

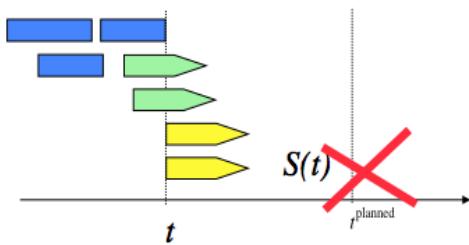
§12 SET - policies

Set policies: A general class of robust policies

Only exploitable information at time t

- set of completed jobs
- set of busy jobs

Jobs start only at completions of other jobs



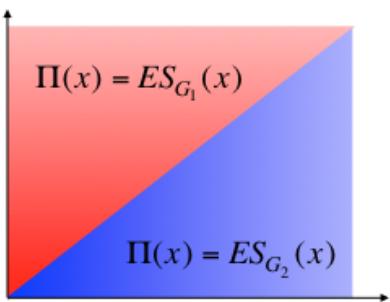
Special cases

- priority policies
- preselective policies

Set policies behave locally like ES-policies

For every set policy Π , there exists

- a partition of \mathbb{R}^n_+ into finitely many convex polyhedral cones Z_1, \dots, Z_k
- and feasible partial orders G_1, \dots, G_k such that $\Pi(x) = ES_{G_i}(x)$ for $x \in Z_i$



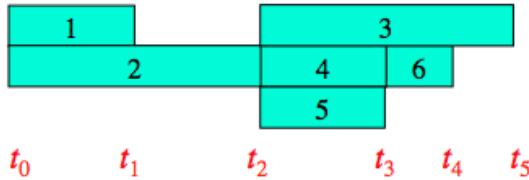
Graham anomalies only at boundaries of cones!

Stability for continuous distributions!

↑ boundaries of cones have
Lebesgue measure 0

Proof of local behaviour of set policies

Decisions depend only on sets
They determine an "abstract" interval order



Processing time vectors x leading to that interval order fulfil a system of homogeneous equations and inequalities

$x_1 < x_2 \quad x_4 + x_6 < x_3 \quad x_4 = x_5 \quad \rightarrow$ polyhedral cone

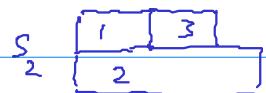
Example:

- ① Π must make a decision at $t=0$
 - ② $m=2$ say, Π starts 1, 2 at $t=0$
 - ③ then 3 as early as possible
- $\left. \begin{array}{l} \text{is a set policy} \\ \end{array} \right\}$

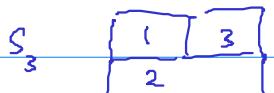
in total we get 7 different "abstract" schedules, which define polyhedral cones



$$x_1 < x_2, x_2 < x_1 + x_3$$



$$x_1 < x_2, x_1 + x_3 < x_2$$



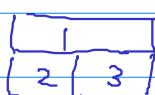
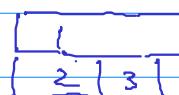
$$x_1 < x_2, x_1 + x_3 = x_2$$

induce the same interval order

G_1

$$\begin{matrix} (1) \\ (2) \end{matrix} \rightarrow (3)$$

minimal feasible



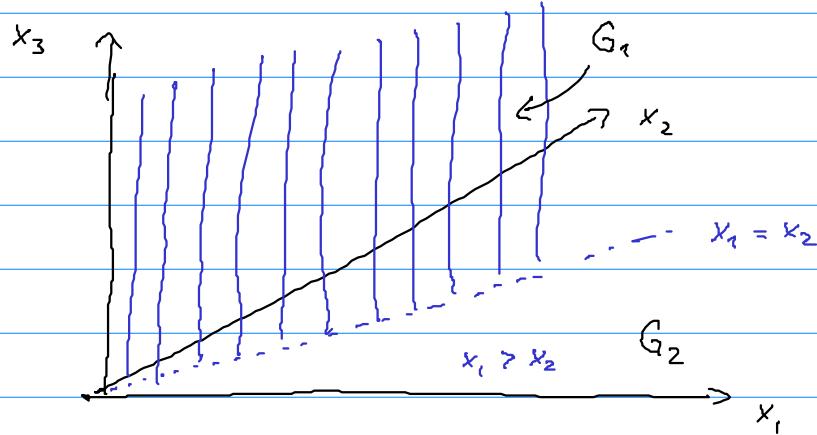
induce same interval order

$$\begin{matrix} (1) \\ (2) \end{matrix} \rightarrow (3)$$

G_2 minimal feasible

1	2
2	

=> interval order G_2 (1) \rightarrow (3) feasible, not minimal
 (2) \rightarrow

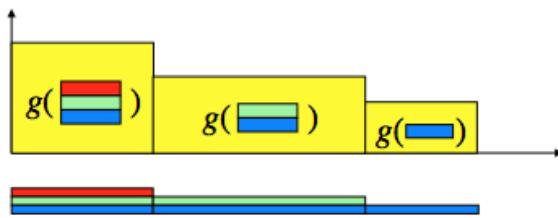


Optimality of set policies

If • all jobs are exponentially distributed and independent
 • the cost function κ is *additive*

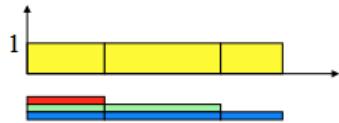
then there is an optimal set policy Π (among all policies).

