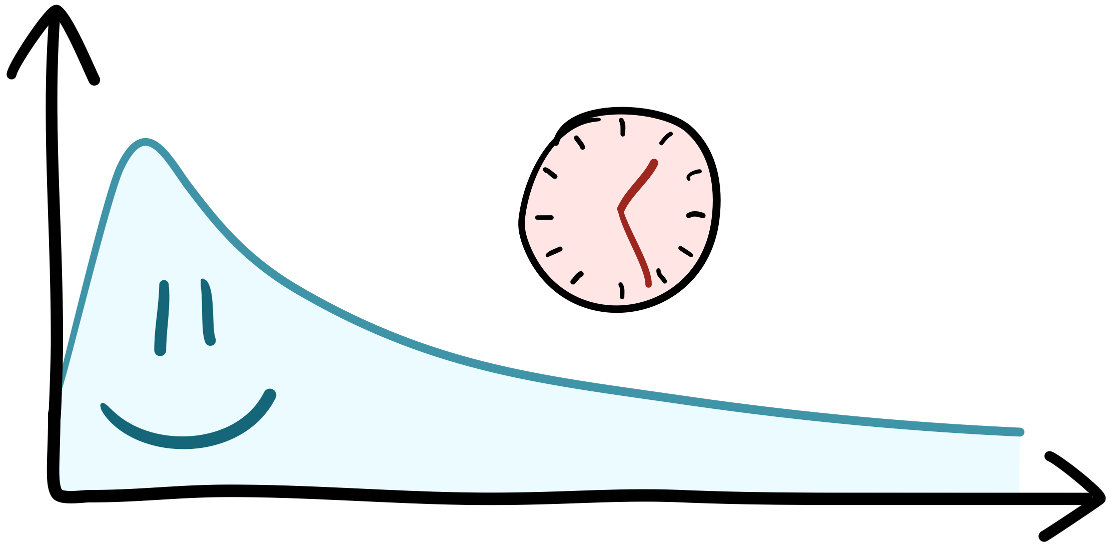
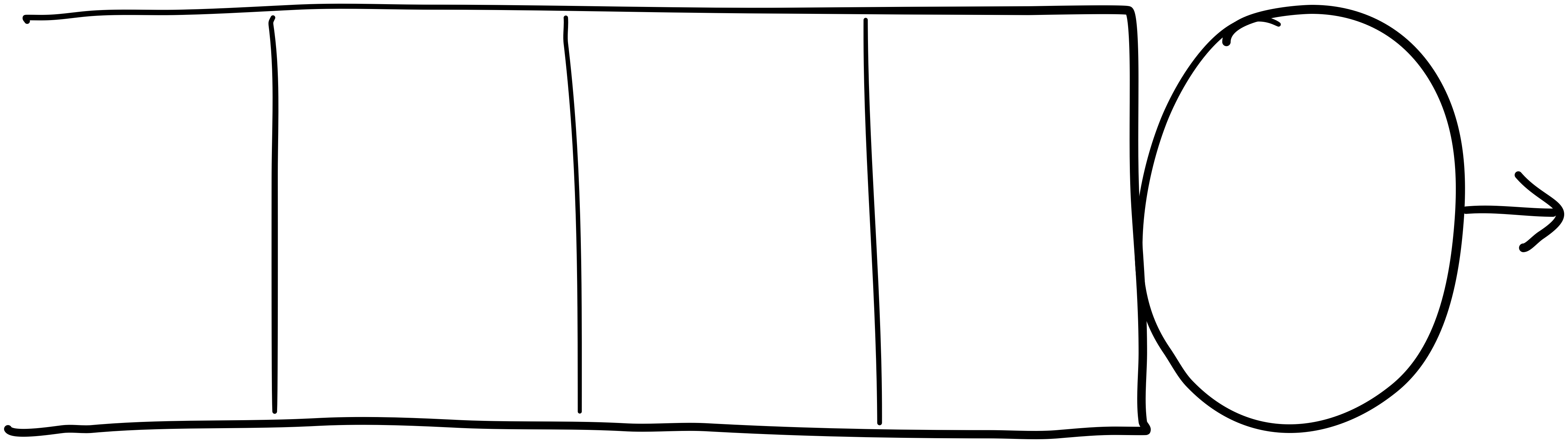
