Scheduling in the Dark

Jeff Edmonds
York University

Non-clairvoyant Multiprocessor

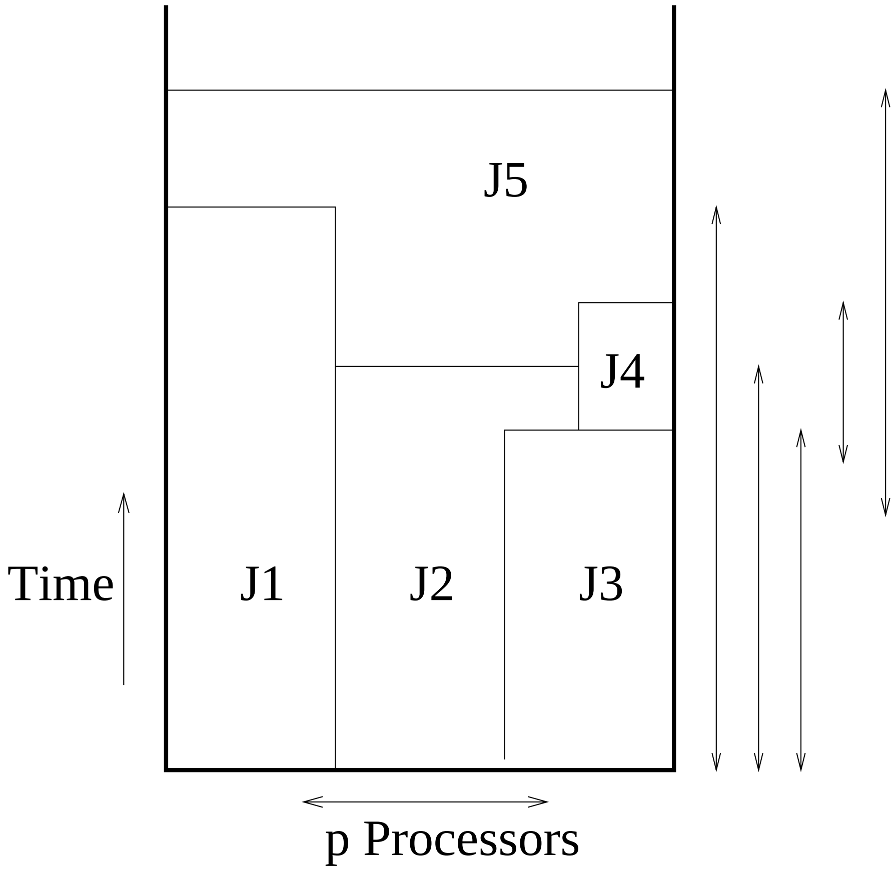
Scheduling of Jobs

with Arbitrary Arrival Times

and Changing Execution Characteristics

The Scheduling Problem

- Allocate p processors to a stream of n jobs



- Average Response Time:

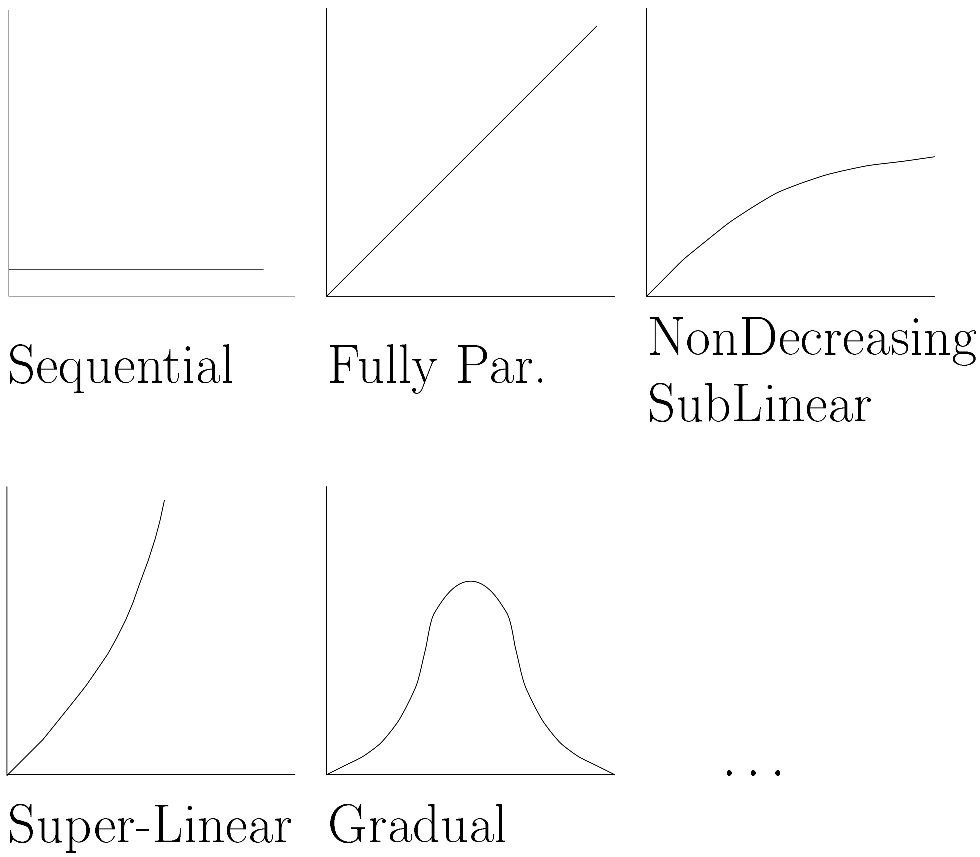
$$AvgResp(S(J)) = \frac{1}{n} \sum_{i \in [1..n]} c_i - r_i$$

- Competitive Ratio:

$$\text{Min}_{S \in \mathcal{S}} \text{Max}_{J \in \mathcal{J}} \frac{AvgResp(S(J))}{AvgResp(OPT(J))}$$

Different Classes $\mathcal{J}$ of Job Sets J

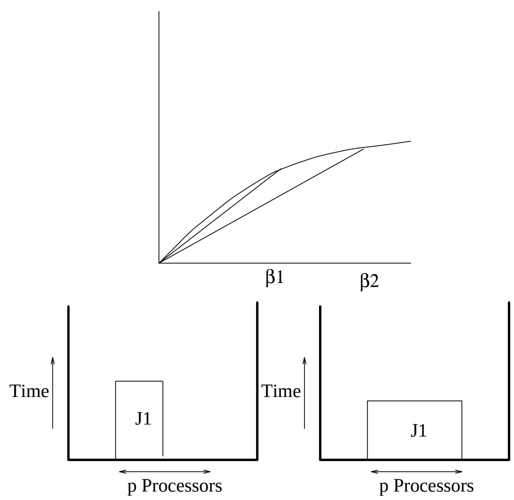
- Arrival Times (Arbitrary or Batch)
- Some Class of Speedup Functions
  - $\Gamma(\beta)$ is the rate (work/time) when allocated β processors.



