§15 Stochastic online scheduling for $\sum w_j C_j$

[Megow, Uetz, Vredeveld ~~2004~~ 2006]

2 phase model for $\sum w_j C_j$

- jobs arrive online and must be assigned to machines now
  - unknown number, have random processing time
- "next day", jobs are scheduled on the assigned machines in the "expected performance" model (optimally on every machine with WSEPT)
  - number is now known
- view this as a scheduling policy, analyze w.r.t. expected performance

Algorithm MinIncrease

- assign job to machine such that $\sum w_j C_j$ based on $\mathbb{E}[X_j]$
  - has minimum increase
  - can be done in polynomial time
- MinIncrease matches best known bounds of previous model
  - even better for NBUE processing times and release dates
- needs LP-based lower bounds in analysis, but not for defining the policy
- first combinatorial approximation algorithm for release dates

Notation: $j \to i$ if job j is assigned to machine i

priority order from WSEPT, i.e. $w_j / E(X_j)$ decreasing

$H(j) = \{ k \in V \mid$ higher priority or equal priority as $j \}$

$L(j) = V \setminus H(j)$ lower priority jobs

$k < j \;\hat{=}\; k$ arrives before j

tie breaking: according to incoming order

15.1 Algorithm MinIncrease MI

(1) Upon arrival of job j, assign it to machine i that minimizes

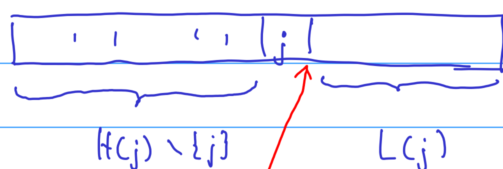
$$w_j \sum_{\substack{k \in H(j) \\ k < j \\ k \to i}} \mathbb{E}[X_k] + \mathbb{E}[X_j] \sum_{\substack{k \in L(j) \\ k \to i \\ k < j}} w_k + w_j \mathbb{E}[X_j] =: z(j,i)$$

(2) In the scheduling phase, schedule the jobs on every machine according to WSEPT (optimal per machine)

15.2 Lemma

$z(j,i)$ is the increase of $\sum_\ell w_\ell \mathbb{E}[C_\ell]$ on machine i when j is assigned to that machine and jobs are scheduled by WSEPT

Proof



$$z(j,i) = w_j \Bigg(\sum_{\substack{k \in H(j) \\ k < j \\ k \to i}} \mathbb{E}(X_k) + \mathbb{E}[X_j] \Bigg) + \mathbb{E}[X_j] \cdot \sum_{\substack{k \in L(j) \\ k < j \\ k \to i}} w_k$$

15.3 Lemma:

$$\underbrace{\mathbb{E}\Big[\sum_j w_j C_j^{MI} \Big]}_{\text{expected value of policy MI}} = \sum_j \underbrace{\min_i z(j,i)}_{\text{min increase for assigning } j \text{ on arrival}}$$

Proof: Let $C_j := C_j^{MI}$

$$\mathbb{E}\Big[\sum_j w_j C_j \Big] = \sum_j w_j \sum_{k \in H(j),\, k \to i_j,\ \text{same priority jobs } k \text{ have } k < j} \mathbb{E}[X_k]$$

partition into jobs that are before j on machine i_j

the machine that j is assigned to

$$= \sum_j w_j \sum_{k \in H(j),\, k \to i_j,\, k<j} E[X_k] + \underbrace{\sum_j w_j \sum_{\substack{k \in H(j) \\ k \to i_j \\ k > j}} E[X_k]}_{=:A} + \sum_j w_j E[X_j]$$

partition into jobs that arrive before / after j

have higher priority than j

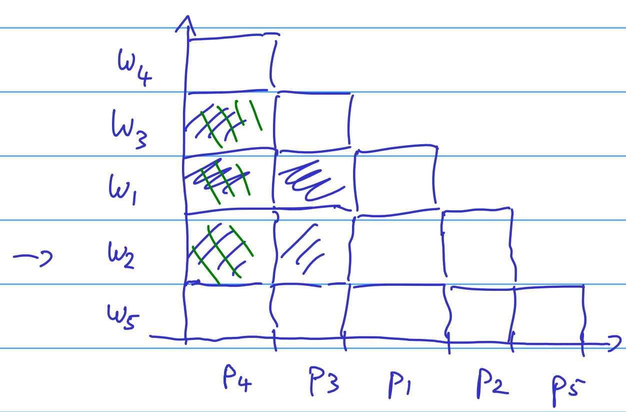
Claim $A = B := \sum_j E[X_j] \sum_{\substack{k \in L(j) \\ k<j \\ k \to i_j}} w_k$

Proof Claim: use different counting (rowwise and columnwise in 2-D Gantt chart)

Expl: 5 jobs on one machine in arrival order 1 2 3 4 5

with priority order 4 3 1 2 5

highest



$p_i \hat{=} E[X_j]$

$A \hat{=} \sum_j w_j \sum_{\substack{k \in H(j) \\ k>j}} E[X_k]$: $w_5 \cdot 0$ rowwise counting

