Static Routing in Stochastic Scheduling: Performance Guarantees and Asymptotic Optimality

Chen Chen

Duke University
The Fuqua School of Business

10/22/2017

*joint work with:
Santiago Balseiro (Columbia) and David Brown (Duke)

Problem formulation

Complete a set $\mathcal{J} = \{1, \ldots, J\}$ of jobs using a set $\mathcal{M} = \{1, \ldots, M\}$ of machines. Each machine can process at most one job at a time; no preemption.

- p_{jm} := random variable for time it takes machine m to process job j
  - p_{jm} are independent across j (independence across m not required)
  - If j is assigned to m, p_{jm} not fully known until j completed
- Each job j has an associated positive weight w_j

Problem formulation

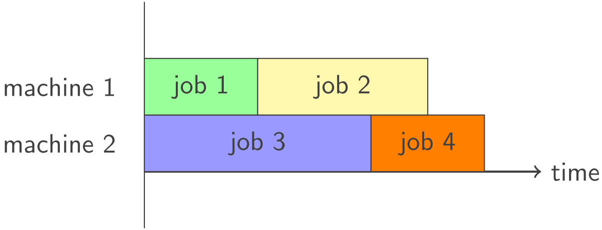
Complete a set $\mathcal{J} = \{1, \ldots, J\}$ of jobs using a set $\mathcal{M} = \{1, \ldots, M\}$ of machines. Each machine can process at most one job at a time; no preemption.

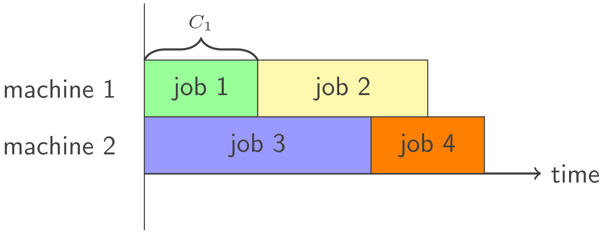
- p_{jm} := random variable for time it takes machine m to process job j
  - p_{jm} are independent across j (independence across m not required)
  - If j is assigned to m, p_{jm} not fully known until j completed
- Each job j has an associated positive weight w_j
- Objective is to minimize the expected weighted total completion time:

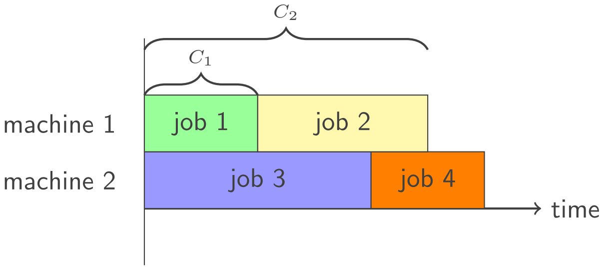
$$V^* = \min_{\pi \in \Pi} \mathbb{E}\left[\sum_{j \in \mathcal{J}} w_j C_j^\pi\right]$$

Π := set of non-anticipative policies

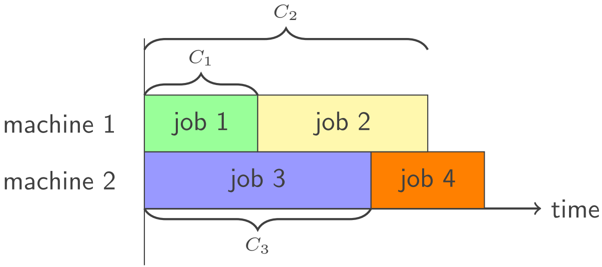
C_j^π := completion time of j using π (waiting time plus processing time)

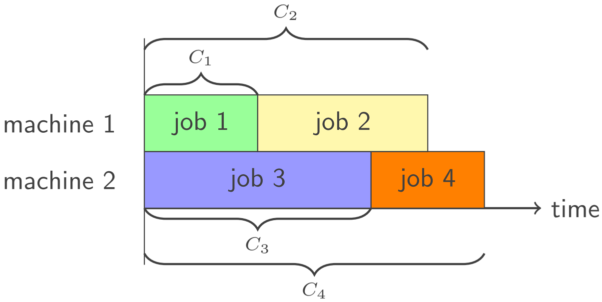

machine 1
machine 2
job 1
job 2
job 3
job 4
time


C_1
machine 1
machine 2
job 1
job 2
job 3
job 4
time



C_2
C_1
machine 1
job 1
job 2
machine 2
job 3
job 4
time


C_2
C_1
machine 1
job 1
job 2
machine 2
job 3
job 4
time
C_3


C_2
C_1
machine 1
job 1
job 2
machine 2
job 3
job 4
time
C_3
C_4

Motivation

- The problem can be written as a stochastic DP but it is very difficult to solve: curse of dimensionality, sometimes idling machines can be optimal

Motivation

- The problem can be written as a stochastic DP but it is very difficult to solve: curse of dimensionality, sometimes idling machines can be optimal
- Goal: find simple heuristic policies whose performance is close to optimality

Motivation

- The problem can be written as a stochastic DP but it is very difficult to solve: curse of dimensionality, sometimes idling machines can be optimal
- Goal: find simple heuristic policies whose performance is close to optimality
- Provide useful analytical bounds on the suboptimality of the heuristic policy

Motivation

- The problem can be written as a stochastic DP but it is very difficult to solve: curse of dimensionality, sometimes idling machines can be optimal
- Goal: find simple heuristic policies whose performance is close to optimality
- Provide useful analytical bounds on the suboptimality of the heuristic policy
- Main result is a uniform bound on the performance loss of a simple static routing policy; proved via the information relaxation duality approach

Literature review

- Möhring et al. (1999): job processing times are stochastic but identical across machines; analysis based on a polyhedral relaxation of the performance space.
- Skutella (2001) and Sethuraman and Squillante (1999): study a deterministic version of our problem; propose a constant factor approximation algorithm.
- Skutella et al. (2016): also study stochastic scheduling on unrelated machines;
  - The policy based on a novel time-indexed linear programming (LP) relaxation
  - Require a discretization of the time dimension that involves a large number of variables
  - Require a full information of all cumulative distributions of job processing times.