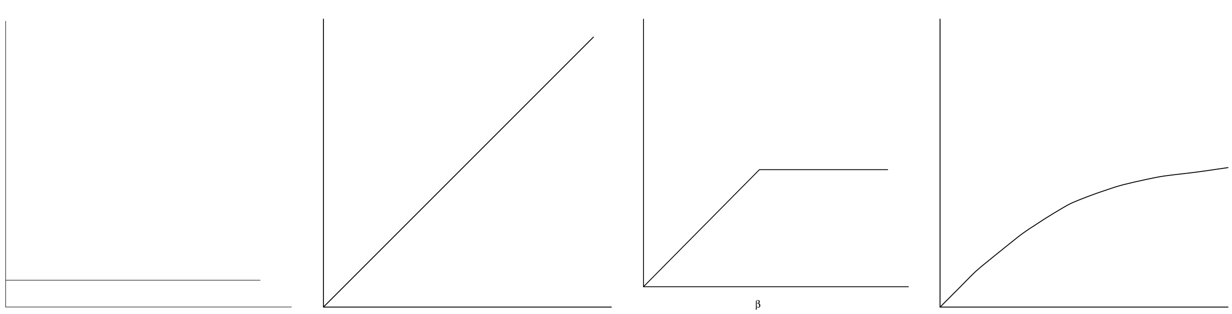
- # of Phases in a Job (Single or Arbitrary)

SubLinear-NonDecreasing Speedup Functions

- A set of jobs $J = \{J_1, J_2, \ldots, J_n\}$
- Each job has phases $J_i = \left\langle J_i^1, J_i^2, \ldots, J_i^{q_i} \right\rangle$
- Each job phase $J_i^q = \langle W_i^q, \Gamma_i^q \rangle$ is defined by
  - W_i^q is the amount of *work*
  - $\Gamma_i^q(\beta)$ is the rate (work/time) with β processors
- Speedup functions must be:
  - NonDecreasing: $\beta_1 \leq \beta_2 \Rightarrow \Gamma(\beta_1) \leq \Gamma(\beta_2)$.
  - SubLinear: $\beta_1 \leq \beta_2 \Rightarrow \Gamma(\beta_1)/\beta_1 \geq \Gamma(\beta_2)/\beta_2$.

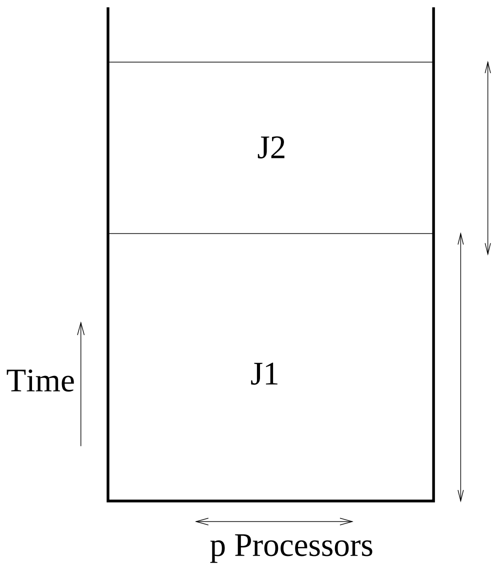


- Examples

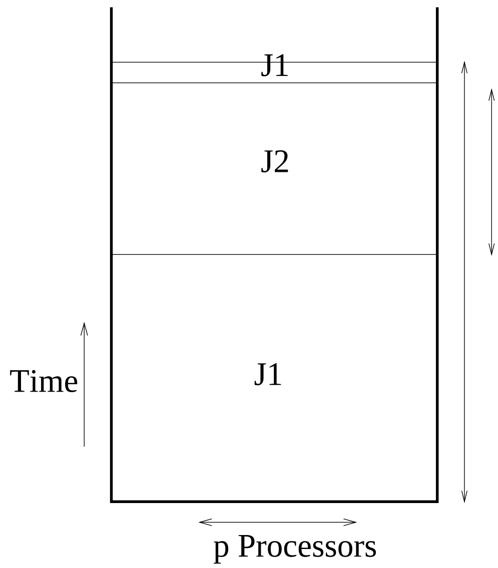


Sequential Fully Par. Fully to β

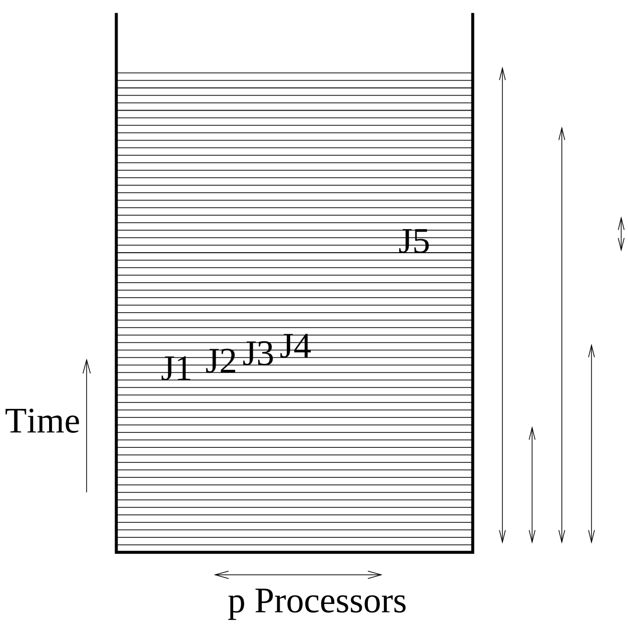
Examples of Schedulers (algorithms)