κ is *additive* if there is a set function $g : 2^V \rightarrow \mathbb{R}$ (the *cost rate*)
 with $\kappa(C_1, \dots, C_n) = \int g(U(t))dt$ $U(t)$ = set of uncompleted jobs at t



Special cases of optimal set policies

$$\kappa = C_{\max}$$

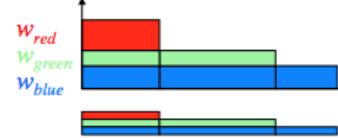


$$g(U) = \begin{cases} 1 & \text{if } U \neq \emptyset \\ 0 & \text{if } U = \emptyset \end{cases}$$

LEPT is optimal for
 $P | p_j \sim \exp | C_{\max}$

Weiss '82

$$\kappa = \sum w_j C_j$$



$$g(U) = \sum_{j \in U} w_j$$

SEPT is optimal for
 $P | p_j \sim \exp | \sum C_j$

Weiss & Pinedo '80

$[G, \bar{s}]$ general

$\} \Rightarrow \exists$ optimal policy
 that is a

set policy

← no precedence
 constraints, in machines

Given a set policy π and independent exponential processing times and additive cost function k

then $E[k^\pi]$ can be calculated via a decision tree

$E[k^\pi] = \text{expected cost until first completion} + \text{expected cost of remaining problem}$

↑
same problem with fewer jobs
because exponential distributions
are memoryless

$$= \sum_{\substack{\text{all possible sets } C \\ \text{of jobs that complete} \\ \text{together first}}} Q(C \text{ completes first}) \cdot [\text{expected first completion time} \cdot g(V)]$$

$$+ E[k^\pi_{\text{remaining problem}}]$$

$$= \sum_{j \in V} Q(j \text{ ends first}) [E(\text{first completion}) \cdot g(V) + E[k^\pi_{\text{remaining problem}}]]$$

↑
double completions

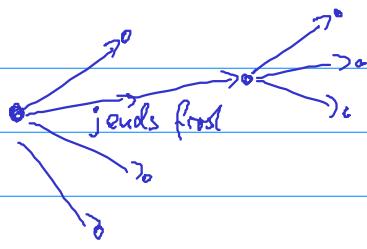
have probability 0

$$= \sum_{j \in V} \frac{\lambda_j}{\lambda_1 + \dots + \lambda_n} \left[\frac{1}{\lambda_j} \cdot g(V) + E[k^\pi_{\text{remaining problem}}] \right]$$

↑

job j has exponential

distribution $\exp(\lambda_j)$



Set policies may be involved

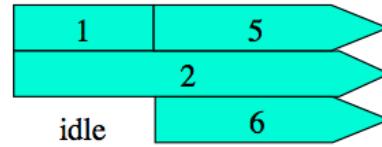
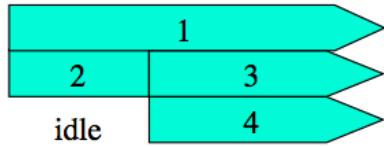
No precedence constraints, $m = 3$ identical machines, 6 jobs

$$X_j \sim \exp(a_j) \text{ with } a_1 = a_2 = a > 1, a_3 = a_4 = a_5 = a_6 = 1$$

$$g(U) = \begin{cases} w \gg 1 & \text{if } 1 \in U \text{ and } 2 \in U \\ 1 & \text{if } 1 \notin U \text{ and } (2, 5 \in U \text{ or } 2, 6 \in U) \\ 1 & \text{if } 2 \notin U \text{ and } (1, 3 \in U \text{ or } 1, 4 \in U) \\ 0 & \text{otherwise} \end{cases}$$

\Rightarrow complete one of 1, 2 fast
i.e. start with 1 and 2

Optimal set policy involves deliberate idleness



Q2: what are conditions on $g: 2^V \rightarrow \mathbb{R}_+^*$ such that
there is an optimal set policy without idle time?

Candidates: $\sum w_j c_j$ open

strange: g is submodular?

Exercises

Exercise is relevant

24. Calculate the expected makespan by the algorithm in
Remark 12.6 for the ES-policy $\Pi = \text{ES}_H$ with

H: $① \rightarrow ③$ $X_j \sim \exp(\lambda_j)$ with $\lambda_1 = 1$ $\lambda_2 = 2$
 $② \rightarrow ④$ $\lambda_3 = 2$ $\lambda_4 = 1$