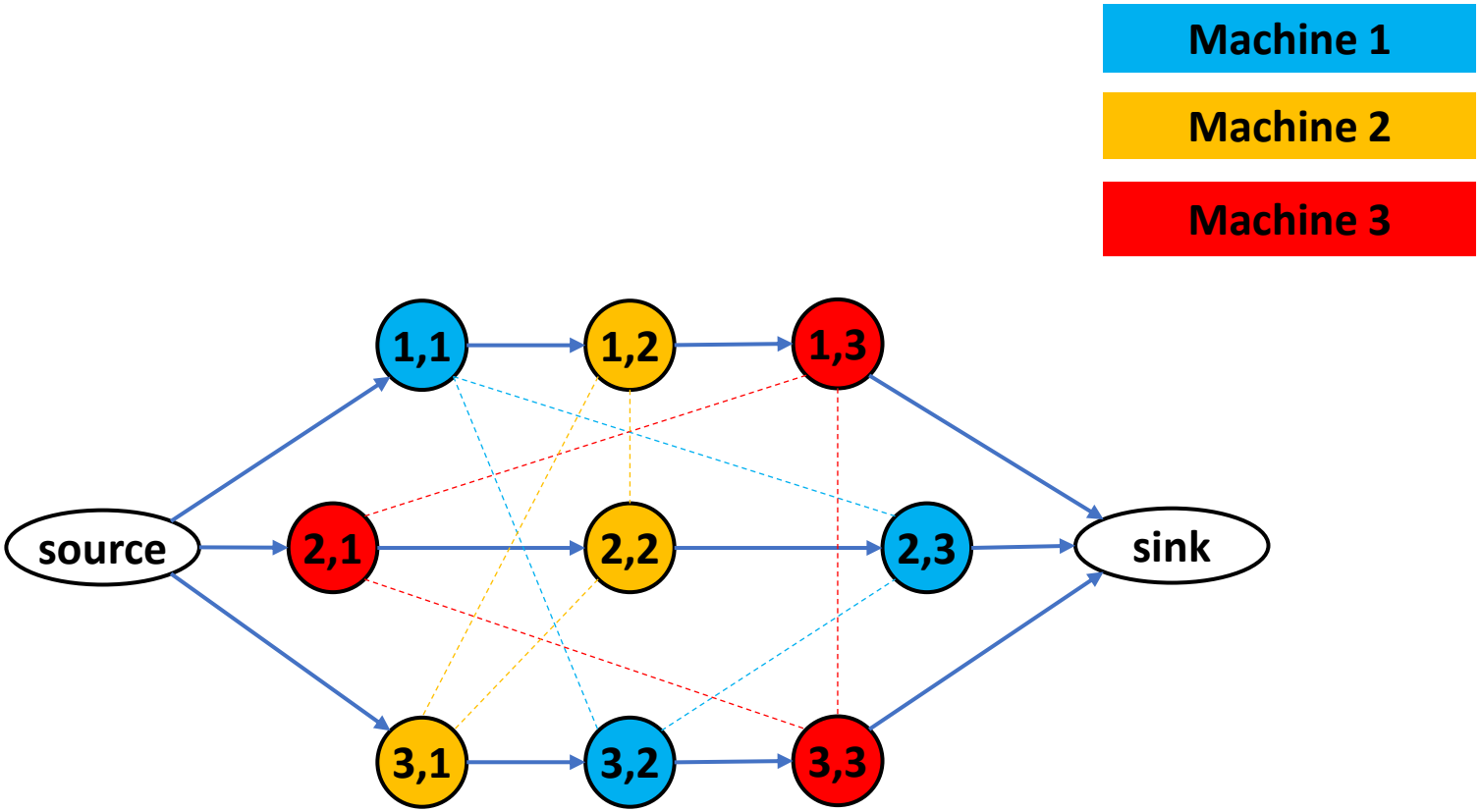
Bài toán lập lịch - JobShop Scheduling

- Ràng buộc
  - Với mỗi job, một task chỉ được bắt đầu thực hiện khi task trước đó đã hoàn thành
  - Mỗi machine chỉ có thể thực hiện duy nhất 1 task tại mỗi thời điểm
  - Mỗi task khi được thực hiện trên 1 machine nào đó thì nó thực hiện liên tục cho đến khi hoàn thành