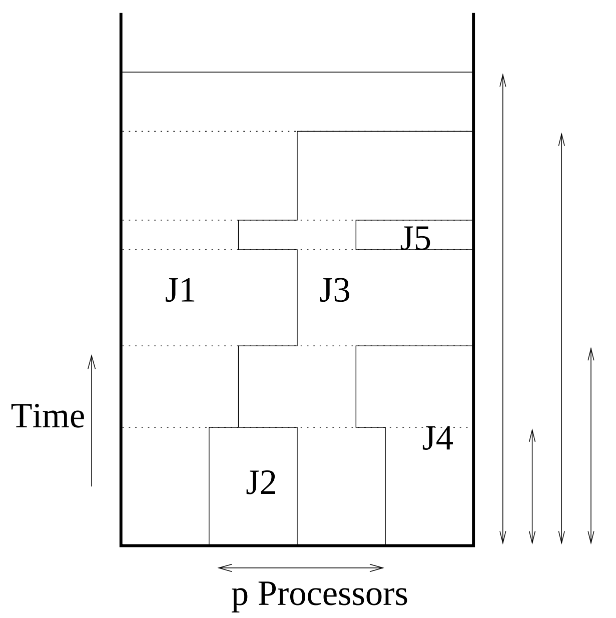
Shortest Remaining Work First (SJF)



Balance (BAL) Shortest Run First



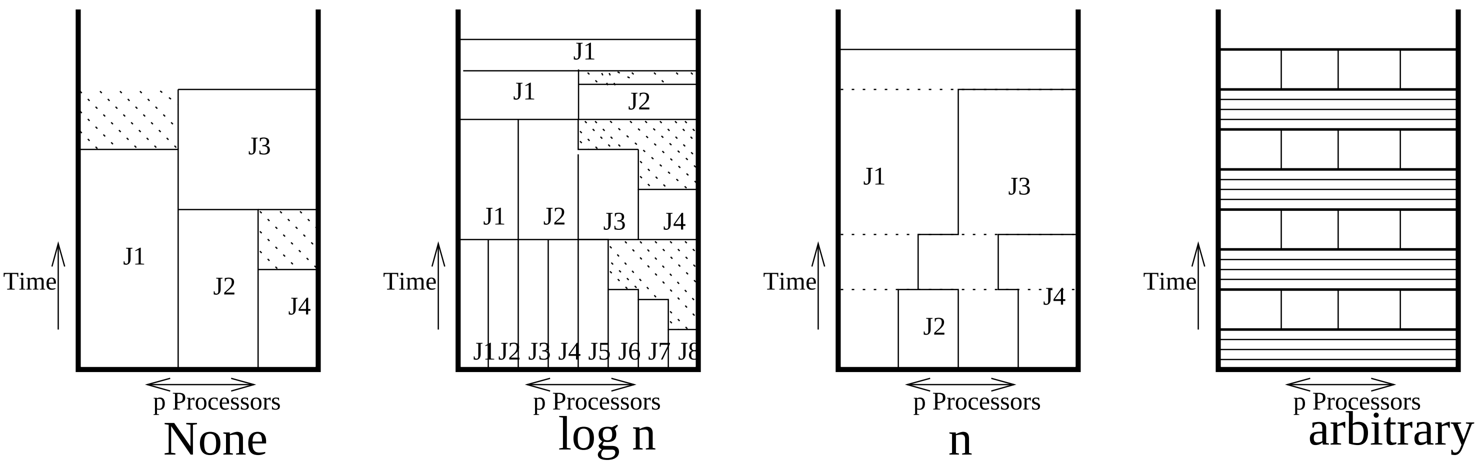
Round Robin (RR)



Equal-Partition (EQUI)

Different Classes $\mathcal{S}$ of Schedulers S

- Clairvoyance
  - No, partial, or complete knowledge
- Computation Time
  - Unbounded, Poly Time, or Reasonable
- # of Preemptions (re-allocation of processors)



