Scheduling on a Channel with Failures and Retransmissions

Predrag R. Jelenković and Evangelia D. Skiani

Department of Electrical Engineering
Columbia University, NY 10027, USA
{predrag,valia}@ee.columbia.edu

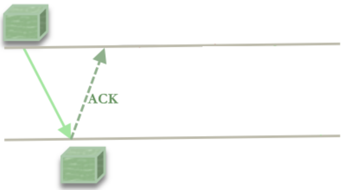
October 6, 2013

**Supported by NSF grant 0915784*

Outline

Failures & Retransmissions (Restarts)

- High variability ⇒ frequent failures
- Possible solution: Restart the system
- Applications
  - networking e.g. ARQ, HTTP
  - computing



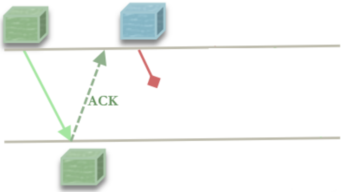
Restarts cause **power law** delays & possibly zero throughput, even for superexponential files [ALSF'05-, JT'06-]:

$$\mathbb{P}[N > n] \sim \Gamma(\alpha+1)/n^{\alpha} \qquad (1)$$

What is the best job scheduling policy?

Failures & Retransmissions (Restarts)

- High variability ⇒ frequent failures
- Possible solution: Restart the system
- Applications
  - networking e.g. ARQ, HTTP
  - computing



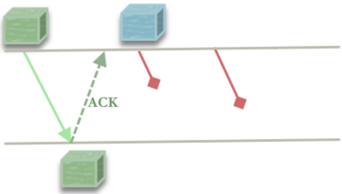
Restarts cause **power law** delays & possibly zero throughput, even for superexponential files [ALSF'05-, JT'06-]:

$$\mathbb{P}[N > n] \sim \Gamma(\alpha+1)/n^{\alpha} \quad (1)$$

What is the best job scheduling policy?

Failures & Retransmissions (Restarts)

- High variability ⇒ frequent failures
- Possible solution: Restart the system
- Applications
  - networking e.g. ARQ, HTTP
  - computing



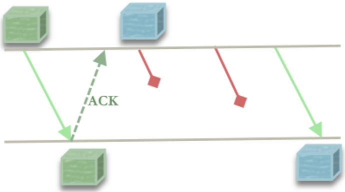
Restarts cause **power law** delays & possibly zero throughput, even for superexponential files [ALSF'05-, JT'06-]:

$$\mathbb{P}[N > n] \sim \Gamma(\alpha+1)/n^{\alpha} \tag{1}$$

What is the best job scheduling policy?

Failures & Retransmissions (Restarts)

- High variability ⇒ frequent failures
- Possible solution: Restart the system
- Applications
  - networking e.g. ARQ, HTTP
  - computing



Restarts cause **power law** delays & possibly zero throughput, even for superexponential files [ALSF'05-, JT'06-]:

$$\mathbb{P}[N > n] \sim \Gamma(\alpha+1)/n^{\alpha} \quad (1)$$

What is the best job scheduling policy?

Scheduling & Retransmissions

No known policies optimize the sojourn time tail across BOTH light and heavy-tailed job size distributions.

Optimality

- Subexponential jobs: PS, shortest remaining processing time [ANA'99]
- Superexponential jobs: First come first served [RS'01]

We study two scheduling policies:

1. First Come First Served (**FCFS**)
2. Processor Sharing (**PS**)

Question:

How do these policies work under retransmissions?

Scheduling & Retransmissions

No known policies optimize the sojourn time tail across BOTH light and heavy-tailed job size distributions.

Optimality

- Subexponential jobs: PS, shortest remaining processing time [ANA'99]
- Superexponential jobs: First come first served [RS'01]

We study two scheduling policies:

1. First Come First Served (**FCFS**)
2. Processor Sharing (**PS**)

Question:

How do these policies work under retransmissions?