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1, \ \forall j \in \mathcal{J}$ is the *routing matrix*

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1, \ \forall j \in \mathcal{J}$ is the *routing matrix*
- Step 2: **Sequence** jobs on each machine.
  For a given routing, the optimal sequencing for each machine is easy:
  - $i \prec_m j$ if and only if $w_i / \mathbb{E}[p_{im}] \geq w_j / \mathbb{E}[p_{jm}]$
  - "WSEPT rule": shown in Rothkopf (1966) to be optimal for one machine

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1,\ \forall j \in \mathcal{J}$ is the *routing matrix*
- Step 2: **Sequence** jobs on each machine.
  For a given routing, the optimal sequencing for each machine is easy:
  - $i \prec_m j$ if and only if $w_i / \mathbb{E}[p_{im}] \geq w_j / \mathbb{E}[p_{jm}]$
  - "WSEPT rule": shown in Rothkopf (1966) to be optimal for one machine
- Performance of **static routing policy** given $\mathbf{x}$:

$$V^{\mathrm{R}}(\mathbf{x}) = \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} x_{jm} \Big(\quad \mathbb{E}[p_{jm}] \qquad + \sum_{i \prec_m j} x_{im} \mathbb{E}[p_{im}] \Big) \, .$$

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1, \ \forall j \in \mathcal{J}$ is the *routing matrix*
- Step 2: **Sequence** jobs on each machine.
  For a given routing, the optimal sequencing for each machine is easy:
  - $i \prec_m j$ if and only if $w_i / \mathbb{E}[p_{im}] \geq w_j / \mathbb{E}[p_{jm}]$
  - "WSEPT rule": shown in Rothkopf (1966) to be optimal for one machine
- Performance of **static routing policy** given $\mathbf{x}$:

$$V^{\mathrm{R}}(\mathbf{x}) = \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} x_{jm} \Big(\underbrace{\mathbb{E}[p_{jm}]}_{j\text{'s processing time}} + \sum_{i \prec_m j} x_{im} \mathbb{E}[p_{im}] \Big) .$$

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1,\ \forall j \in \mathcal{J}$ is the *routing matrix*
- Step 2: **Sequence** jobs on each machine.
  For a given routing, the optimal sequencing for each machine is easy:
  - $i \prec_m j$ if and only if $w_i / \mathbb{E}[p_{im}] \geq w_j / \mathbb{E}[p_{jm}]$
  - "WSEPT rule": shown in Rothkopf (1966) to be optimal for one machine
- Performance of **static routing policy** given $\mathbf{x}$:

$$V^{\mathrm{R}}(\mathbf{x}) = \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} x_{jm} \Big(\underbrace{\mathbb{E}[p_{jm}]}_{j\text{'s processing time}} + \overbrace{\sum_{i \prec_m j} x_{im} \mathbb{E}[p_{im}]}^{j\text{'s waiting time}} \Big) .$$

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1,\ \forall j \in \mathcal{J}$ is the *routing matrix*
- Step 2: **Sequence** jobs on each machine.
  For a given routing, the optimal sequencing for each machine is easy:
  - $i \prec_m j$ if and only if $w_i / \mathbb{E}[p_{im}] \geq w_j / \mathbb{E}[p_{jm}]$
  - "WSEPT rule": shown in Rothkopf (1966) to be optimal for one machine
- Performance of **static routing policy** given $\mathbf{x}$:

$$V^{\mathrm{R}}(\mathbf{x}) = \sum_{j \in \mathcal{J}} w_j \underbrace{\sum_{m \in \mathcal{M}} x_{jm} \Big(\underbrace{\mathbb{E}[p_{jm}]}_{j\text{'s processing time}} + \overbrace{\sum_{i \prec_m j} x_{im} \mathbb{E}[p_{im}]}^{j\text{'s waiting time}} \Big)}_{\mathbb{E}[C_j]} .$$

Static routing policies

We consider static policies: commit jobs to machines from the beginning

- Step 1: **Route** jobs to machines.
  - Each job j is routed (independently) to machine m with probability x_{jm}
  - $\mathbf{x} = (x_{jm})_{jm} \in \mathbb{R}_+^{J \times M}$, with $\sum_{m \in \mathcal{M}} x_{jm} = 1, \ \forall j \in \mathcal{J}$ is the *routing matrix*
- Step 2: **Sequence** jobs on each machine.
  For a given routing, the optimal sequencing for each machine is easy:
  - $i \prec_m j$ if and only if $w_i / \mathbb{E}[p_{im}] \geq w_j / \mathbb{E}[p_{jm}]$
  - "WSEPT rule": shown in Rothkopf (1966) to be optimal for one machine
- Performance of **static routing policy** given $\mathbf{x}$:

$$V^{\mathrm{R}}(\mathbf{x}) = \sum_{j \in \mathcal{J}} w_j \underbrace{\sum_{m \in \mathcal{M}} x_{jm} \Big(\underbrace{\mathbb{E}[p_{jm}]}_{j\text{'s processing time}} + \overbrace{\sum_{i \prec_m j} x_{im} \mathbb{E}[p_{im}]}^{j\text{'s waiting time}} \Big)}_{\mathbb{E}[C_j]} .$$

Will show: for a good choice of $\mathbf{x}$, the static routing policy approaches optimality as the number of jobs $J \to \infty$ (under some conditions...)

Convex relaxations

Given $\mathbf{x}$, evaluating $V^{\mathrm{R}}(\mathbf{x})$ is easy; optimizing $V^{\mathrm{R}}(\mathbf{x})$ is hard (not convex...)