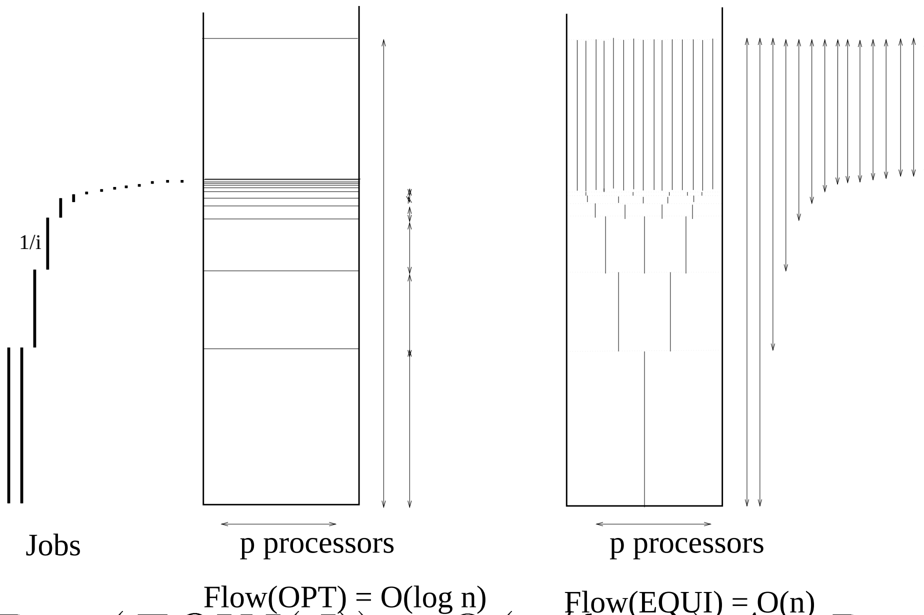
The Optimal Scheduler

- Unbounded
  - Clairvoyance
  - Computation Time
  - Preemptions

Devil and one player

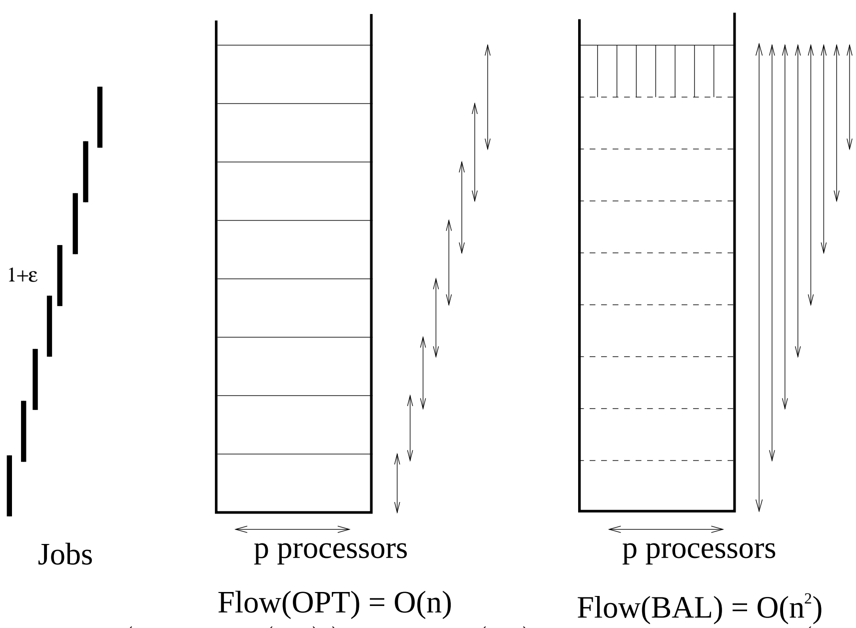
Lower Bounds

- Equal-Partition



$$AvgResp(EQUI(J)) \geq \Omega\left(n/\log n\right) \cdot AvgResp(OPT(J))$$

- Balance



$$AvgResp(BAL(J)) \geq \Omega\left(n\right) \cdot AvgResp(OPT(J))$$

- General Non-Clairvoyant Schedulers S

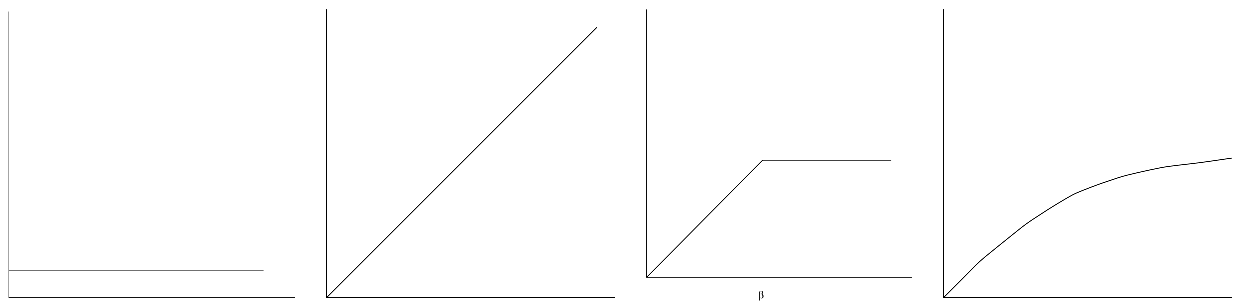
$$AvgResp(S(J)) \geq \Omega\left(\sqrt{n}\right) \cdot AvgResp(OPT(J))$$

Devil $2 + \epsilon$.

Main Result

For any set of jobs J with

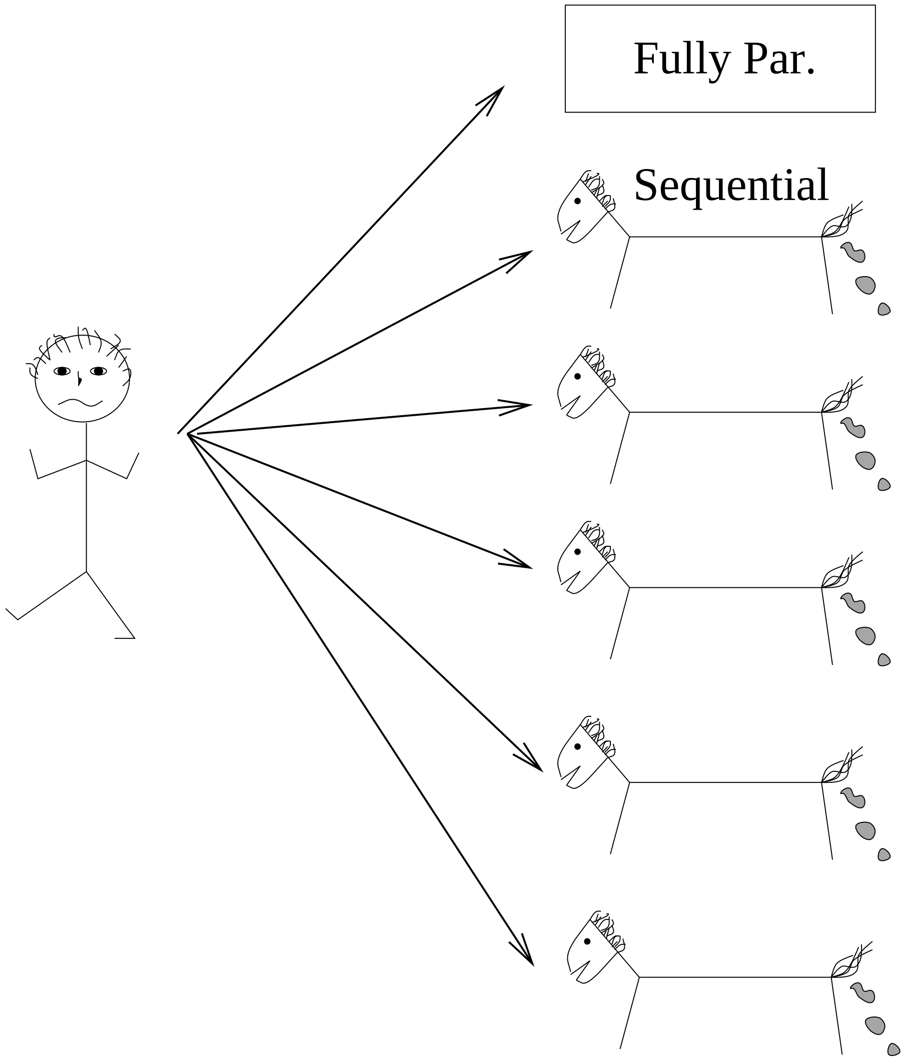
- arbitrary arrival times
- arbitrary number of phases
- sublinear-nondecreasing speedup functions



Sequential   Fully Par.   Fully to β

$$\frac{AvgResp(EQUI_{2+\epsilon}(J))}{AvgResp(OPT(J))} \leq \mathcal{O}\left(1 + \frac{1}{\epsilon}\right)$$