Scheduling & Retransmissions

No known policies optimize the sojourn time tail across BOTH light and heavy-tailed job size distributions.

Optimality

- Subexponential jobs: PS, shortest remaining processing time [ANA'99]
- Superexponential jobs: First come first served [RS'01]

We study two scheduling policies:

1. First Come First Served (**FCFS**)
2. Processor Sharing (**PS**)

Question:

How do these policies work under retransmissions?

Model of Channel

- Available periods $\{A_n\}_{n\geq 1}$: i.i.d.
- Unit Capacity

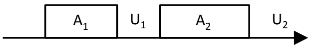


Figure: A failure-prone system.

Retransmission Model

- Generic job $B \in (0, \infty)$
- **if** $B \leq A_n$, success; **else**, retransmit at period A_{n+1}

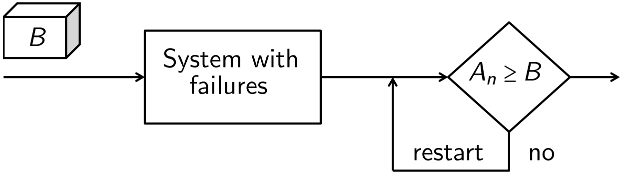


Figure: Jobs over a system with failures.

Definitions & Notation

Definition 1 (Service Time)

The service time is the total time until a job is successfully served and is denoted as

$$S := \sum_{i=1}^{N-1} A_i + B,$$

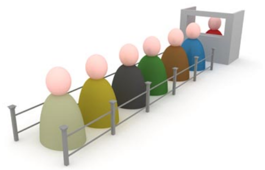
where N is the number of attempts until the successful completion of the job.

Denote the tail distributions of job sizes B and availability periods A as

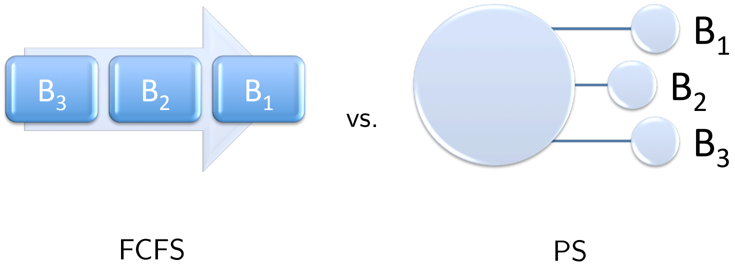
$$\bar{F}(x) = \mathbb{P}(B > x) \qquad \text{and} \qquad \bar{G}(x) = \mathbb{P}(A > x)$$

A Simple Scenario

- There are m jobs of size B_i, $i = 1 \ldots m$
- Each job requires S_i time units
- No future arrivals



Job Scheduling:



Definitions & Notation

Definition 2 (Total Completion Time)

The total completion time is defined as the total time until all the jobs in the queue are successfully served and is denoted as

$$\Theta_m := \sum_{i=1}^{m} S_i,$$

where m is the total number of jobs in the system and S_i's are the service times for each job.

Note: Total completion time **without** retransmissions → trivial!
⇒ Always equal to $\sum_{i=1}^{m} B_i$

First Come First Served (FCFS)

Theorem 1

If $\log \bar{F}(x) \approx \alpha \log \bar{G}(x)$ *for all* $x \geq 0$ *and* $\alpha > 0$, *and* $\mathbb{E}[A^{1+\theta}] < \infty$ *for some* $\theta > 0$, *then*

$$\lim_{t \to \infty} \frac{\log \mathbb{P}[\Theta_m > t]}{\log t} = -\alpha.$$

Proof [of Theorem 1].

Under the conditions of the Theorem, the result in [JT'06-] yields

$$\lim_{t \to \infty} \frac{\log \mathbb{P}[S > t]}{\log t} = -\alpha \qquad \text{as } t \to \infty, \qquad (\star)$$

where S is the service time of one job if served **alone**. □

First Come First Served (FCFS)

Theorem 1

If $\log \bar{F}(x) \approx \alpha \log \bar{G}(x)$ *for all* $x \geq 0$ *and* $\alpha > 0$, *and* $\mathbb{E}[A^{1+\theta}] < \infty$ *for some* $\theta > 0$, *then*

$$\lim_{t\to\infty} \frac{\log \mathbb{P}[\Theta_m > t]}{\log t} = -\alpha.$$

Proof [of Theorem 1].