Characterizing Policies *with* **Optimal Response Time Tails** *under Heavy-Tailed Job Sizes*

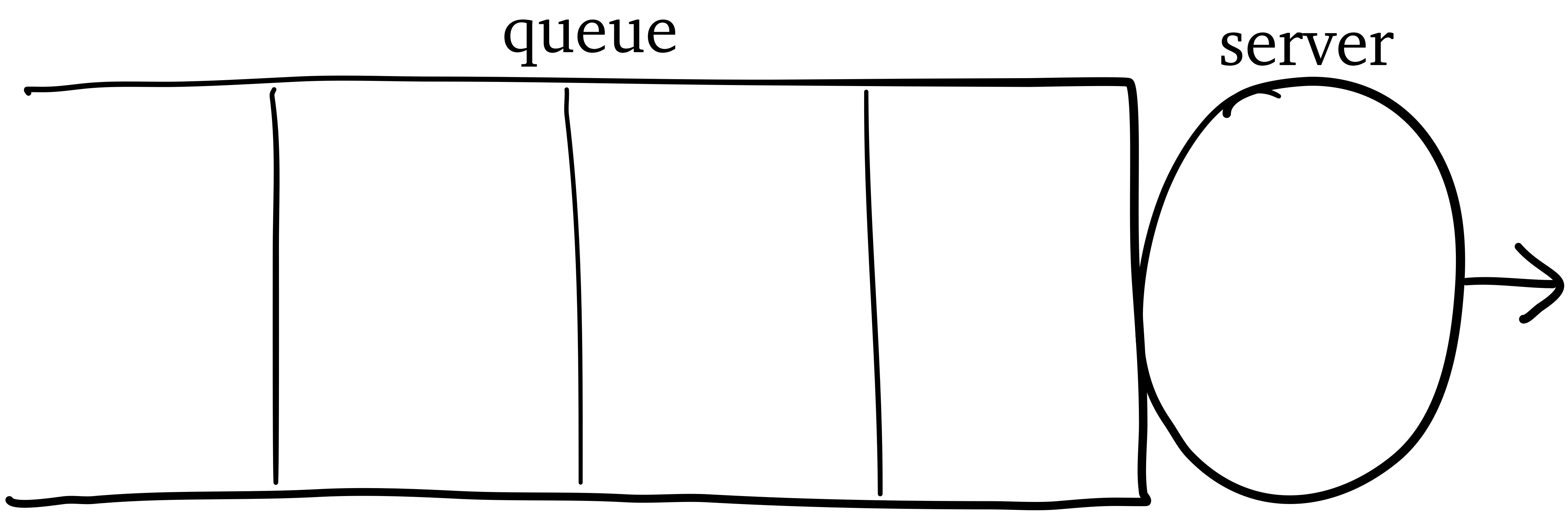
Ziv Scully *CMU*
Lucas van Kreveld *UvA*
Onno Boxma *TU/e*