Wasting Resources on Sequential Jobs

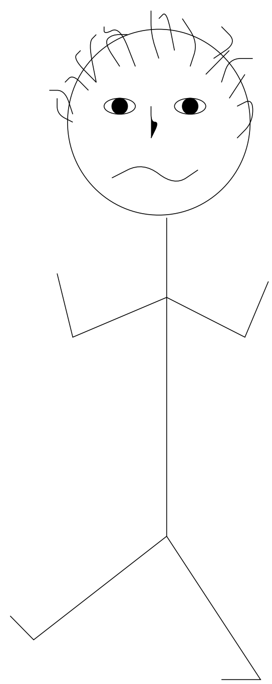


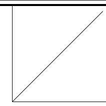
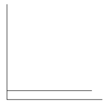
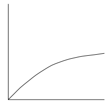
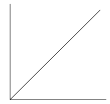
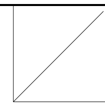
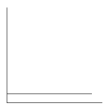
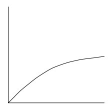
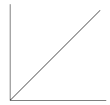
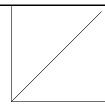
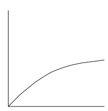
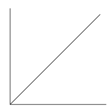
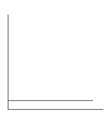
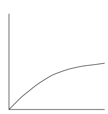
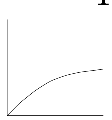
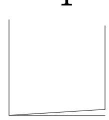
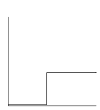
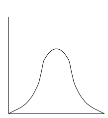
At most $\frac{1}{2+\epsilon}$ of our resources are wasted on sequential jobs.

Designing an Operating System

- Predict the future.
- How much work in job?
- Fully par. or seq.?
- Design & code better algs.
- Spend more cpu time.

- Buy $2+\epsilon$ times as many processors.
- Run $EQUI$.

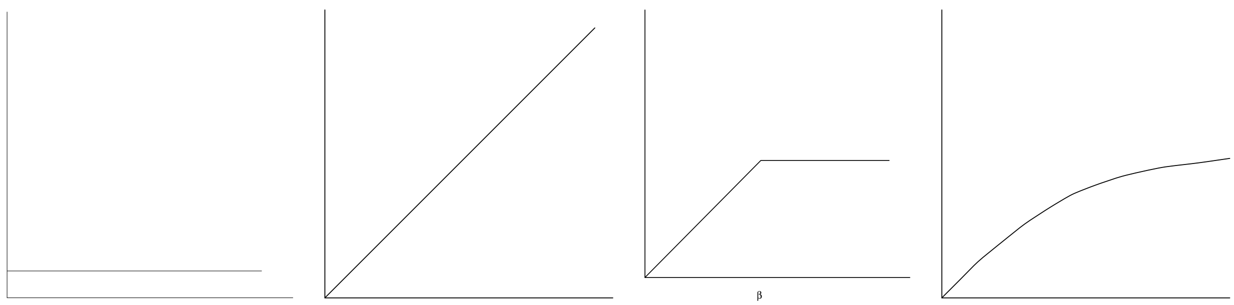


Who	Sched	Jobs	s	comp
[MPT]	$EQUI$	Batch 	1	2
[ECBD]		, , or 	1	$[2.71, 3.74]$
[KP]	BAL	Arb. Arr. 	1	$\Omega(n)$
			$1+\epsilon$	$1+\frac{1}{\epsilon}$
[BC]			$s \geq 2$	$\frac{2}{s}$
new		, , or 	s	$\Omega(n)$
[MPT]	$EQUI$		1	$\Omega(\frac{n}{\log n})$
[KP]			$1+\epsilon$	$\Omega(n^{1-\epsilon})$
new		, , or 	$2+\epsilon$	$[1+\frac{1}{\epsilon}, 2+\frac{4}{\epsilon}]$
			s	≥ 1
				$\Theta(\frac{1}{s})$
		 or 	1	$\Theta(1)$
new	$EQUI'$	few preempt	$4+\epsilon$	$\Theta(1)$
	$HEQUI$	 or 	$4+\epsilon$	$\Theta(1)$
	$HEQUI'$	 or 	$\Theta(\log p)$	$\Theta(1)$

Main Result

For any set of jobs J with

- arbitrary arrival times
- arbitrary number of phases
- sublinear-nondecreasing speedup functions



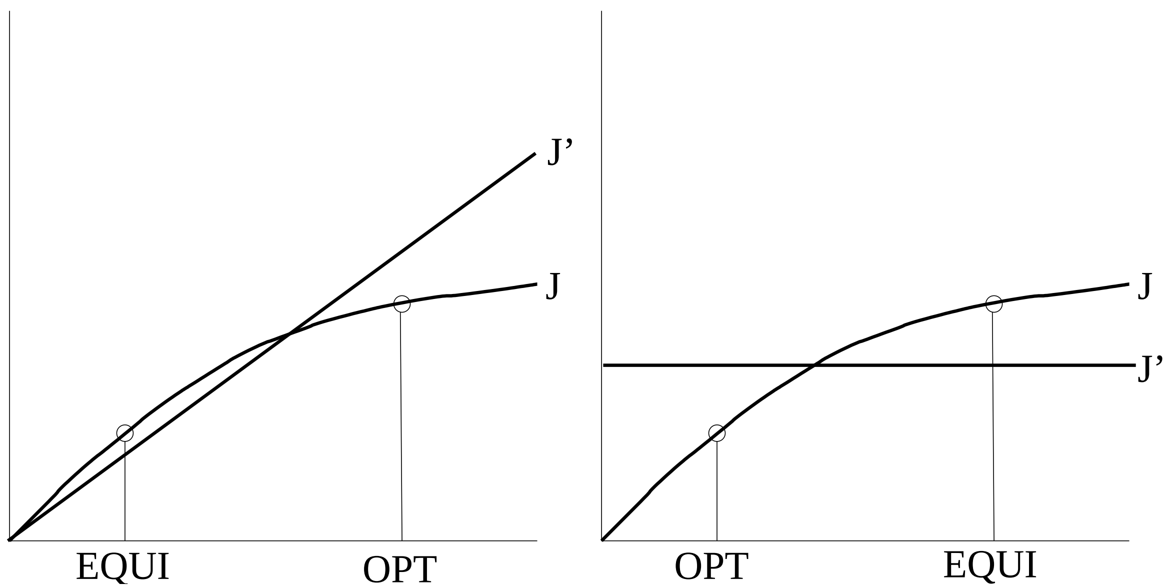
Sequential Fully Par. Fully to β

$$\frac{AvgResp(EQUI_{2+\epsilon}(J))}{AvgResp(OPT(J))} \leq \mathcal{O}\left(1 + \frac{1}{\epsilon}\right)$$