Bài toán lập lịch - JobShop Scheduling

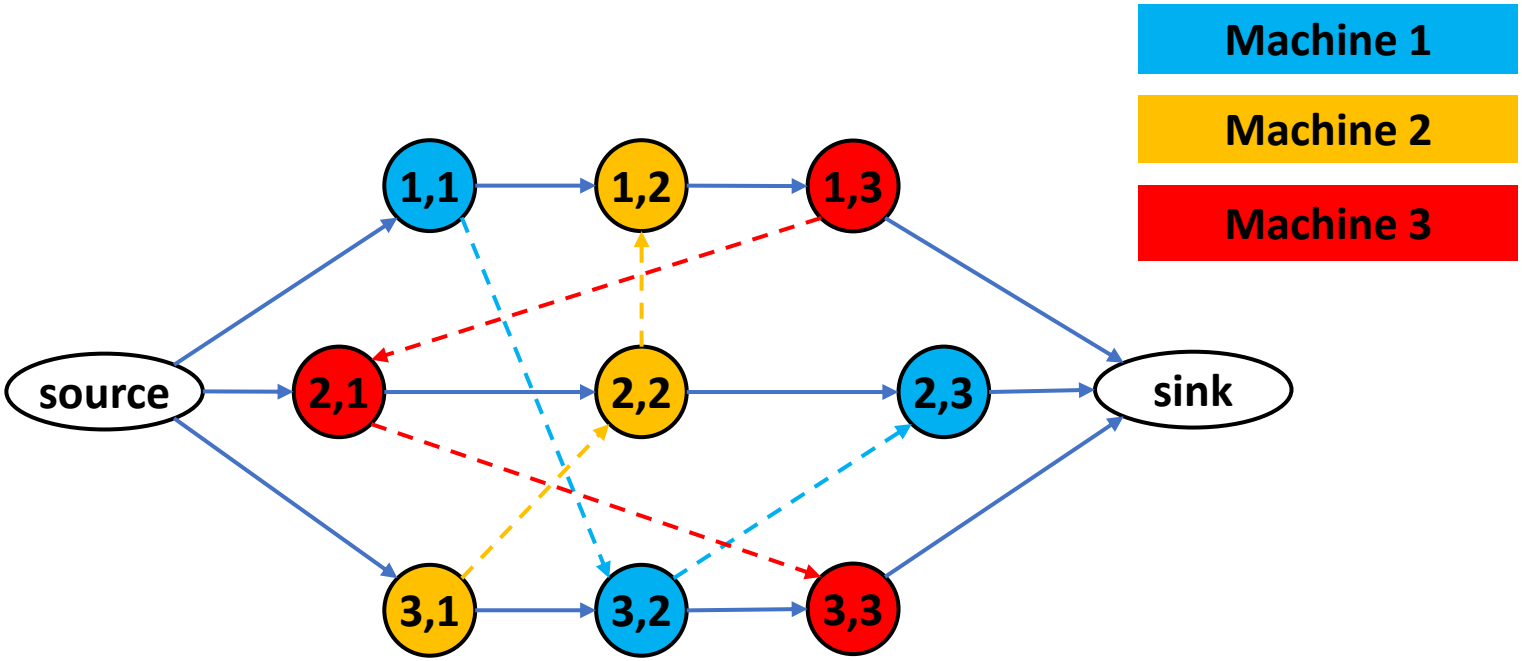


Bài toán lập lịch - JobShop Scheduling



Bài toán lập lịch - JobShop Scheduling

- Thời gian hoàn thành tất cả các job được xác định thông qua đường đi dài nhất từ source đến sink trên đồ thị có hướng và không chu trình