Jan-Pieter Dorsman *UvA*
Adam Wierman *Caltech*



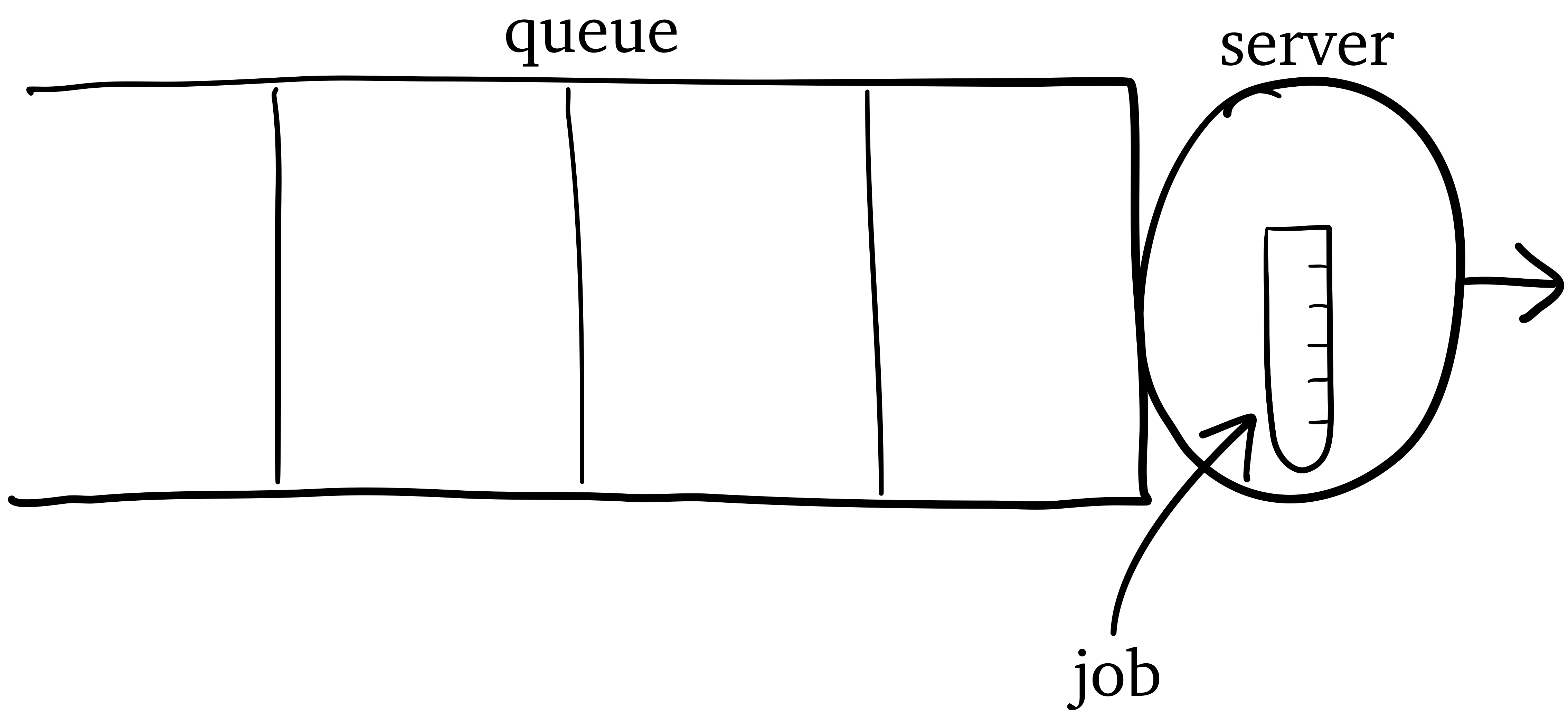