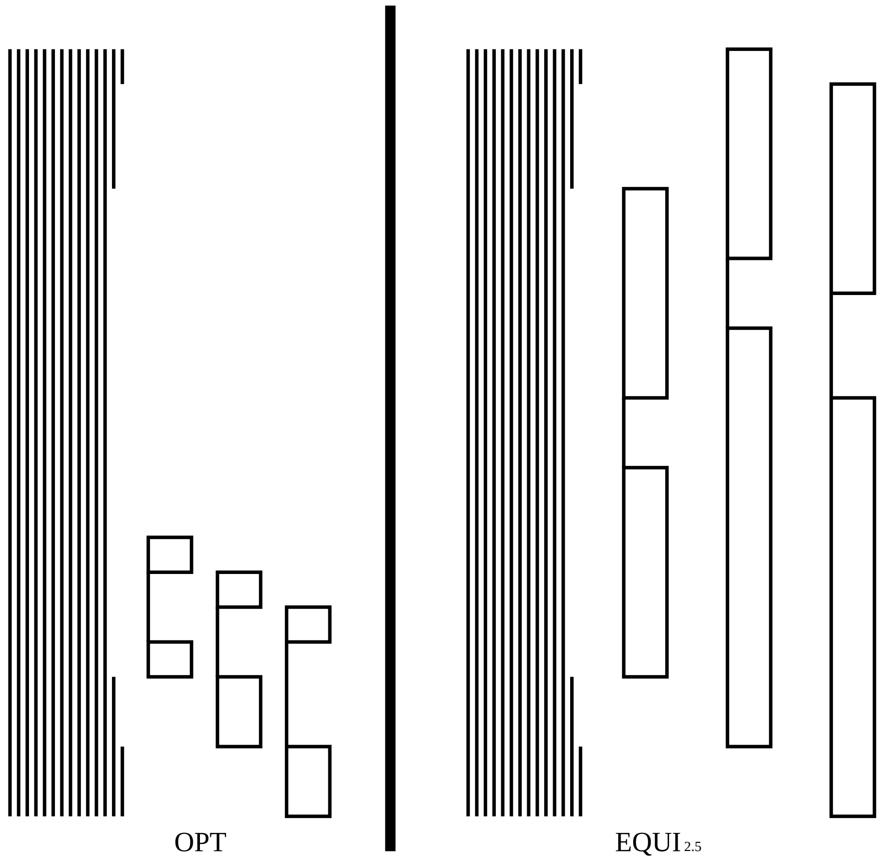
Worst Case J

In the worst case set of jobs J
each phase is either fully parallelizable or sequential.

$$\forall J\ \exists J'\ \frac{AvgResp(EQUI_{2+\epsilon}(J))}{AvgResp(OPT(J))} \leq \frac{AvgResp(EQUI_{2+\epsilon}(J'))}{AvgResp(OPT(J'))}$$



Integrating Through Time



$$\frac{AvgResp(EQUI_{2+\epsilon}(J'))}{AvgResp(OPT(J'))}$$

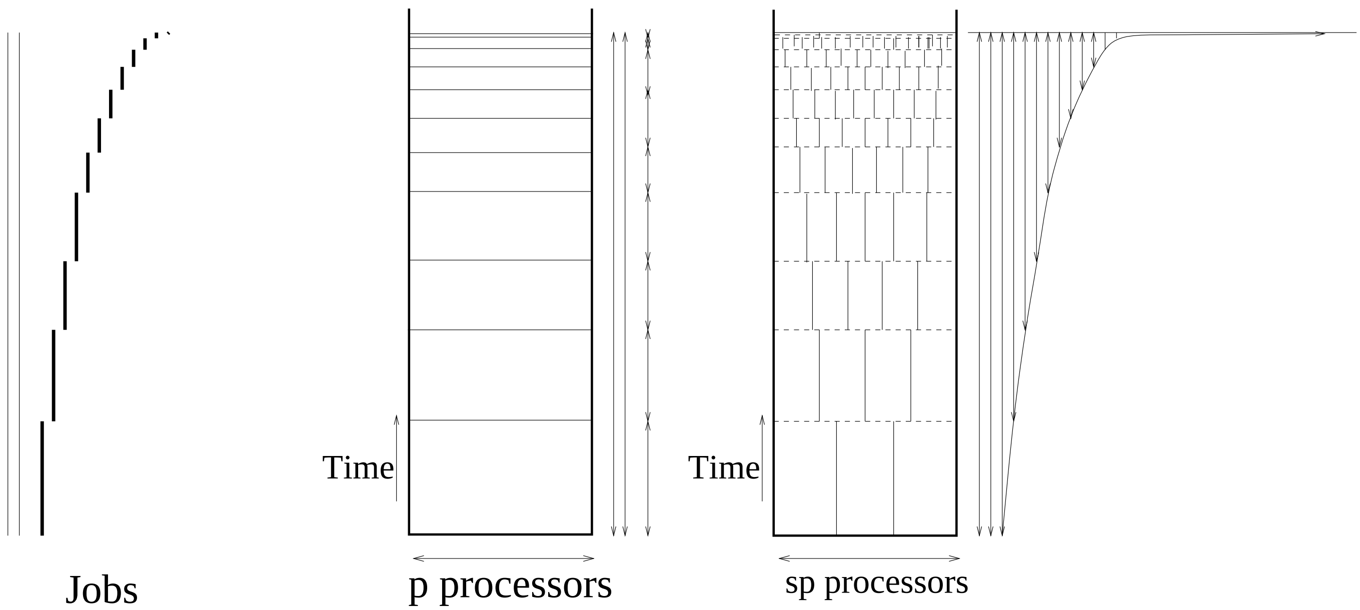
$$= \frac{\int_0^\infty (\#\text{ par. EQUI})_t + (\#\text{ seq. EQUI})_t \delta t}{\int_0^\infty \quad 1 \quad + (\#\text{ seq. OPT})_t \delta t}$$

$$= \frac{\int_0^\infty (\#\text{ par. EQUI})_t + (\#\text{ seq. EQUI})_t \delta t}{\int_0^\infty \quad 1 \quad + (\#\text{ seq. EQUI})_t \delta t}$$

Extra Resources $s = 2 + \epsilon$

Still Number of Jobs Alive

is Unbounded



Flow(OPT) = O(1) Flow($EQUI_{2+\varepsilon}$) = O(1)

