Nudge: Stochastically Improving Upon FCFS

Isaac Grosof – CMU
Kunhe Yang – Tsinghua University
Ziv Scully – CMU
Mor Harchol-Balter – CMU
M/G/1 Scheduling

Poisson arrivals, arrival rate: $\lambda$

General job size distribution: $S$, i.i.d.

Response time: $T$

Q: How should we schedule?
Baseline: First-Come First-Served

FCFS: Simple, practical, good theoretical properties

FCFS is weakly optimal:

\[ P(T_{FCFS} > t) \sim C e^{-\theta t}, \]

best possible \( \theta \).

Holds whenever \( S \) is light-tailed.

Setting: \( S \sim \) Hyperexponential with \( C^2 = 3, \rho = 0.4 \)
FCFS versus SRPT

FCFS: Large or small, similar response time
FCFS versus SRPT

FCFS: Large or small, similar response time
FCFS versus SRPT

FCFS: Large or small, similar response time
FCFS versus SRPT

FCFS: Large or small, similar response time
FCFS versus SRPT

FCFS: Large or small, similar response time
SRPT:
• Helps small jobs, better $P(T>t)$ for small $t$.
• Delays large jobs, worse $P(T>t)$ for large $t$. 
Fundamental question of tradeoffs

All previous policies have tradeoffs:
Better than FCFS at small t, worse than FCFS at large t.
Is that inevitable?

Is it possible to beat FCFS everywhere? ($\forall t$)

Yes, with Nudge!
Nudge: Stochastic Improvement

We introduce a new policy: Nudge

We prove:
\[ P(T^{Nudge} > t) < P(T^{FCFS} > t) \] \( \forall t \), for all light-tailed job size distributions \( S \).

We prove Nudge has better asymptotic tail:
\[ P(T^{Nudge} > t) \sim C' e^{-\theta t} \]. Same \( \theta \), better \( C' \).

Previously conjectured to be impossible [WZ '12]

Setting: \( S \sim \text{Hyperexponential with } C^2 = 3, \rho = 0.4 \)

12% improvement
Our contribution: Nudge

- Default: FCFS
- Classify jobs as small or large by size
- When small arrives, if large is last in queue, small nudges ahead of large

Job size distribution
Our contribution: Nudge

- Default: FCFS
- Classify jobs as small or large by size
- When small arrives, if large is last in queue, small nudges ahead of large
- Large can only be nudged once.
Our contribution: Nudge

- Default: FCFS
- Classify jobs as small or large by size
- When small arrives, if large is last in queue, small nudges ahead of large
- Large can only be nudged once.
Nudge policy: Jack gets to go ahead of the giant,
Nudge intuition: Jack and the Giant

Nudge policy: Jack gets to go ahead of the giant, but nobody else gets to.
Proof intuition

Want to show:
\[ P(T^{Nudge} > t) < P(T^{FCFS} > t) \forall t \]

# Jobs with \( T > t \)

Threshold \( t \)

\( T^{FCFS}_{Small} \) \( T^{FCFS}_{Large} \)
Want to show:

\[ P(T_{Nudge} > t) < P(T_{FCFS} > t) \forall t \]
Proof intuition: One t, many nudges

Want to show: Rate of improve exceeds degrade.
• Key idea 1: Rates of improve and degrade determined by FCFS tail and pdf.
• Key idea 2: Bound FCFS tail and pdf relative to limiting exponential.
• Bounds imply more improve than degrade, given correct small and large cutoffs.
Empirical results

Tail improvement ratio:
$$1 - \frac{P(T^{Nudge} > t)}{P(T^{FCFS} > t)}$$

Threshold $t$

Load $\rho=0.8$

- Bounded Lomax
- Exponential
- Hyperexponential
- Uniform
Future directions

• What about 2+ swaps per job? Constant # is important.
• What range of size cutoffs work?
  • Single threshold, all jobs either large or small?
• Beyond FCFS, what other policies can be improved everywhere?
Conclusion

Introduce policy called Nudge.

First policy to achieve stochastic improvement over FCFS, for any light-tailed job size distribution.

First policy to achieve multiplicative asymptotic improvement over FCFS.