Under the conditions of the Theorem, the result in [JT'06-] yields

$$\lim_{t\to\infty} \frac{\log \mathbb{P}[S > t]}{\log t} = -\alpha \qquad \text{as } t \to \infty, \qquad (\star)$$

where S is the service time of one job if served **alone**. □

FCFS

Proof [of Theorem 1].

The total completion time is **lower** bounded by a single job service time:

$$\mathbb{P}[\Theta_m > t] \geq \mathbb{P}[S_1 > t] \quad \overset{(\star)}{\Longrightarrow} \quad \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \lesssim \alpha.$$

Let $\bar{S}_i$ be the service time of a job i when we *idle* the server after job completion until next failure. Then, the **upper** bound is

$$\mathbb{P}[\Theta_m > t] \leq \mathbb{P}\left[\sum_{i=1}^{m} \bar{S}_i > t\right] \leq m\mathbb{P}\left[\bar{S}_1 > \frac{t}{m}\right]$$

$$\overset{(\star)}{\Longrightarrow} \quad \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \gtrsim \alpha.$$

FCFS

Proof [of Theorem 1].

The total completion time is **lower** bounded by a single job service time:

$$\mathbb{P}[\Theta_m > t] \geq \mathbb{P}[S_1 > t] \quad \stackrel{(\star)}{\Longrightarrow} \quad \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \lesssim \alpha.$$

Let $\bar{S}_i$ be the service time of a job i when we *idle* the server after job completion until next failure. Then, the **upper** bound is

$$\mathbb{P}[\Theta_m > t] \leq \mathbb{P}\left[\sum_{i=1}^{m} \bar{S}_i > t\right] \leq m\mathbb{P}\left[\bar{S}_1 > \frac{t}{m}\right]$$
$$\stackrel{(\star)}{\Longrightarrow} \quad \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \gtrsim \alpha.$$

□

Processor Sharing (PS)

Theorem 2

If the hazard function $-\log \bar{F}(x)$ *is regularly varying with index* $\gamma \geq 0$*, then, under the conditions of Theorem 1,*

i) *if* $\gamma \leq 1$*, i.e.* B *is subexponential or exponential, then*

$$\lim_{t\to\infty} \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} = \alpha,$$

ii) *if* $\gamma > 1$*, i.e.* B *is superexponential, then*

$$\lim_{t\to\infty} \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} = \frac{\alpha}{m^{\gamma-1}} < \alpha.$$

Processor Sharing (PS)

Theorem 2

If the hazard function $-\log \bar{F}(x)$ *is regularly varying with index* $\gamma \geq 0$*, then, under the conditions of Theorem 1,*

i) *if* $\gamma \leq 1$*, i.e. B is subexponential or exponential, then*

$$\lim_{t\to\infty} \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} = \alpha,$$

ii) *if* $\gamma > 1$*, i.e. B is superexponential, then*

$$\lim_{t\to\infty} \frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} = \frac{\alpha}{m^{\gamma-1}} < \alpha.$$

Idea of the proof (I)

The **upper** bound is

$$\mathbb{P}[\Theta_m > t] \leq \mathbb{P}\left[\sum_{i=1}^{m} \bar{S}_i > t\right] \leq (1+\varepsilon)\sum_{i=1}^{m}\mathbb{P}[\bar{S}_i > t].$$

1. If $\widehat{B}_1$ is the smallest job, then

$$\mathbb{P}[N_1 > n] = \mathbb{E}\mathbb{P}\left[\widehat{B}_1 > \frac{A}{m}\right]^n = \mathbb{E}\left(1 - \bar{G}(m\widehat{B}_1)\right)^n = \mathbb{E}\left(1 - \bar{F}_1(m\widehat{B}_1)^{\frac{1}{\alpha_1}}\right)^n$$