M/G/1 Queue



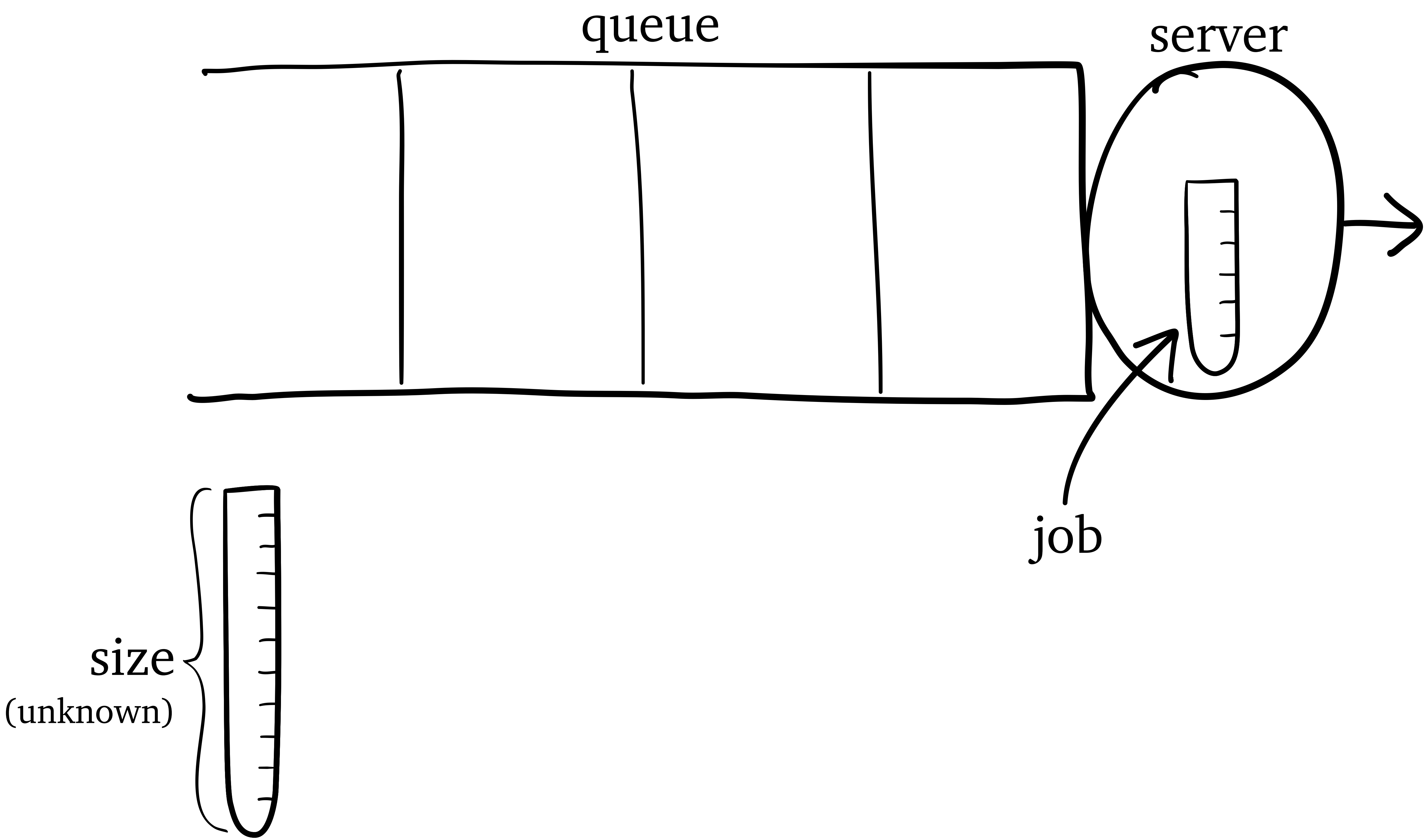
M/G/1 Queue



