Job Shop Scheduling

Example 4.

$F_2 \mid \text{max}$

Flow Shop

Maximum Completion Time

Permutation Schedule

Jobs go onto $M_2$ in the same order as on $M_1$

We can assume a permutation schedule

\[ \ldots 4 \quad 3 \quad 2 \quad 1 \rightarrow M_1 \]

We assume that $j$ is ready to go on $M_2$

\[ \ldots 4 \quad 3 \quad 2 \quad 1 \rightarrow M_1 \]

\[ \cdots \quad j \quad k \quad \cdots \]

Old

New

$C_{\text{max}}$ unchanged
Permutation Schedule

\[ M_1 \]

\[ M_2 \]

No delays from now on

1. \( C_{\text{max}} \) is always the sum of \( N+1 \) processing times.
2. Therefore, subtract \( \beta \) from every processing time reduces all permutation schedule by the same amount.
Assume now that processing times are $a_1, a_2, \ldots, a_n$ on $M_1$ and $b_1, b_2, \ldots, b_n$ on $M_2$.

Let $\rho = \min \{ a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_n \}$.

$a'_i = a_i - \rho$ and $b'_i = b_i - \rho$, $i = 1, 2, \ldots, n$.

Case 1: $\rho = 0$.

Case 2: $\rho > 0$.

\[ \frac{A: \{ a_1, \ldots, a_n \}, \quad B = \{ b_1, \ldots, b_n \}}{\text{Johnson's Rule}} \]

Nothing is changed by this move.

\[ \rightarrow \text{Johnson Rule} \]

No change in delay.
P-NP QUESTION

What can be solved efficiently — what does this mean.

\[ n \rightarrow \text{Description of Problem} \]
\[ \text{YES/NO ANSWER} \]

1. CAN THIS GRAPH BE 2-COLORED?

2. IS THERE A TSP TOUR OF LENGTH \( L \)?

\[ \text{Computer} \quad \text{Turing Machine} \]
\[ \text{YES ANSWER IS PRODUCED IN TIME } T(n) \]
\[ \text{NO} \]

\[ \text{P is the set of problems for which there is a program which solves the problem in time } T(n) \leq \text{const} \]
\[ \text{NP complete Polynomial Time} \]

\[ \text{NP} \]
\[ \text{Non-Deterministic Polynomial Time} \]

- Computational can do many things simultaneously