2. What is the relationship between $\bar{F}_1(x)$ and $\bar{G}(x)$?

$$\log \bar{F}_1(x) = \log \mathbb{P}[m\widehat{B}_1 > x] = \log\left(\bar{F}(x/m)\right)^m \approx m^{1-\gamma}\log \bar{F}(x).$$

3. Recalling $(\star)$,

$$\frac{-\log \mathbb{P}[\bar{S}_1 > t]}{\log t} \xrightarrow[t\to\infty]{} \frac{\alpha}{m^{\gamma-1}} \qquad (\star)$$

Idea of the proof (I)

The **upper** bound is

$$\mathbb{P}[\Theta_m > t] \leq \mathbb{P}\left[\sum_{i=1}^{m} \bar{S}_i > t\right] \leq (1+\varepsilon)\sum_{i=1}^{m} \mathbb{P}[\bar{S}_i > t].$$

1. If $\widehat{B}_1$ is the smallest job, then

$$\mathbb{P}[N_1 > n] = \mathbb{E}\mathbb{P}\left[\widehat{B}_1 > \frac{A}{m}\right]^n = \mathbb{E}\left(1 - \bar{G}(m\widehat{B}_1)\right)^n = \mathbb{E}\left(1 - \bar{F}_1(m\widehat{B}_1)^{\frac{1}{\alpha_1}}\right)^n$$

2. What is the relationship between $\bar{F}_1(x)$ and $\bar{G}(x)$?

$$\log \bar{F}_1(x) = \log \mathbb{P}[m\widehat{B}_1 > x] = \log\left(\bar{F}(x/m)\right)^m \approx m^{1-\gamma} \log \bar{F}(x).$$

3. Recalling $(\star)$,

$$\frac{-\log \mathbb{P}[\bar{S}_1 > t]}{\log t} \xrightarrow[t\to\infty]{} \frac{\alpha}{m^{\gamma-1}} \qquad (\star)$$

Idea of the proof (I)

The **upper** bound is

$$\mathbb{P}[\Theta_m > t] \le \mathbb{P}\left[\sum_{i=1}^{m} \bar{S}_i > t\right] \le (1+\varepsilon)\sum_{i=1}^{m} \mathbb{P}[\bar{S}_i > t].$$

1. If $\widehat{B}_1$ is the smallest job, then

$$\mathbb{P}[N_1 > n] = \mathbb{E}\mathbb{P}\left[\widehat{B}_1 > \frac{A}{m}\right]^n = \mathbb{E}\left(1 - \bar{G}(m\widehat{B}_1)\right)^n = \mathbb{E}\left(1 - \bar{F}_1(m\widehat{B}_1)^{\frac{1}{\alpha_1}}\right)^n$$

2. What is the relationship between $\bar{F}_1(x)$ and $\bar{G}(x)$?

$$\log \bar{F}_1(x) = \log \mathbb{P}[m\widehat{B}_1 > x] = \log\left(\bar{F}(x/m)\right)^m \approx m^{1-\gamma} \log \bar{F}(x).$$

3. Recalling $(\star)$,

$$\frac{-\log \mathbb{P}[\bar{S}_1 > t]}{\log t} \underset{t\to\infty}{\longrightarrow} \frac{\alpha}{m^{\gamma-1}} \qquad (\star)$$

Idea of the proof (II)

4. Similarly, for the 2^{nd} smallest job $\sim 1/t^{\alpha(m-1)^{1-\gamma}}$
5. … and the last one $\sim 1/t^{\alpha}$

- If $\gamma > 1$ (superexponential), then the **lower** bound is determined by the *minimum power law index* $(\alpha m^{1-\gamma} < \ldots < \alpha)$

$$\frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \gtrsim \frac{\alpha}{m^{\gamma-1}}. \quad (1)$$