Convex relaxations

Given $\mathbf{x}$, evaluating $V^{\mathrm{R}}(\mathbf{x})$ is easy; optimizing $V^{\mathrm{R}}(\mathbf{x})$ is hard (not convex...)

- Consider instead a relaxation of the objective function:

$$Z^{\mathrm{R}}(\mathbf{x}) := V^{\mathrm{R}}(\mathbf{x}) - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} \mathbb{E}[p_{jm}] x_{jm} (1 - x_{jm})$$

Convex relaxations

Given $\mathbf{x}$, evaluating $V^{\mathrm{R}}(\mathbf{x})$ is easy; optimizing $V^{\mathrm{R}}(\mathbf{x})$ is hard (not convex...)

- Consider instead a relaxation of the objective function:

$$Z^{\mathrm{R}}(\mathbf{x}) := V^{\mathrm{R}}(\mathbf{x}) - \underbrace{\frac{1}{2}\sum_{j\in\mathcal{J}} w_j \sum_{m\in\mathcal{M}} \mathbb{E}[p_{jm}]x_{jm}(1-x_{jm})}_{x_{jm}\in[0,1] \implies Z^{\mathrm{R}}(\mathbf{x}) \le V^{\mathrm{R}}(\mathbf{x})}$$

Convex relaxations

Given $\mathbf{x}$, evaluating $V^{\mathrm{R}}(\mathbf{x})$ is easy; optimizing $V^{\mathrm{R}}(\mathbf{x})$ is hard (not convex...)

- Consider instead a relaxation of the objective function:

$$Z^{\mathrm{R}}(\mathbf{x}) := V^{\mathrm{R}}(\mathbf{x}) - \underbrace{\frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} \mathbb{E}[p_{jm}] x_{jm}(1 - x_{jm})}_{x_{jm} \in [0,1] \implies Z^{\mathrm{R}}(\mathbf{x}) \leq V^{\mathrm{R}}(\mathbf{x})}$$

- Moreover, $Z^{\mathrm{R}}(\mathbf{x})$ is convex and quadratic in $\mathbf{x}$ (Skutella (2001) and Sethuraman and Squillante (1999) - study deterministic case)

Convex relaxations

Given $\mathbf{x}$, evaluating $V^{\mathrm{R}}(\mathbf{x})$ is easy; optimizing $V^{\mathrm{R}}(\mathbf{x})$ is hard (not convex...)

- Consider instead a relaxation of the objective function:

$$Z^{\mathrm{R}}(\mathbf{x}) := V^{\mathrm{R}}(\mathbf{x}) - \underbrace{\frac{1}{2}\sum_{j\in\mathcal{J}} w_j \sum_{m\in\mathcal{M}} \mathbb{E}[p_{jm}] x_{jm}(1 - x_{jm})}_{x_{jm}\in[0,1] \implies Z^{\mathrm{R}}(\mathbf{x}) \le V^{\mathrm{R}}(\mathbf{x})}$$

- Moreover, $Z^{\mathrm{R}}(\mathbf{x})$ is convex and quadratic in $\mathbf{x}$ (Skutella (2001) and Sethuraman and Squillante (1999) - study deterministic case)
- We route according to an optimal solution $\mathbf{x}^*$ of the convex problem:

$$\underset{\mathbf{x}\in\mathbb{R}_+^{J\times M}}{\text{minimize}} \left\{ Z^{\mathrm{R}}(\mathbf{x}) \ : \ \sum_{m\in\mathcal{M}} \mathbf{x}_m = \mathbf{1} \right\}.$$

- Optimization only depends on processing times through $\mathbb{E}[p_{jm}]$
- Let $Z^{\mathrm{R}} := Z^{\mathrm{R}}(\mathbf{x}^*)$ and $V^{\mathrm{R}} := V^{\mathrm{R}}(\mathbf{x}^*)$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2}\sum_{j\in\mathcal{J}} w_j \sum_{m\in\mathcal{M}} p_{jm} x^*_{jm}(1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow Z^{\mathrm{R}} = \min_{\mathbf{x}\in\mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \leq \min_{\mathbf{x}\in\mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) = V^* \leq V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \leq V^* + \frac{1}{2}\sum_{j\in\mathcal{J}} w_j \sum_{m\in\mathcal{M}} p_{jm} x^*_{jm}(1 - x^*_{jm})$$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow Z^{\mathrm{R}} = \min_{\mathbf{x} \in \mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \leq \min_{\mathbf{x} \in \mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) = V^* \leq V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \quad \leq \quad V^* \quad + \frac{M-1}{2M} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} p_{jm}$$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow Z^{\mathrm{R}} = \min_{\mathbf{x} \in \mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \le \min_{\mathbf{x} \in \mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) = V^* \le V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \quad \le \quad V^* \quad + \underbrace{\frac{M-1}{2M} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} p_{jm}}_{O(J)}$$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow Z^{\mathrm{R}} = \min_{\mathbf{x} \in \mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \leq \min_{\mathbf{x} \in \mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) = V^* \leq V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \leq \underbrace{V^*}_{\Omega(J^2)} + \underbrace{\frac{M-1}{2M} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} p_{jm}}_{O(J)}$$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow Z^{\mathrm{R}} = \min_{\mathbf{x} \in \mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \leq \min_{\mathbf{x} \in \mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) = V^* \leq V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \quad \leq \quad \underbrace{V^*}_{\Omega(J^2)} + \underbrace{\frac{M-1}{2M} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} p_{jm}}_{O(J)}$$

- Relative suboptimality of static routing: $\xrightarrow{J \to \infty} 0$

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow$ $Z^{\mathrm{R}} = \min_{\mathbf{x} \in \mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \leq \min_{\mathbf{x} \in \mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) = V^* \leq V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \quad \leq \quad \underbrace{V^*}_{\Omega(J^2)} + \underbrace{\frac{M-1}{2M} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} p_{jm}}_{O(J)}$$

- Relative suboptimality of static routing: $\xrightarrow{J \to \infty} 0$

Can we derive similar guarantees in the **stochastic** case?