Steady State

Potential Function

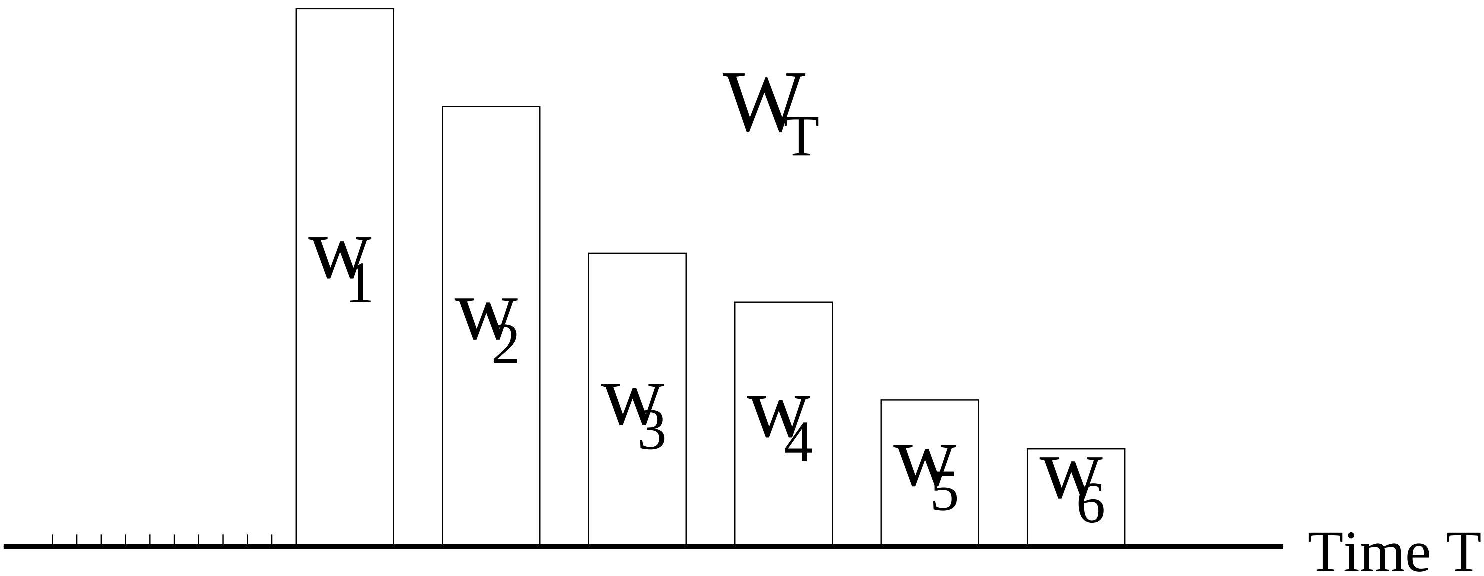
- $W_t =$ set of work completed by OPT but not by $EQUI$.
- $F(W_t)$ = a measure of the work
- $(\#\text{ par. EQUI})_t \geq \frac{1}{\epsilon}(\#\text{ seq. EQUI})_t$
  $\Rightarrow F(W_t)$ decreases with time

- $F_T = F(W_T) + \int_0^T (\#\text{ par. EQUI})_t - \frac{1}{\epsilon}(\#\text{ seq. EQUI})_t \delta t$
- $F_0 = 0$
- $\frac{\delta F_T}{\delta T} \leq 0$
- $F_\infty \leq 0$
- $\int_0^\infty (\#\text{ par. EQUI})_t - \frac{1}{\epsilon}(\#\text{ seq. EQUI})_t \delta t \leq 0.$

- $\frac{AvgResp(EQUI_{2+\epsilon}(J))}{AvgResp(OPT(J))}$
  $\leq \frac{\int_0^\infty (\#\text{ par. EQUI})_t + (\#\text{ seq. EQUI})_t \delta t}{\int_0^\infty 1 + (\#\text{ seq. EQUI})_t \delta t} \leq \mathcal{O}(1 + \frac{1}{\epsilon})$

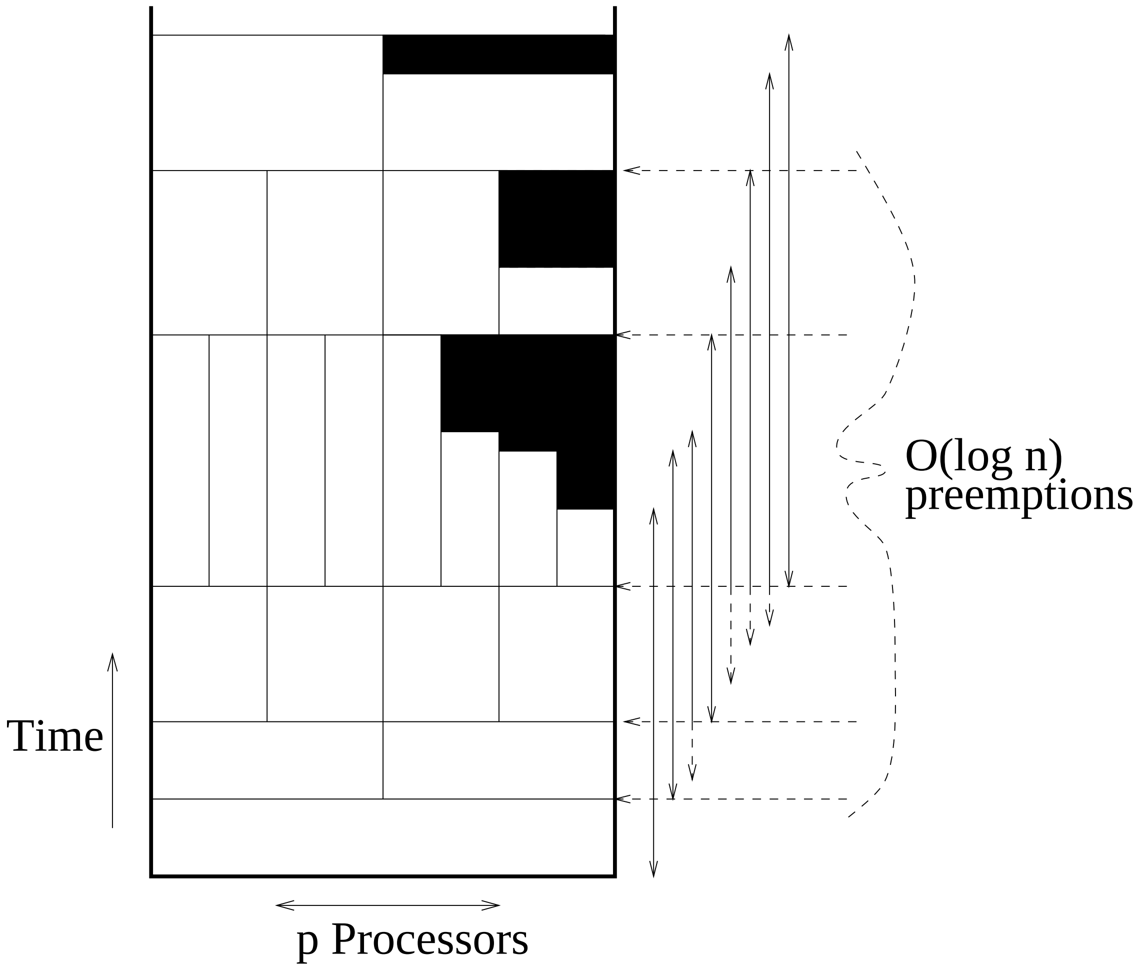
All Jobs Fully Parallelizable or Sequential