- Equivalently, if $\gamma \leq 1$ ((sub)exponential), then

$$\frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \gtrsim \alpha. \quad (2)$$

Idea of the proof (II)

4. Similarly, for the 2^{nd} smallest job $\sim 1/t^{\alpha(m-1)^{1-\gamma}}$
5. ... and the last one $\sim 1/t^{\alpha}$

- If $\gamma > 1$ (superexponential), then the **lower** bound is determined by the *minimum power law index* $(\alpha m^{1-\gamma} < \ldots < \alpha)$

$$\frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \gtrsim \frac{\alpha}{m^{\gamma-1}}. \tag{1}$$

- Equivalently, if $\gamma \leq 1$ ((sub)exponential), then

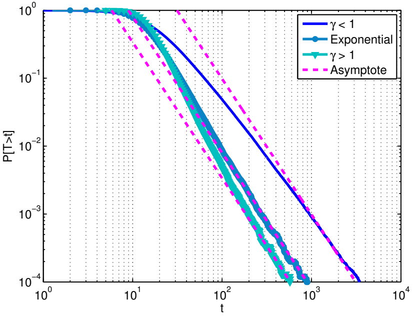
$$\frac{-\log \mathbb{P}[\Theta_m > t]}{\log t} \gtrsim \alpha. \tag{2}$$

Simulations

Example 1. FCFS: All job types generate same power law asymptotics

- Service time $S \sim 1/t^2$
- # jobs: $m = 10$

Figure: Logarithmic asymptotics for $\alpha = 2$ under FCFS.

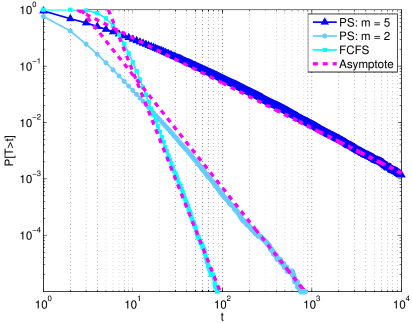


Simulations

Example 2. PS: The effect of the number of (superexponential) jobs

- $B \sim$ superexponential ($\gamma > 1$)
- # jobs: $m = 2$ and $m = 5$, service time with $\alpha = 4$

Figure: Logarithmic asymptotics for $\alpha = 4$ under PS and FCFS discipline.



Queueing: PS could be always unstable

Theorem 3

If jobs are superexponential ($\gamma > 1$), then for any arrival rate $\lambda > 0$ and any $\alpha > 0$, the PS queue is unstable.

- Queueing with retransmissions & scheduling is hard
- More to come in our forthcoming paper…

Queueing: PS could be always unstable

Theorem 3

If jobs are superexponential ($\gamma > 1$), then for any arrival rate $\lambda > 0$ and any $\alpha > 0$, the PS queue is unstable.

- Queueing with retransmissions & scheduling is hard
- More to come in our forthcoming paper...

Conclusions

- FCFS: power law of same index for both super/subexponential
- PS: new phenomenon - dramatic difference between super/subexponential jobs
- Queueing: for superexponential jobs, sharing induces instabilities → zero throughput
- **Sharing** is not always good ☹

Conclusions

- FCFS: power law of same index for both super/subexponential
- PS: new phenomenon - dramatic difference between super/subexponential jobs
- Queueing: for superexponential jobs, sharing induces instabilities → zero throughput
- **Sharing** is not always good ☹

Conclusions

- FCFS: power law of same index for both super/subexponential
- PS: new phenomenon - dramatic difference between super/subexponential jobs
- Queueing: for superexponential jobs, sharing induces instabilities → zero throughput
- **Sharing** is not always good ☹

Conclusions

- FCFS: power law of same index for both super/subexponential
- PS: new phenomenon - dramatic difference between super/subexponential jobs
- Queueing: for superexponential jobs, sharing induces instabilities → zero throughput
- **Sharing** is not always good ☹

Thank you

Questions?