Insights from the deterministic case

When the p_{jm} are deterministic, static routing is optimal (no uncertainty!)

How well does $\mathbf{x}^*$ perform in this case?

- Skutella (2001), Sethuraman and Squillante (1999): within 3/2 of V^*
- On the other hand:

$$Z^{\mathrm{R}} = V^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \sum_{m \in \mathcal{M}} p_{jm} x^*_{jm} (1 - x^*_{jm})$$

- Problem is deterministic $\Rightarrow Z^{\mathrm{R}} = \min_{\mathbf{x} \in \mathcal{X}} Z^{\mathrm{R}}(\mathbf{x}) \le \min_{\mathbf{x} \in \mathcal{X}} V^{\mathrm{R}}(\mathbf{x}) \ge V^* \le V^{\mathrm{R}}$:

$$V^{\mathrm{R}} \quad \le \quad \underbrace{V^*}_{\Omega(J^2)} + \underbrace{\frac{M-1}{2M} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} p_{jm}}_{O(J)}$$

- Relative suboptimality of static routing: $\xrightarrow{J \to \infty} 0$

Can we derive similar guarantees in the **stochastic** case?

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j \in \mathcal{J}, m \in \mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j \in \mathcal{J}, m \in \mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*
  1. Sample $\mathbf{p}$
  2. Solve a **deterministic** scheduling problem given $\mathbf{p}$
  3. Average over $\mathbf{p}$

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j\in\mathcal{J},m\in\mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*
  1. Sample $\mathbf{p}$
  2. Solve a **deterministic** scheduling problem given $\mathbf{p}$
  3. Average over $\mathbf{p}$

  ... usually gives a weak lower bound!

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j \in \mathcal{J}, m \in \mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*
  1. Sample $\mathbf{p}$
  2. Solve a **deterministic** scheduling problem given $\mathbf{p}$
  3. Average over $\mathbf{p}$

  ... usually gives a weak lower bound!

- Knowing all the processing times in advance is extremely valuable

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j\in\mathcal{J},m\in\mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*
  1. Sample $\mathbf{p}$
  2. Solve a **deterministic** scheduling problem given $\mathbf{p}$
  3. Average over $\mathbf{p}$

  ... usually gives a weak lower bound!

- Knowing all the processing times in advance is extremely valuable
  (a) Can route jobs to fast machines in every sample path

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j \in \mathcal{J}, m \in \mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*
  1. Sample $\mathbf{p}$
  2. Solve a **deterministic** scheduling problem given $\mathbf{p}$
  3. Average over $\mathbf{p}$

  ... usually gives a weak lower bound!

- Knowing all the processing times in advance is extremely valuable
  (a) Can route jobs to fast machines in every sample path
  (b) Can sequence jobs on a given machine differently in every sample path - prioritize relatively fast jobs

Perfect information bound

Returning to the stochastic problem... imagine all processing times $\mathbf{p} := \{p_{jm}\}_{j\in\mathcal{J},m\in\mathcal{M}}$ are revealed before scheduling jobs

- Leads to a lower bound on V^*
  1. Sample $\mathbf{p}$
  2. Solve a **deterministic** scheduling problem given $\mathbf{p}$
  3. Average over $\mathbf{p}$

  ... usually gives a weak lower bound!

- Knowing all the processing times in advance is extremely valuable
  (a) Can route jobs to fast machines in every sample path
  (b) Can sequence jobs on a given machine differently in every sample path - prioritize relatively fast jobs

To improve the lower bound: we need a **penalty**!

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

$$\mathbb{E}\left[\textbf{Cost with any non-anticipative policy}\right]$$

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

$$\mathbb{E}\left[\textbf{Cost with any non-anticipative policy}\right]$$

$$=$$

$$\mathbb{E}\left[\textbf{Cost plus penalty with any non-anticipative policy}\right]$$

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

$$\mathbb{E}\left[\textbf{Cost with any non-anticipative policy}\right]$$

$$=$$

$$\mathbb{E}\left[\textbf{Cost plus penalty with any non-anticipative policy}\right]$$

$$\geq$$

$$\mathbb{E}\left[\textbf{Cost plus penalty with perfect information}\right]$$

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

$$\mathbb{E}\left[\textbf{Cost with any non-anticipative policy}\right]$$

$$=$$

$$\mathbb{E}\left[\textbf{Cost plus penalty with any non-anticipative policy}\right]$$

$$\geq$$

$$\underbrace{\mathbb{E}\left[\textbf{Cost plus penalty with perfect information}\right]}_{\text{optimize cost plus penalty along each path}}$$

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

$$\mathbb{E}\left[\textbf{Cost with any non-anticipative policy}\right]$$

$$=$$

$$\mathbb{E}\left[\textbf{Cost plus penalty with any non-anticipative policy}\right]$$

$$\geq$$

$$\underbrace{\mathbb{E}\left[\textbf{Cost plus penalty with perfect information}\right]}_{\text{optimize cost plus penalty along each path}}$$

- In particular: lower bound on the performance of an optimal policy

Improving perfect information bounds

- Include a penalty that compensates for the extra information
- "Dual feasibility": Expected Penalty = 0 for all non-anticipative policies

With any dual feasible penalty:

$$\mathbb{E}\left[\textbf{Cost with any non-anticipative policy}\right]$$

$$=$$

$$\mathbb{E}\left[\textbf{Cost plus penalty with any non-anticipative policy}\right]$$

$$\geq$$

$$\underbrace{\mathbb{E}\left[\textbf{Cost plus penalty with perfect information}\right]}_{\text{optimize cost plus penalty along each path}}$$

- In particular: lower bound on the performance of an optimal policy

With the right penalty: provides a **tight** bound!

