

# BÀI BÁO CÁO THU HOẠCH

## SINGLE MACHINE SCHEDULING

Nguyễn Chí Bằng

Ngày 12 tháng 6 năm 2024

# Mục lục

Mục lục	i
Danh mục các kí hiệu	ii
<b>1 Giới thiệu</b>	<b>iv</b>
1.1 Vấn đề . . . . .	iv
1.2 Ví dụ minh hoạ . . . . .	iv
<b>2 Phương pháp</b>	<b>v</b>
2.1 Sắp thu tu . . . . .	v
2.2 EDD . . . . .	v
2.2.1 $L_{\max}$ . . . . .	v
2.2.2 $T_{\max}$ . . . . .	v
2.3 SPT . . . . .	v
2.3.1 1 . . . . .	v
2.3.2 tổng quát . . . . .	v
2.4 release date . . . . .	v
2.4.1 preemptive . . . . .	v
2.4.2 non-preemptive . . . . .	v
<b>3 Kết luận</b>	<b>vi</b>

# Danh mục ký hiệu và ý nghĩa

$\alpha|\beta|\gamma$  Ký hiệu dùng để nhận dạng loại bài toán. Trong đó  $\alpha$  chỉ số lượng máy cần lập lịch, trường hợp cho máy đơn ta ký hiệu  $\alpha = 1$ , tức  $1|\beta|\gamma$ . Ký hiệu  $\beta$  chỉ đặc tính hay kiểu ràng buộc của bài toán. Ký hiệu  $\gamma$  chỉ hàm mục tiêu cần tối ưu.

$p_j$  Khoảng thời gian xử lý (processing time) của công việc thứ  $j$ , tức từ thời điểm công việc bắt đầu đến thời điểm công việc hoàn thành.

$r_j$  Thời điểm phát hành (release time) công việc thứ  $j$ . Nếu  $r_j$  xuất hiện trong trường  $\beta$  của bài toán, đồng nghĩa công việc thứ  $j$  sẽ không được phép bắt đầu trước thời điểm phát hành  $r_j$  ( $S_j \geq r_j$ ), ngược lại, nếu  $r_j$  không xuất hiện trong trường  $\beta$  của bài toán, các công việc sẽ được phép bắt đầu tại bất kỳ thời điểm nào.

$C_j$  Thời điểm hoàn thành (completion time) của công việc thứ  $j$ .

$S_j$  Thời điểm bắt đầu (starting time) của công việc thứ  $j$ , được định nghĩa bằng công thức  $S_j = \max(C_{j-1}, r_j)$ .

$W_j$  Thời gian chờ (waiting time) của công việc thứ  $j$ , tức khoảng thời gian kể từ thời điểm công việc được phát hành cho đến thời điểm bắt đầu công việc, được định nghĩa bằng công thức  $W_j = C_j - p_j - r_j = S_j - r_j$ .

$F_j$  Chu trình (flow time) của công việc thứ  $j$ , tức khoảng thời gian kể từ thời điểm công việc được phát hành cho đến khi hoàn thành, được định nghĩa bằng công thức  $F_j = C_j - r_j = W_j + p_j$ .

$w_j$  Trọng số (weight) của công việc thứ  $j$ , tức mức độ ưu tiên của công việc thứ  $j$ .

$d_j$  Thời điểm đáo hạn (due date) của công việc thứ  $j$ .

$L_j$  Mức độ trễ (lateness) của công việc thứ  $j$ , được định nghĩa là độ dài từ  $d_j$  đến  $C_j$ , xác định

$T_j$  Thời gian muộn (tardiness) của công việc thứ  $j$ .

---

- $E_j$  Thời gian sớm (earliness) của công việc thứ  $j$ .
- $prec$  Bài toán tồn tại ràng buộc có thứ tự (precedence constraint).  
Nếu  $prec$  xuất hiện trong trường  $\beta$  của bài toán, đồng nghĩa tồn tại những công việc đòi hỏi phải hoàn thành trước khi công việc khác được bắt đầu, hay còn gọi là công việc tiền nhiệm (predecessor) và công việc kế nhiệm (successor). Nếu trường hợp bài toán có mỗi công việc tồn tại tối đa một tiền nhiệm và một kế nhiệm, bài toán có ràng buộc dạng dây chuyền (chains). Trường hợp có tối đa một kế nhiệm, bài toán có ràng buộc dạng in-tree. Trường hợp có tối đa một tiền nhiệm, bài toán có ràng buộc dạng out-tree. Ngược lại, nếu  $prec$  không xuất hiện trong trường  $\beta$  của bài toán, bài toán sẽ được phép có các thứ tự công việc được sắp tự do.
- $prmp$  Bài toán tồn tại tính ưu tiên ngắt (preemption), thường được sử dụng khi có sự xuất hiện của  $r_j \neq 0$ . Nếu  $prmp$  xuất hiện trong trường  $\beta$  của bài toán thì công việc sẽ được phép ngắt quãng tại bất kỳ thời điểm nào để ưu tiên cho công việc khác nhằm mục đích tối ưu hàm mục tiêu của bài toán. Ngược lại, nếu  $prmp$  không xuất hiện trong trường  $\beta$  của bài toán, các công việc sẽ không được phép ngắt quãng.
-

# Chương 1

## Giới thiệu

1.1. Vấn đề

1.2. Ví dụ minh họa

# Chương 2

## Phương pháp

2.1. Sắp thu tu

2.2. EDD

2.2.1 Lmax

2.2.2 Tmax

2.3. SPT

2.3.1 1

2.3.2 tổng quát

2.4. release date

2.4.1 preemptive

2.4.2 non-preemptive

## Chương 3

## Kết luận