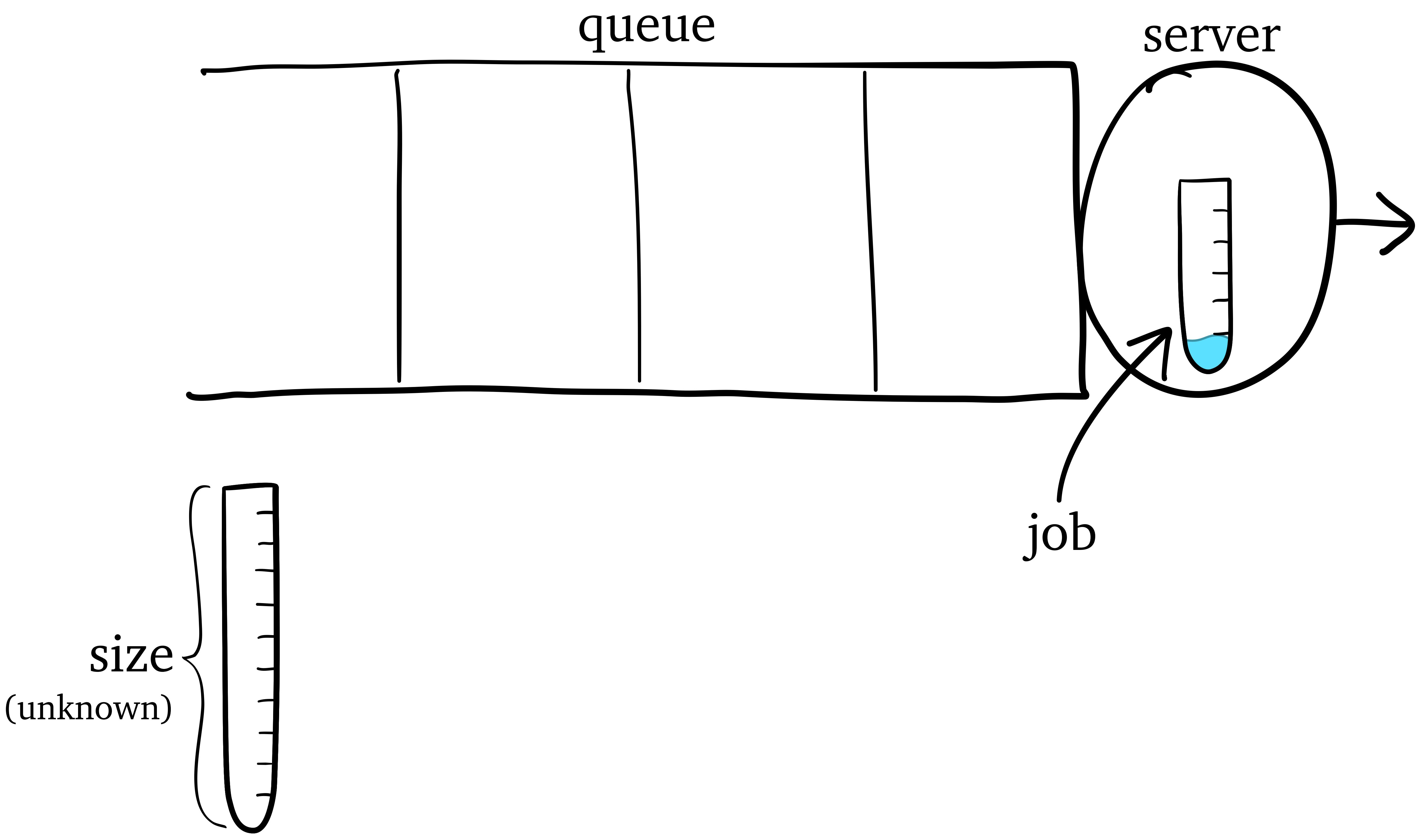
M/G/1 Queue



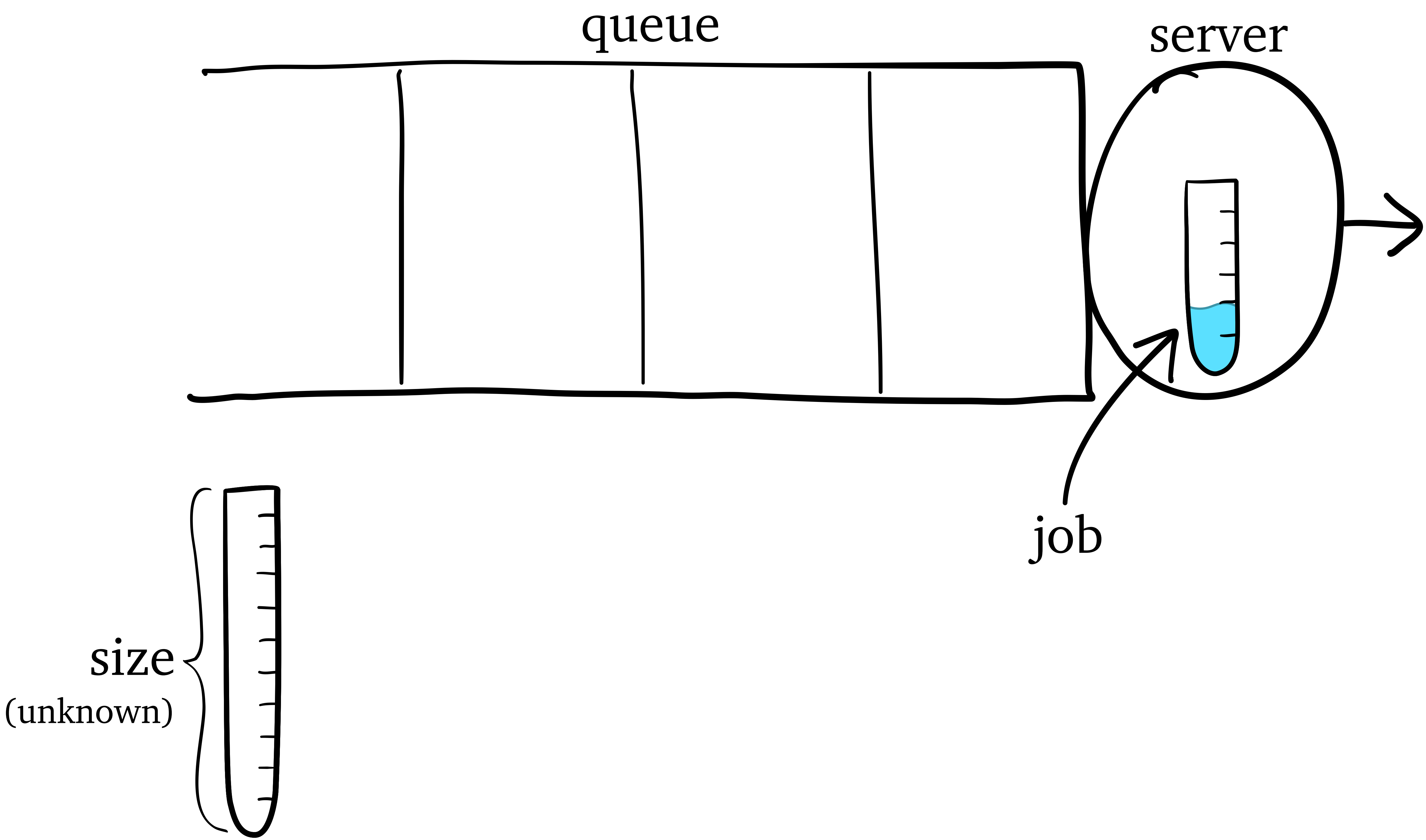