Penalty

- Consider a decomposed penalty:

$$Y^{\pi} = Y^{\pi}_{\mathrm{S}} + Y^{\pi}_{\mathrm{R}}$$

Penalty

- Consider a decomposed penalty:

$$Y^{\pi} \; = \; Y^{\pi}_{\mathrm{S}} + Y^{\pi}_{\mathrm{R}}$$

  - Y^{π}_{S} is a **sequencing** penalty: "push" to the WSEPT sequencing

Penalty

- Consider a decomposed penalty:

$$Y^{\pi} = Y^{\pi}_{\mathrm{S}} + Y^{\pi}_{\mathrm{R}}$$

  - Y^{π}_{S} is a **sequencing** penalty: "push" to the WSEPT sequencing
  - Y^{π}_{R} is a **routing** penalty: "push" to routing with $\mathbf{x}^*$

Penalty

- Consider a decomposed penalty:

$$Y^{\pi} = Y^{\pi}_{\mathrm{S}} + Y^{\pi}_{\mathrm{R}}$$

  - Y^{π}_{S} is a **sequencing** penalty: "push" to the WSEPT sequencing
  - Y^{π}_{R} is a **routing** penalty: "push" to routing with $\mathbf{x}^*$

- Dual feasibility - for all non-anticipative policies $\pi \in \Pi$:

$$\mathbb{E}[Y^{\pi}_{\mathrm{S}}] = 0 \text{ and } \mathbb{E}[Y^{\pi}_{\mathrm{R}}] = 0, \text{ thus } \mathbb{E}[Y^{\pi}] = 0.$$

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where $S_{jm}^{\pi} =$ start time of j on m.

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where $S_{jm}^{\pi} =$ start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

$\sum_i w_i C_i$ **Penalty**

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

p_{jm} $\boldsymbol{\sum_i w_i C_i}$ **Penalty**

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

p_{jm}	$\sum_i w_i C_i$	**Penalty**
Low		

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

p_{jm}	$\sum_i w_i C_i$	**Penalty**
Low	⟵ advance j	

Sequencing penalty

Set $Y_{\text{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\text{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

p_{jm}	$\sum_i w_i C_i$	Penalty
Low	⟵ advance j	⟶ delay j

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

p_{jm}	$\sum_i w_i C_i$	**Penalty**
Low	⟵ advance j	delay j ⟶
High		

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

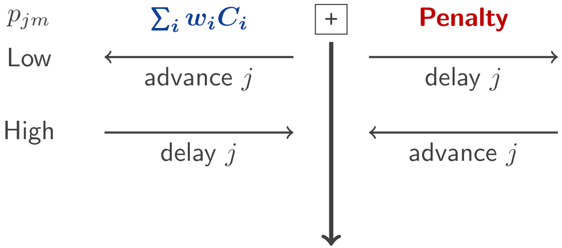
p_{jm}	$\sum_i w_i C_i$	**Penalty**
Low	← advance j	→ delay j
High	→ delay j	← advance j

Sequencing penalty

Set $Y_{\mathrm{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\mathrm{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective

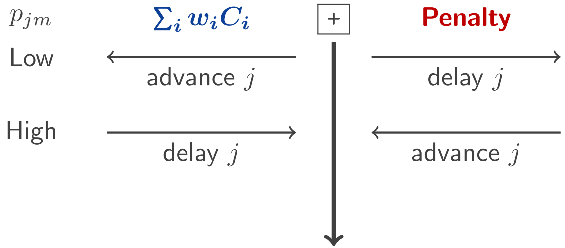


Sequencing penalty

Set $Y_{\text{S}}^{\pi} = \sum_{j,m} w_j S_{jm}^{\pi} \left(\frac{p_{jm}}{\mathbb{E}[p_{jm}]} - 1 \right)$, where S_{jm}^{π} = start time of j on m.

Dual feasibility: $\mathbb{E}[Y_{\text{S}}^{\pi}] = 0$ for all $\pi \in \Pi$ ✓

Combined objective



Optimal: always sequence in order of $\frac{w_j}{\mathbb{E}[p_{jm}]} \Longrightarrow$ WSEPT!

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$.

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$. Dual feasible? ✓

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$. Dual feasible? ✓

Good choice for λ_{jm}?

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$. Dual feasible? ✓

Good choice for λ_{jm}?

Inspiration: $\underset{\mathbf{x} \in \mathbb{R}_+^{J \times M}}{\text{minimize}} \left\{ Z^{\mathrm{R}}(\mathbf{x}) \ : \ \sum_{m \in \mathcal{M}} \mathbf{x}_m = \mathbf{1} \right\}$

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$. Dual feasible? ✓

Good choice for λ_{jm}?

Inspiration: $\underset{\mathbf{x} \in \mathbb{R}_{+}^{J \times M}}{\text{minimize}} \left\{ Z^{\mathrm{R}}(\mathbf{x}) \ : \ \sum_{m \in \mathcal{M}} \mathbf{x}_m = \mathbf{1} \right\}$

↑ Lagrange multiplier $\boldsymbol{\nu}^* \in \mathbb{R}^J$

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$. Dual feasible? ✓

Good choice for λ_{jm}?