$w_2(p_3 + p_4)$

$w_1(p_3 + p_4)$

$w_3 \cdot p_4$

$w_4 \cdot 0$

$B \overset{!}{=} \sum_j E[X_j] \sum_{\substack{k \in L(j) \\ k<j}} w_k$ $p_4(w_1 + w_2 + w_3)$

$$\Rightarrow \mathbb{E}[w_j C_j] \overset{A=B}{=} \sum_j \Big[w_j \sum_{\substack{k \in H(j) \\ k \to i_j \\ k < j}} \mathbb{E}[X_k] + \mathbb{E}[X_j] \sum_{\substack{k \in L(j) \\ k \to i_j \\ k < j}} w_k + w_j \mathbb{E}[X_j] \Big]$$

$$\overset{\text{La 15.2}}{=} \sum_j \min_i z(j,i)$$

15.4 Theorem:

Let $CV[X_j] \leq \Delta$. Then MI is a ϱ-approximation algorithm with $\varrho = 1 + \frac{(m-1)(\Delta+1)}{2m}$

Proof: $\mathbb{E}[MI(I)] \underset{\text{La 15.3}}{=} \sum_j \min_i z(j,i)$

$$\overset{\text{La 15.2}}{=} \sum_j \min_i \Big\{ w_j \sum_{\substack{k \in H(j) \\ k < j \\ k \to i}} \mathbb{E}[X_k] + \mathbb{E}[X_j] \sum_{\substack{k \in L(j) \\ k < j \\ k \to i}} w_k + w_j \mathbb{E}[X_j] \Big\}$$

$$= \sum_j \min_i \Big\{ \quad \text{''} \quad\quad \text{''} \quad \Big\} + \sum_j w_j \mathbb{E}[X_j]$$

$$\underset{\text{min} \,\leq\, \text{arithmetic mean}}{\leq} \sum_j \frac{1}{m} \sum_i \Big\{ \quad \text{''} \quad\quad \text{''} \quad \Big\} + \sum_j w_j \mathbb{E}[X_j]$$

$$\underset{\text{interchange summation}}{=} \sum_i \frac{1}{m} \Big\{ \sum_j w_j \sum_{\substack{k \in H(j) \\ k < j \\ k \to i}} \mathbb{E}[X_k] + \underbrace{\sum_j \mathbb{E}[X_j] \sum_{\substack{k \in L(j) \\ k < j \\ k \to i}} w_k}_{\text{Claim } A = B} \Big\} + \sum_j w_j \mathbb{E}[X_j]$$

$$= \sum_j w_j \sum_{\substack{k \in H(j) \\ k > j \\ k \to i}} \mathbb{E}[X_k]$$

$$= \sum_i \frac{1}{m} \sum_j w_j \sum_{\substack{k \in H(j) \\ k \neq j \\ k \to i}} \mathbb{E}[X_k] + \sum_j w_j \mathbb{E}[X_j]$$

$$= \sum_j \frac{1}{m} w_j \underbrace{\sum_i \sum_{\substack{k \in H(j) \\ k \neq j \\ k \to i}} \mathbb{E}[X_k]}_{= \sum_{k \in H(j),\, k \neq j} \mathbb{E}[X_k]} + \sum_j w_j \mathbb{E}[X_j]$$

Change summation order

$$= \sum_j \frac{1}{m} w_j \sum_{k \in H(j)} \mathbb{E}[X_k] + \frac{m-1}{m} \sum_j w_j \mathbb{E}[X_j]$$

include j in first sum

Now use (see next lemma)

$$\mathbb{E}(OPT(I)) \geq \sum_j w_j \frac{1}{m} \sum_{k \in H(j)} \mathbb{E}[X_k] - \frac{(m-1)(\Delta-1)}{2m} \sum_j w_j \mathbb{E}[X_j] \quad (1)$$

$$(1) \Rightarrow \mathbb{E}[MI(I)] \leq \mathbb{E}[OPI(I)] + \underbrace{\left[\frac{(m-1)(\Delta-1)}{2m} + \frac{m+1}{m}\right]}_{\frac{(m-1)(\Delta+1)}{2m}} \underbrace{\sum_j w_j \mathbb{E}[X_j]}_{\leq \mathbb{E}(OPT(I))}$$

$$\Rightarrow \mathbb{E}[MI(I)] \leq \rho \cdot \mathbb{E}[OPT(I)] \quad \square$$

15.5 Lemma: Consider priorities w.r.t. non-increasing $w_j / \mathbb{E}[X_j]$

Then (1) holds

Proof: Recall Theorem 14.11, i.e. an optimal solution to the LP defined by inequality (4) and $C_j^{LP} \geq E[X_j]$ in §14 is given by

$$C_j^{LP} = \frac{1}{m} \sum_{k \in H(j)} E[X_k] - \frac{(\Delta-1)(m+1)}{2m} E[X_j]$$

$$\Rightarrow \sum_j w_j C_j^{LP} = \sum_j w_j \frac{1}{m} E[X_k] - \frac{(\Delta-1)(m+1)}{2m} \sum_j w_j E[X_j]$$

and $E(OPT(I)) \geq \sum_j w_j C_j^{LP}$ $\square$

Remark.

(1) Performance guarantee of MI matches the best known in the offline setting, but does not require the knowledge of all jobs and their expected processing times in advance, but only when they arrive

(2) WSEPT and MI produce different schedules in general

(3) The lower bound of Theorem 14.12 applies also to MI

Exercise:

15.1 Show by example that WSEPT and MI generate different schedules.