M/G/1 Queue



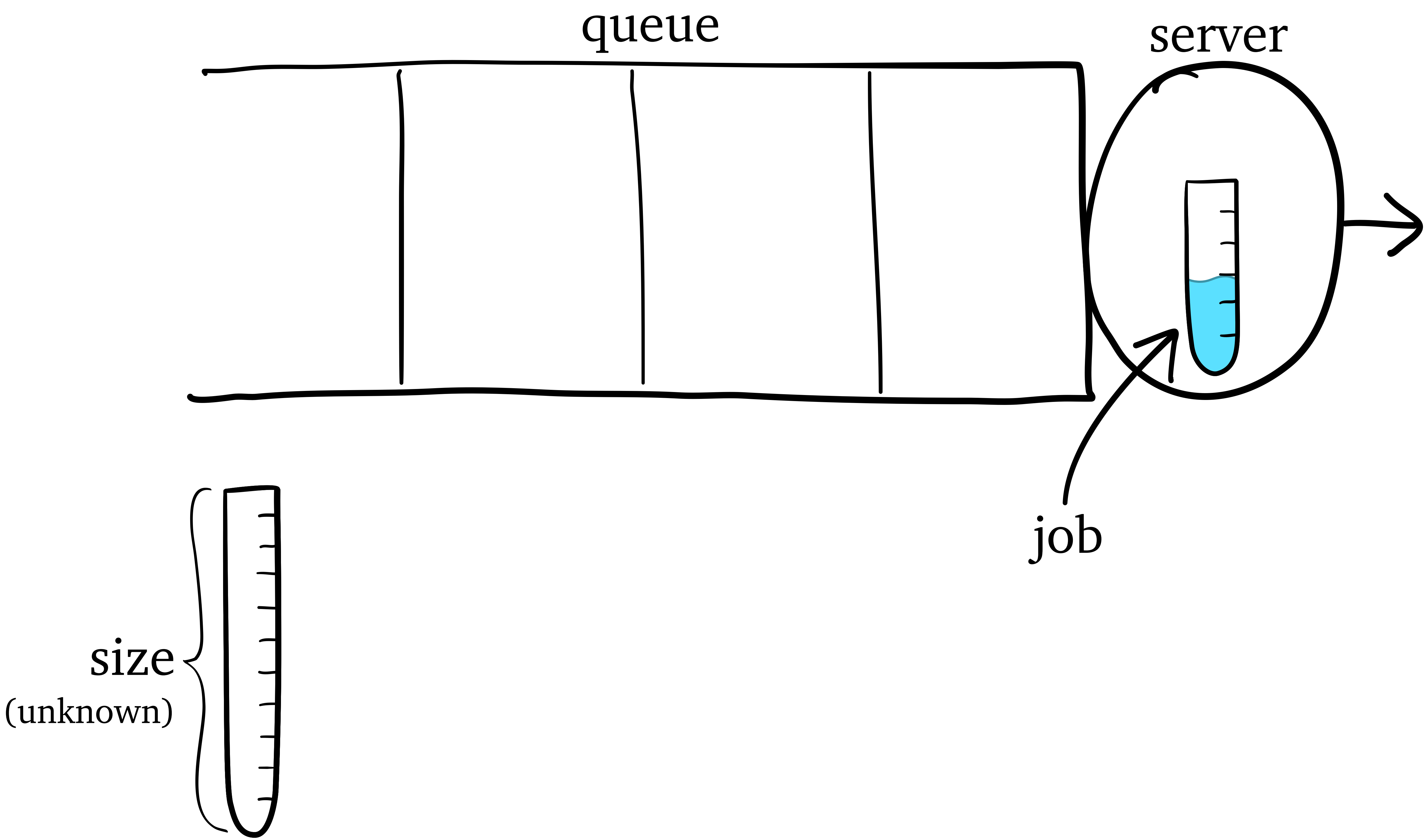
M/G/1 Queue



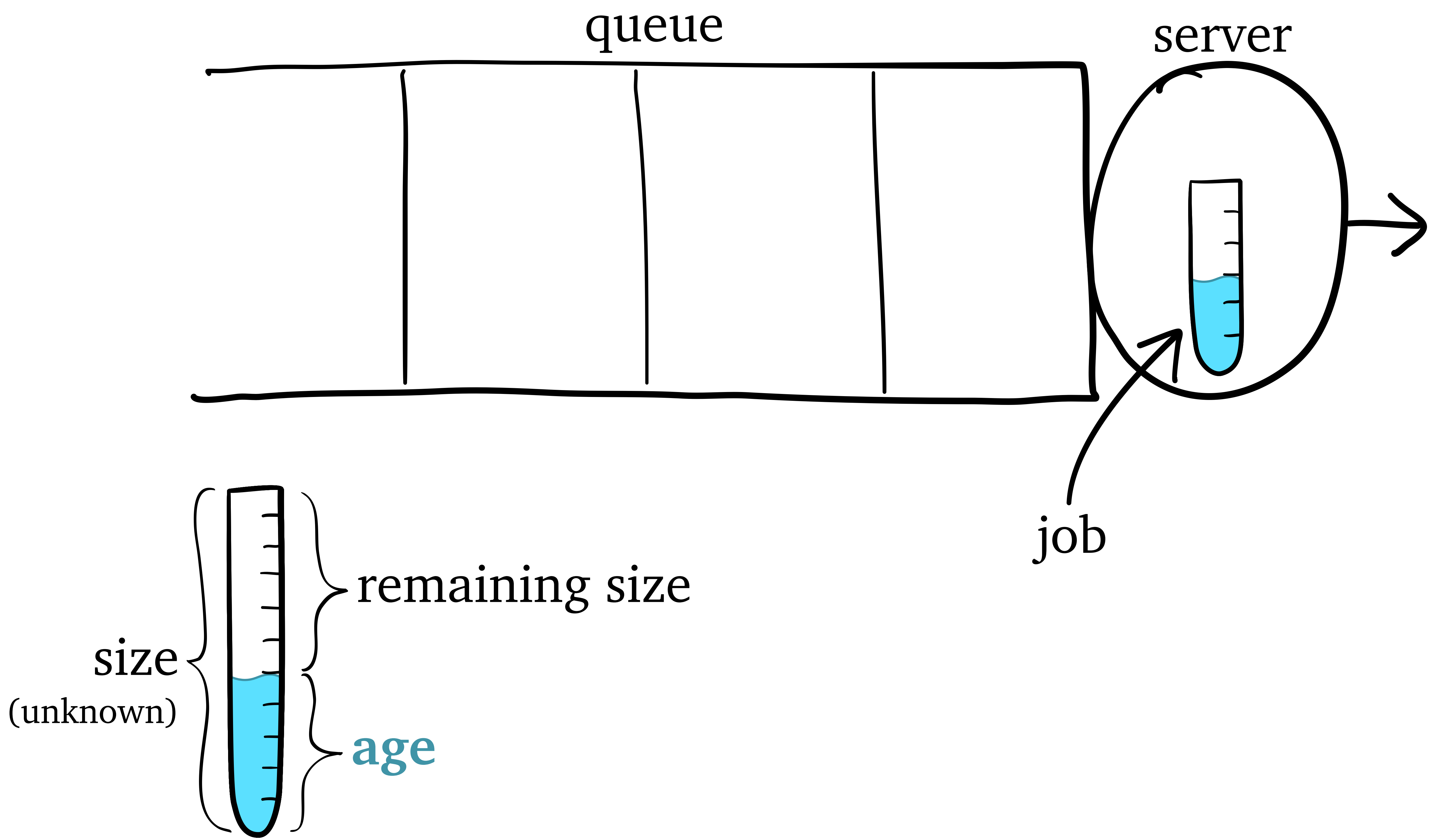