Inspiration: $\underset{\mathbf{x} \in \mathbb{R}_+^{J \times M}}{\text{minimize}} \left\{ Z^{\mathrm{R}}(\mathbf{x}) \;:\; \sum_{m \in \mathcal{M}} \mathbf{x}_m = \mathbf{1} \right\}$

↑ Lagrange multiplier $\boldsymbol{\nu}^* \in \mathbb{R}^J$

Marginal cost of assigning j to m at optimality $= \frac{\partial Z^{\mathrm{R}}(\mathbf{x}^*)}{\partial x_{jm}} = \nu_j^*$

Routing penalty

Set $Y_{\mathrm{R}}^{\pi} = \sum_{j,m} \lambda_{jm} \left(1 - \frac{p_{jm}}{\mathbb{E}[p_{jm}]}\right) x_{jm}$ for some $\lambda_{jm} \in \mathbb{R}$. Dual feasible? ✓

Good choice for λ_{jm}? $\boxed{\lambda_{jm} = \nu_j^*}$

Inspiration: $\underset{\mathbf{x} \in \mathbb{R}_+^{J \times M}}{\text{minimize}} \left\{ Z^{\mathrm{R}}(\mathbf{x}) \;:\; \sum_{m \in \mathcal{M}} \mathbf{x}_m = \mathbf{1} \right\}$

↑ Lagrange multiplier $\boldsymbol{\nu}^* \in \mathbb{R}^J$

Marginal cost of assigning j to m at optimality $= \frac{\partial Z^{\mathrm{R}}(\mathbf{x}^*)}{\partial x_{jm}} = \nu_j^*$

Key technical result

With the sequencing and routing penalites, we obtain *in every sample path*:

Key technical result

With the sequencing and routing penalites, we obtain *in every sample path*:

Penalized perfect information optimal value

$\geq$

Key technical result

With the sequencing and routing penalites, we obtain *in every sample path*:

Penalized perfect information optimal value

$$\geq$$

$$\boxed{Z^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}}$$

Key technical result

With the sequencing and routing penalites, we obtain *in every sample path*:

Penalized perfect information optimal value

$$\geq$$

$$\underbrace{\boxed{Z^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}}}_{\Longrightarrow \text{ Lower bound on } V^*}$$

Key technical result

With the sequencing and routing penalites, we obtain *in every sample path*:

Penalized perfect information optimal value

$$\geq$$

$$\boxed{Z^{\mathrm{R}} - \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}}$$

$\Longrightarrow$ Lower bound on V^*

- Use Lagrange multipliers $\boldsymbol{\nu}^*$ to dualize the assignment constraint of the penalized perfect information relaxation problem
- Can show $x^*_{jm}\mathbb{E}[p_{jm}]/p_{jm}$ is the optimal routing

Key technical result

With the sequencing and routing penalites, we obtain *in every sample path*:

Penalized perfect information optimal value

$$\geq$$

$$\boxed{Z^{\mathrm{R}} - \frac{1}{2}\sum_{j\in\mathcal{J}} w_j \max_{m\in\mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}}$$

$\Longrightarrow$ Lower bound on V^*

- Use Lagrange multipliers $\boldsymbol{\nu}^*$ to dualize the assignment constraint of the penalized perfect information relaxation problem
- Can show $x^*_{jm}\mathbb{E}[p_{jm}]/p_{jm}$ is the optimal routing

similar to the primal-dual scheme!

Performance result

The performance V^{R} of the static routing policy satisfies:

$$V^* \leq V^{\mathrm{R}} \leq V^* + \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \Big(\frac{M-1}{M} \max_{m \in \mathcal{M}} \mathbb{E}[p_{jm}] + \max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]} \Big).$$

Performance result

The performance V^{R} of the static routing policy satisfies:

$$V^* \leq V^{\mathrm{R}} \leq V^* + \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \Big(\frac{M-1}{M} \max_{m \in \mathcal{M}} \mathbb{E}[p_{jm}] + \max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]} \Big).$$

- Two terms in the performance gap:

Performance result

The performance V^{R} of the static routing policy satisfies:

$$V^* \le V^{\mathrm{R}} \le V^* + \frac{1}{2}\sum_{j\in\mathcal{J}} w_j\Big(\underline{\frac{M-1}{M}\max_{m\in\mathcal{M}}\mathbb{E}[p_{jm}]} + \max_{m\in\mathcal{M}}\frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}\Big).$$

- Two terms in the performance gap:
  - First: loss due to using convex relaxation (vs. fully optimal static routing)

Performance result

The performance V^{R} of the static routing policy satisfies:

$$V^* \le V^{\mathrm{R}} \le V^* + \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \Big(\underline{\frac{M-1}{M} \max_{m \in \mathcal{M}} \mathbb{E}[p_{jm}]} + \underline{\max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}} \Big).$$

- Two terms in the performance gap:
  - First: loss due to using convex relaxation (vs. fully optimal static routing)
  - Second: loss due to using a static policy (vs. being adaptive)

Performance result

The performance V^{R} of the static routing policy satisfies:

$$V^* \le V^{\mathrm{R}} \le V^* + \frac{1}{2}\sum_{j\in\mathcal{J}} w_j\Big(\frac{M-1}{M}\max_{m\in\mathcal{M}}\mathbb{E}[p_{jm}] + \max_{m\in\mathcal{M}}\frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]}\Big).$$

- Two terms in the performance gap:
  - First: loss due to using convex relaxation (vs. fully optimal static routing)
  - Second: loss due to using a static policy (vs. being adaptive)

- **Corollary.** If weights, mean processing times, and the coefficients of variation of p_{jm} are uniformly bounded, and if $M/J \xrightarrow{J\to\infty} 0$, then

$$\frac{V^{\mathrm{R}} - V^*}{V^*} \xrightarrow{J\to\infty} 0.$$

Performance result

The performance V^{R} of the static routing policy satisfies:

$$V^* \leq V^{\mathrm{R}} \leq V^* + \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \Big(\frac{M-1}{M} \max_{m \in \mathcal{M}} \mathbb{E}[p_{jm}] + \max_{m \in \mathcal{M}} \frac{\mathrm{Var}[p_{jm}]}{\mathbb{E}[p_{jm}]} \Big).$$