Work Completed by OPT and not by $EQUI$



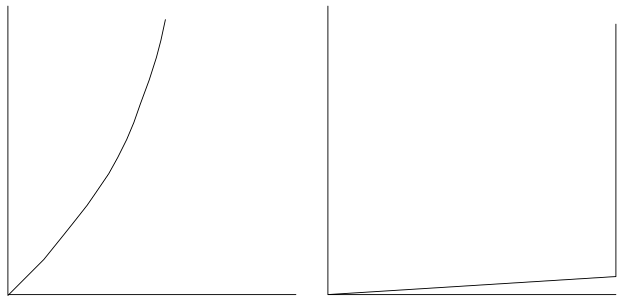
$$
\begin{aligned}
F_T &= \textstyle\int_0^T (m_t - \frac{\ell_t}{\epsilon})\delta t \; + F(W_T) \\
&= \textstyle\int_0^T (m_t - \frac{\ell_t}{\epsilon}) \quad + \frac{2}{\epsilon} \Sigma_{i=1}^{m_T} i w_i \\
\frac{\delta F_T}{\delta T} &= (m_T - \frac{\ell_T}{\epsilon}) \quad + \frac{2}{\epsilon} \Big[\; (m_T \cdot 1) - \Sigma_{i=1}^{m_T} i \cdot \left(\frac{2+\epsilon}{m_T + \ell_T}\right) \Big] \\
& \qquad \left(\frac{(m_T)^2}{2}\right) \cdot \left(\frac{2+\epsilon}{m_T+\ell_T}\right) \\
& \qquad m_T + \frac{\epsilon}{2} m_T - \mathcal{O}(\ell_T) \\
&\leq 0
\end{aligned}
$$

"$log n$" Preemptions

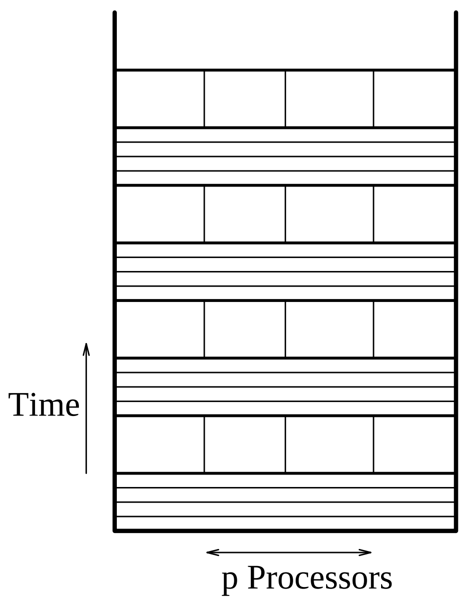


- Preempts only when # jobs increases or decreases by a factor of 2.
- Competitive with $s = 8 + \epsilon$.

Super Linear Speedup Functions

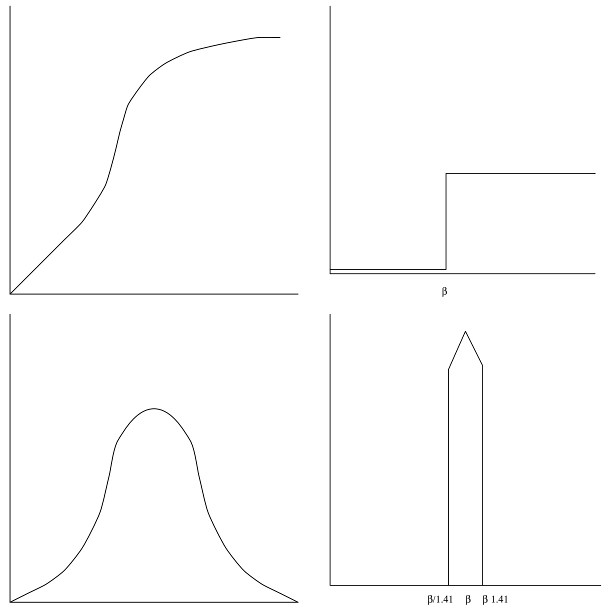


- Time-Space Tradeoff and Highly Parallelizable
- Competitive with $s = 4 + \epsilon$.
  - *Round Robin* (super linear phases)
  - $EQUI$ (sub-linear phases)



- Bounded preemptions $\Rightarrow \Omega(n)$

NonDecreasing or "Gradual" Speedup Functions



- Competitive with $s = \mathcal{O}(\log p)$:
  - Run each job
    - for a slice of time
    - with 2^k processors ($\forall k \in [1, \log p]$)
  - Guaranteed to run each job phase
    - with the "right" # of processors

Conjectures

- Are the $2 + \epsilon$ extra resources needed?

$$\forall \epsilon > 0, \exists \text{ a Non-Clairvoyant Scheduler } S$$

$$\forall J \ \frac{AvgResp(S_{1+\epsilon}(J))}{AvgResp(OPT(J))} \leq \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$$

- Jobs arrive in a Random order

$$\frac{AvgResp(EQUI_1(J))}{AvgResp(OPT(J))} \leq \mathcal{O}\left(1\right)$$

- Lower bound for Non-clairvoyant Schedulers.