一.(1) 如机器长时间工作会磨损率增加.因此要尽可能将总加工量均分到每台机器上.

以降低整体磨损.

$Pm||\sum C_j$ 每个工件可在多个机器上加工.且目标函数也不同(最优时解不一定相同).

(2) 即 $\min_{M_i} p_j \geq \max_{M_{i+1}} p_j$. 反设 $\exists i$ s.t 其不成立. 不妨令 $n_1 \leq n_2 \leq \cdots \leq n_m$. 否则可以交换两台机器

任取一最优解.

不妨设 $C_i = n_i(p_1+p_2+\cdots+p_{n_i})$ $\quad C_{i+1} = n_{i+1}(p_1'+p_2'+\cdots+p'_{n_{i+1}})$.

不妨令 $p_1 < p_1'$. 则交换 p_1 与 p_1'. 有 $C_i' = n_i(p_1'+p_2+\cdots+p_{n_i})$ $\quad C_{i+1}' = n_{i+1}(p_1+p_2'+\cdots+p'_{n_{i+1}})$

则 $\Delta \sum n_i p_i = C_i' + C_{i+1}' - C_i - C_{i+1} = n_i p_1' + n_{i+1} p_1 - n_i p_1 - n_{i+1} p_1'$

$= (n_{i+1} - n_i)(p_1 - p_1') \leq 0$.

故交换后目标函数值变小,与最优解矛盾.故假设不成立.

(3): $C_{j,1} = \sum_{i=1}^{j} j p_i$. 且 $n_1 \leq n_2 \leq \cdots \leq n_k$.

当 $j \equiv 0 \mod k$ 时. $\quad C_{j,k} = \sum_{i=0}^{k} n \sum_{s=1}^{n} p_{in+s}$. 其中 $n = j/k$. 表示每台机器上的工件数.

当 $j \equiv l \mod k$ 时. $\quad C_{j,k} = C_{j,k-1} - \sum_{i=1}^{l} p_i$.

故 $C_{j,k} = \begin{cases} \sum_{i=0}^{k} n \sum_{s=1}^{n} p_{in+s} & n = j/k \text{ 且 } j \equiv 0 \mod k. \\ C_{j,k-1} - \sum_{i=1}^{l} p_i. & j \equiv l \mod k \text{ 且 } l \neq 0. \end{cases}$

二.(1): 当 $t = C_j(\sigma^P)$ 时. 前 j 个工件已完工. 从而要花费 $\sum_{k=1}^{j} p_k$. 而每个工件还有准备时间的限制

故 $C_j(\sigma^P) \geq \sum_{k=1}^{j} p_k$.

(2): $\min r_j = r_{j-1} \quad \min r_0 = r_0$.

$$C_j(\sigma^N) \leq C_j(\sigma^P) + \sum_{i=1}^{j-1}(r_{i+1} - C_i(\sigma^N))$$

对右式后半部分有

$$\sum_{i=1}^{j-1}(r_{i+1} - C_i(\sigma^N)) \leq \sum_{i=1}^{j-1}(r_{i+1} - r_i + p_i) = \sum_{i=1}^{j-1} p_i + \sum_{i=1}^{j-1}(r_{i+1} - r_i) \leq \sum_{i=1}^{j-1} p_i \leq C_{j-1}(\sigma^P)$$

故 $C_j(\sigma^N) \leq C_j(\sigma^P) + C_{j-1}(\sigma^P) \leq 2C_j(\sigma^P)$.

(3): 先构造 $1|r_j|\sum C_j$ 的一个最优解. 假设其为 $\sigma = (1, 2 \cdots n)$

则 $C_j = \sum_{i=1}^{j} p_i + \sum_{i=1}^{j-1}(S_{i+1} - S_i + p_i)$ $\quad S_i$ 为 J_i 的开工时间.

按此插入 $n+1$ 排任务. J_{n+1} 的准备时间 $r_{n+1} = C_1 - 2\varepsilon$, 时长为 $p_{n+1} = \varepsilon$. 此时 $\sigma = (1, n+1, 2, \ldots, n)$.

再令 J_{n+2} 的准备时间为 $r_{n+2} = C_2' - 2\varepsilon$. 时长为 ε. ……

由此, 易知这 $2n$ 个任务的最优解为 $\sigma_0 = (1, n+1, 2, n+2, \ldots, n, 2n)$.

目标值 $\sum C_j^o = 2\sum C_j + \frac{1+n}{2} n\varepsilon$

若按 (2) 中算法得到的解为 $\sigma_* = (n+1, 1, n+2, 2, \ldots, 2n, n)$.

目标值

$$\begin{aligned}\sum C_j^* &= \sum C_j - \frac{1+n}{2} n\varepsilon + \sum C_j + np_1 + (n-1)p_2 + \cdots + p_n \\ &\le 2\sum C_j - \frac{1+n}{2} n\varepsilon + C_n(\sigma^P) + C_{n-1}(\sigma^P) + \cdots + C_1(\sigma^P) \\ &\le 3\sum C_j - \frac{1+n}{2} n\varepsilon.\end{aligned}$$

令 $\varepsilon \to 0$. 有 $\sum C_j^* / \sum C_j^o = \frac{3}{2}$

(4). 按 r_i 增序排列 n 个工件. 依次加工. 当工件 J_i 未完成但时间 $t = r_{i+1}$ 时. 转到加入 J_{i+1}. 缓存 J_i 至队列 L 队尾. 若某工件已加工完成未到下一工件准备时间. 则从 L 队首提取工件加工. 若 L 为空则机器闲置. 直至加工完全部工件.

排列需多项式时间, 后续操作显然也可在多项式时间内完成.