- Two terms in the performance gap:
  - First: loss due to using convex relaxation (vs. fully optimal static routing)
  - Second: loss due to using a static policy (vs. being adaptive)

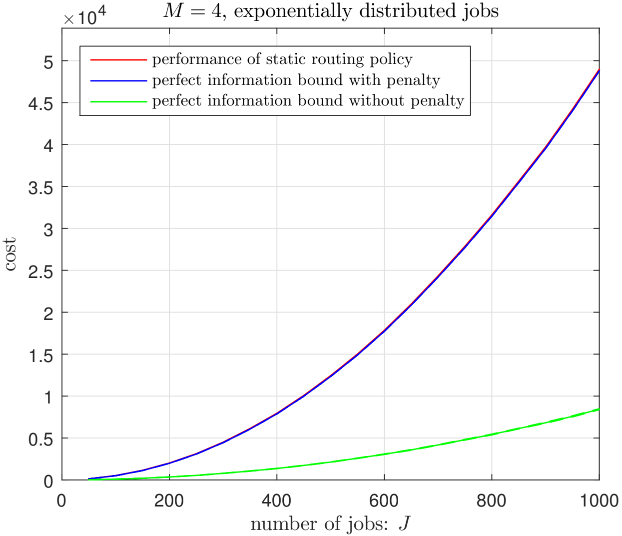
- **Corollary.** If weights, mean processing times, and the coefficients of variation of p_{jm} are uniformly bounded, and if $M/J \xrightarrow{J \to \infty} 0$, then

$$\frac{V^{\mathrm{R}} - V^*}{V^*} \xrightarrow{J \to \infty} 0.$$

In the regime with many jobs compared to machines, static routing is optimal!

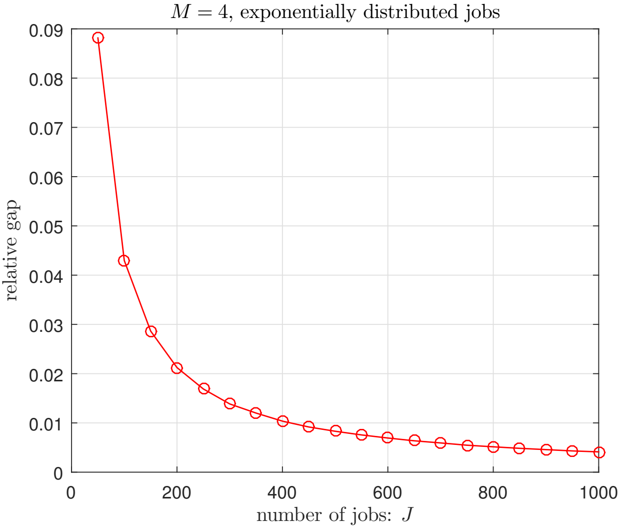
Examples

Weights and mean processing times generated randomly and fixed M:



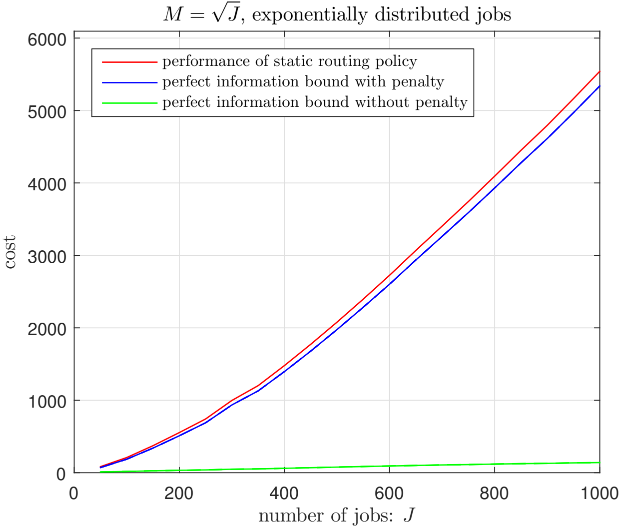
Examples

Relative gaps to penalized perfect information bounds, fixed M examples:



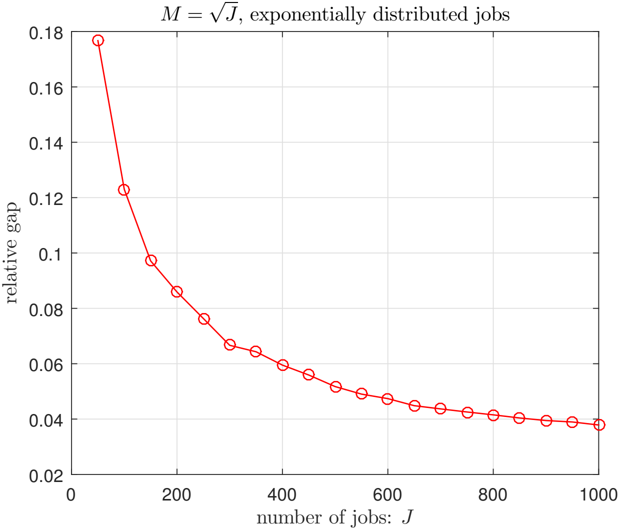
Examples

Weights and mean processing times generated randomly and $M = O(\sqrt{J})$:



Examples

Relative gaps to penalized perfect information bounds, scaled M examples:



Summary

- We study a class of **static routing policies** that can be easily computed by solving a **convex quadratic program** that only depends on job processing distributions through their expected values.

Summary

- We study a class of **static routing policies** that can be easily computed by solving a **convex quadratic program** that only depends on job processing distributions through their expected values.

- Via an **information relaxation duality** approach, we provide a uniform additive bound on the performance loss of this policy compared to the performance of an optimal non-anticipative scheduling policy.