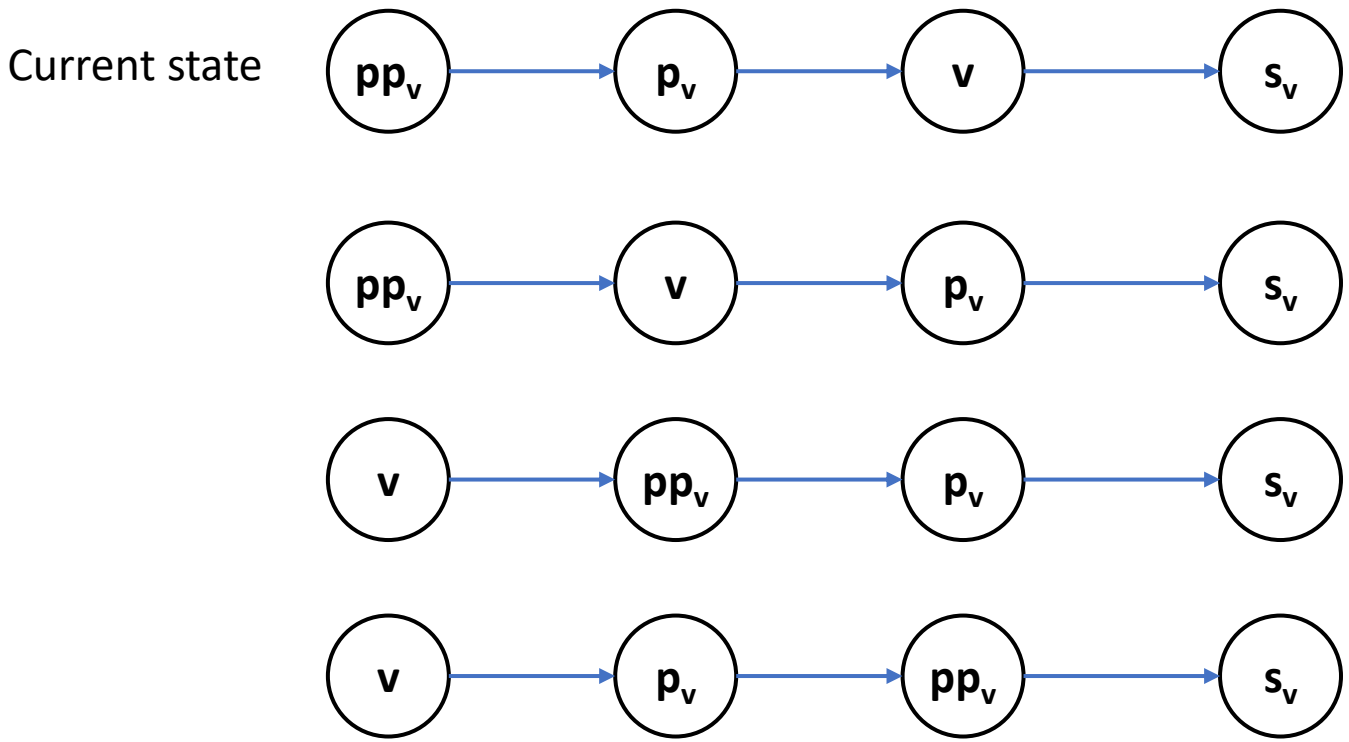


Mô hình MIP

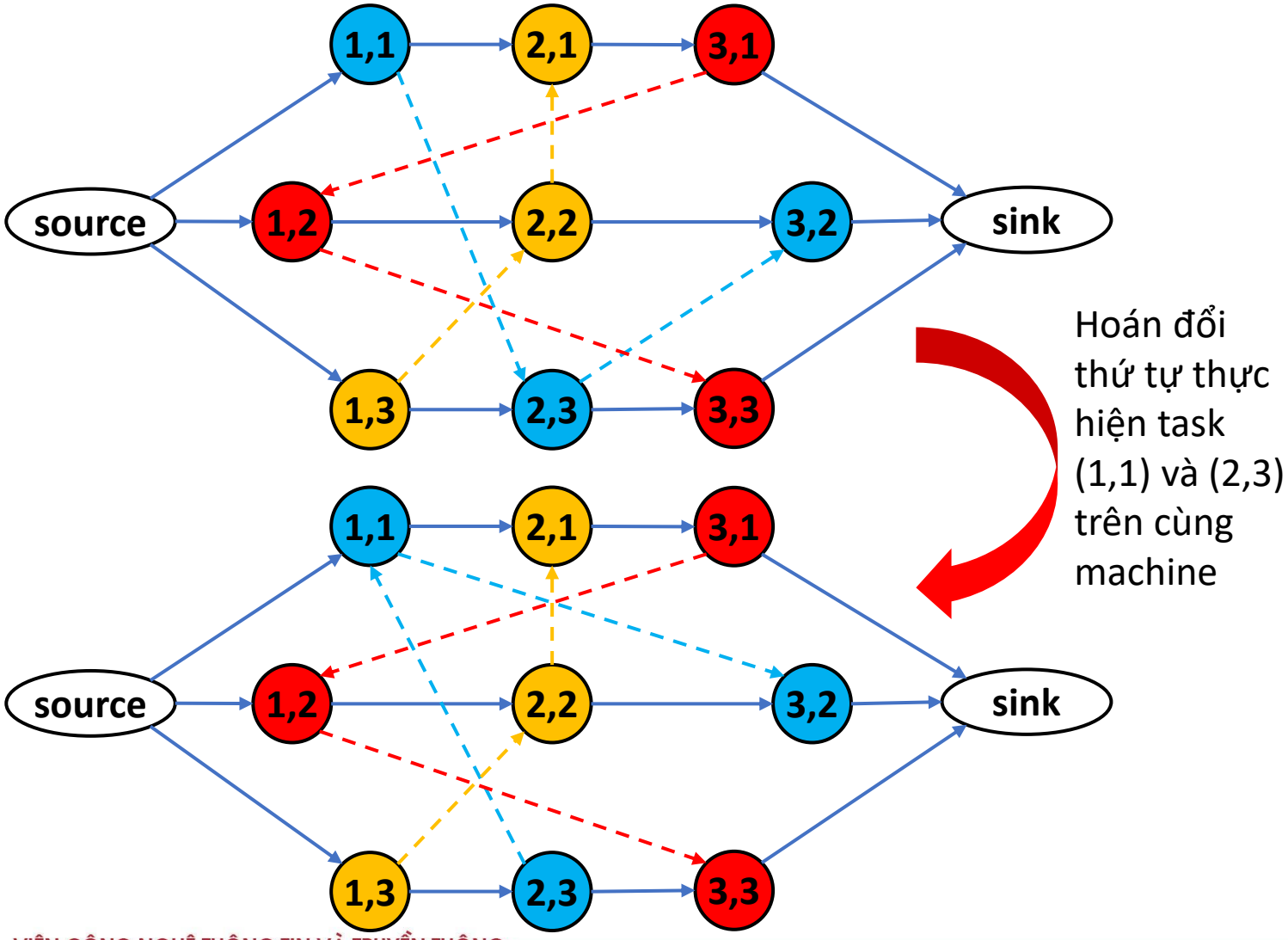
- Biến
  - $X(j, i)$ là thời gian bắt đầu của task $t(j, i)$
  - $Z(j_1,i_1,j_2,i_2) = 1$ nếu task $t(j_1, i_1)$ được thực hiện trước task $t(j_2, i_2)$ trên cùng 1 máy
  - Y: thời điểm hoàn thành toàn bộ các jobs (hàm mục tiêu)
- Ràng buộc
  - C1: $X(j, i) + d(j, i) \leq X(j, i+1), \forall j \in J, i = 1, 2, \ldots, \lambda_j - 1$
  - C2: $Z(j_1,i_1,j_2,i_2) + Z(j_2,i_2,j_1,i_1) = 1$, với mọi tasks $t(j_1,i_1)$, $t(j_2,i_2)$ trong đó $r(j_1,i_1) = r(j_2,i_2)$
  - C3: $X(j_2,i_2) \geq X(j_1,i_1) + d(j_1,i_1) - V(1-Z(j_1,i_1,j_2,i_2))$, với mọi tasks $t(j_1,i_1)$, $t(j_2,i_2)$ trong đó $r(j_1,i_1) = r(j_2,i_2)$, V là hằng số rất lớn
  - C4: $Y \geq X(j,\lambda_j) + d(j,\lambda_j), j \in J$
- Hàm mục tiêu: $Y \rightarrow$ minimize

Phương pháp tìm kiếm cục bộ

- Láng giềng của một phương án được sinh ra bằng việc hoán đổi vị trí các task trên cùng 1 máy theo các cách sau:

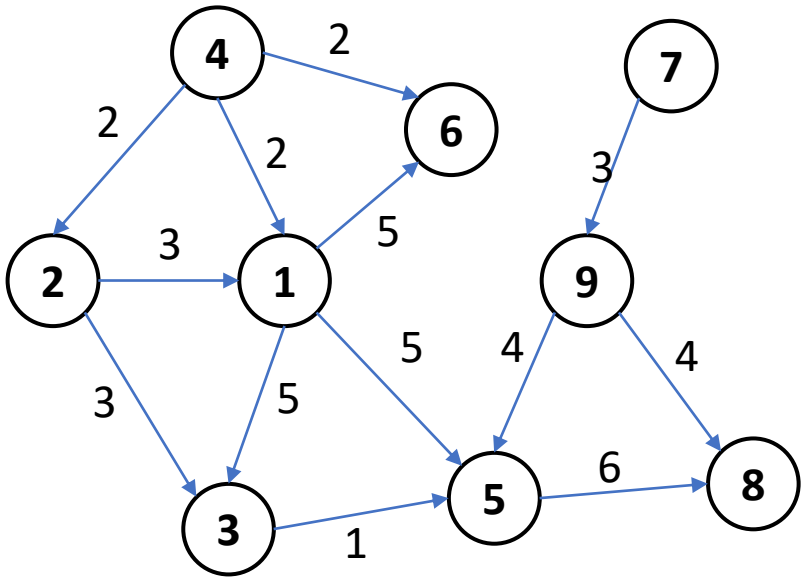


Phương pháp tìm kiếm cục bộ



Xác định thời gian hoàn thành

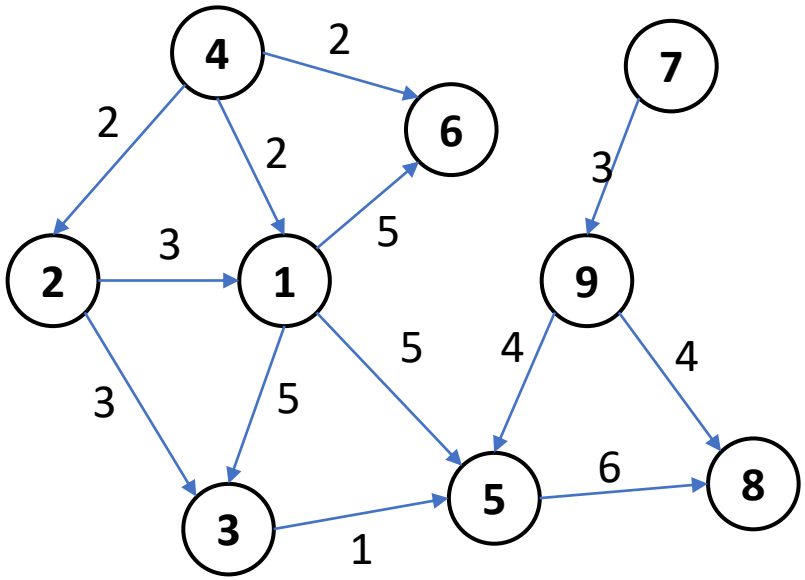
- Có n công việc 1, 2, …, n. công việc i có thời gian hoàn thành là d(i), với mọi i = 1,2,…, n. Giữa các công việc có quan hệ về thứ tự thực hiện, được biểu diễn bởi một tập các bộ (i,j) trong đó công việc j chỉ có thể được thực hiện sau khi công việc i được thực hiện xong. Cần xác định thời gian sớm nhất hoàn thành tất cả n công việc



Công việc	Thời gian hoàn thành
1	5
2	3
3	1
4	2
5	6
6	4
7	3
8	1
9	4

Xác định thời gian hoàn thành

- Có n công việc 1, 2, …, n. công việc i có thời gian hoàn thành là d(i), với mọi i = 1,2,…, n. Giữa các công việc có quan hệ về thứ tự thực hiện, được biểu diễn bởi một tập các bộ (i,j) trong đó công việc j chỉ có thể được thực hiện sau khi công việc i được thực hiện xong. Cần xác định thời gian sớm nhất hoàn thành tất cả n công việc



Công việc	Thời điểm thực hiện
1	5
2	2
3	10
4	0
5	11
6	10
7	0
8	17
9	3

Thời gian hoàn thành toàn bộ là 17+1 = 18

Xác định thời gian hoàn thành

- Thuật toán
  - $A(t)$: Danh sách các công việc mà chỉ có thể bắt đầu được thực hiện sau khi công việc t kết thúc
  - d(t): thời gian hoàn thành công việc t
  - $F(t)$: thời điểm sớm nhất của thể bắt đầu thực hiện công việc t (t = 1,…, n)

```
L = sắp xếp topo các công việc (sử dụng hàng đợi)
for i = 1 to n do
  F(i) = 0;
makeSpan = 0
for t in L do {// duyet DS các công việc t trong L từ trái qua phải
   if F(t) + d(t) > makeSpan then makeSpan = F(t) + d(t)
   for x in A(t) do
      F(x) = max(F(x), F(t) + d(t));
}
```

ĐẠI HỌC BÁCH KHOA

25 YEARS ANNIVERSARY SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

soict.hust.edu.vn/ fb.com/groups/soict