Summary

- We study a class of **static routing policies** that can be easily computed by solving a **convex quadratic program** that only depends on job processing distributions through their expected values.

- Via an **information relaxation duality** approach, we provide a uniform additive bound on the performance loss of this policy compared to the performance of an optimal non-anticipative scheduling policy.

- The additive bound implies the static routing policy approaches optimality in the **asymptotic** regime of many jobs.

Summary

- We study a class of **static routing policies** that can be easily computed by solving a **convex quadratic program** that only depends on job processing distributions through their expected values.
- Via an **information relaxation duality** approach, we provide a uniform additive bound on the performance loss of this policy compared to the performance of an optimal non-anticipative scheduling policy.
- The additive bound implies the static routing policy approaches optimality in the **asymptotic** regime of many jobs.
- Other results:
  - uniformly related machines: route jobs in proportion to machine speeds is asymptotic optimal: $x_{jm} = s_m / \sum_{m' \in \mathcal{M}} s_{m'}$.
  - dependent jobs: static routing is close to optimality if job processing times are slightly correlated.

Thank you!

Other results: uniformly related machines

- Machines differ only in their speeds: $p_{jm} = p_j/s_m$: s_m the speed of machine m.

Other results: uniformly related machines

- Machines differ only in their speeds: $p_{jm} = p_j / s_m$: s_m the speed of machine m.
- Speed proportional routing: $x_{jm} = s_m / S$ with $S = \sum_{m' \in \mathcal{M}} s_{m'}$ (no optimization).

Other results: uniformly related machines

- Machines differ only in their speeds: $p_{jm} = p_j / s_m$: s_m the speed of machine m.
- Speed proportional routing: $x_{jm} = s_m / S$ with $S = \sum_{m' \in \mathcal{M}} s_{m'}$ (no optimization).

The performance V^{S} of the speed proportional routing satisfies:

$$V^* \leq V^{\mathrm{S}} \leq V^* + \frac{1}{2} \sum_{j \in \mathcal{J}} w_j \mathbb{E}[p_j] \Big\{ \Big(\frac{2M-1}{S} - \frac{1}{\kappa_j} \Big) + \Big(\frac{1}{\kappa_j} - \frac{1}{S} \Big) \frac{\mathrm{Var}[p_j]}{\mathbb{E}[p_j]^2} \Big\},$$

where $\kappa_j = s_M$ *if* $\mathrm{Var}[p_j]/\mathbb{E}[p_j]^2 > 1$ *and* $\kappa_j = s_1$ *if* $\mathrm{Var}[p_j]/\mathbb{E}[p_j]^2 \leq 1$.

Other results: dependent jobs

- Assume there exist constants α_j and β_j such that

$$\left|\mathbb{E}[p_{jm}|\mathcal{F}_t^{\pi}] - \mathbb{E}[p_{jm}]\right| \le \alpha_j,$$
$$\left|\mathbb{E}[p_{jm}^2|\mathcal{F}_t^{\pi}] - \mathbb{E}[p_{jm}^2]\right| \le \beta_j,$$

hold a.s. for every $j \in \mathcal{J}$, $m \in \mathcal{M}$, for all $\pi \in \Pi$, and for all $t \le S_{jm}^{\pi}$.

Other results: dependent jobs

- Assume there exist constants α_j and β_j such that

$$\left|\mathbb{E}[p_{jm}|\mathcal{F}_t^\pi] - \mathbb{E}[p_{jm}]\right| \le \alpha_j,$$
$$\left|\mathbb{E}[p_{jm}^2|\mathcal{F}_t^\pi] - \mathbb{E}[p_{jm}^2]\right| \le \beta_j,$$

  hold a.s. for every $j \in \mathcal{J}$, $m \in \mathcal{M}$, for all $\pi \in \Pi$, and for all $t \le S_{jm}^\pi$.

- **Corollary.** Suppose parameters are uniformly bounded. Let $\bar{\alpha} \triangleq \sum_{j\in\mathcal{J}} \alpha_j/J$ and $\bar{\beta} \triangleq \sum_{j\in\mathcal{J}} \beta_j/J$. If $\bar{\alpha}M^{\frac{3}{2}} \xrightarrow{J\to\infty} 0$, $\bar{\beta}M/J \xrightarrow{J\to\infty} 0$, and $M/J \xrightarrow{J\to\infty} 0$, then

$$\frac{V^{\mathrm{R}} - V^*}{V^*} \xrightarrow{J\to\infty} 0.$$

Other results: dependent jobs

- Assume there exist constants α_j and β_j such that

$$\left|\mathbb{E}[p_{jm}|\mathcal{F}_t^\pi] - \mathbb{E}[p_{jm}]\right| \leq \alpha_j,$$
$$\left|\mathbb{E}[p_{jm}^2|\mathcal{F}_t^\pi] - \mathbb{E}[p_{jm}^2]\right| \leq \beta_j,$$

  hold a.s. for every $j \in \mathcal{J}$, $m \in \mathcal{M}$, for all $\pi \in \Pi$, and for all $t \leq S_{jm}^\pi$.

- **Corollary.** Suppose parameters are uniformly bounded. Let $\bar{\alpha} \triangleq \sum_{j\in\mathcal{J}} \alpha_j/J$ and $\bar{\beta} \triangleq \sum_{j\in\mathcal{J}} \beta_j/J$. If $\bar{\alpha}M^{\frac{3}{2}} \xrightarrow{J\to\infty} 0$, $\bar{\beta}M/J \xrightarrow{J\to\infty} 0$, and $M/J \xrightarrow{J\to\infty} 0$, then

$$\frac{V^{\mathrm{R}} - V^*}{V^*} \xrightarrow{J\to\infty} 0.$$

- Assumption on $\bar{\alpha}$ is necessary (even with a fixed number of machines) for any static routing policy to be asymptotically optimal.