M/G/1 Queue



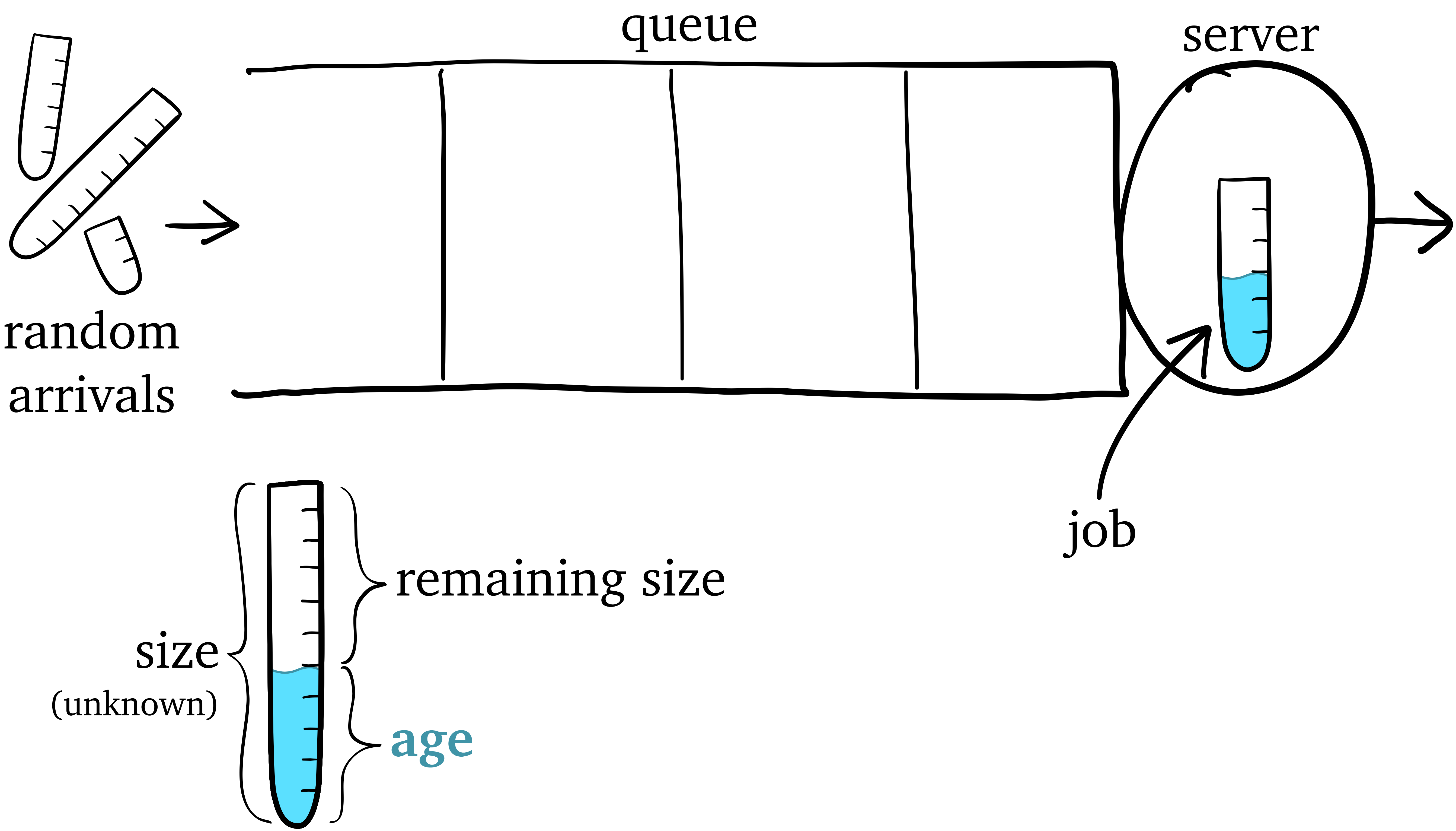
M/G/1 Queue



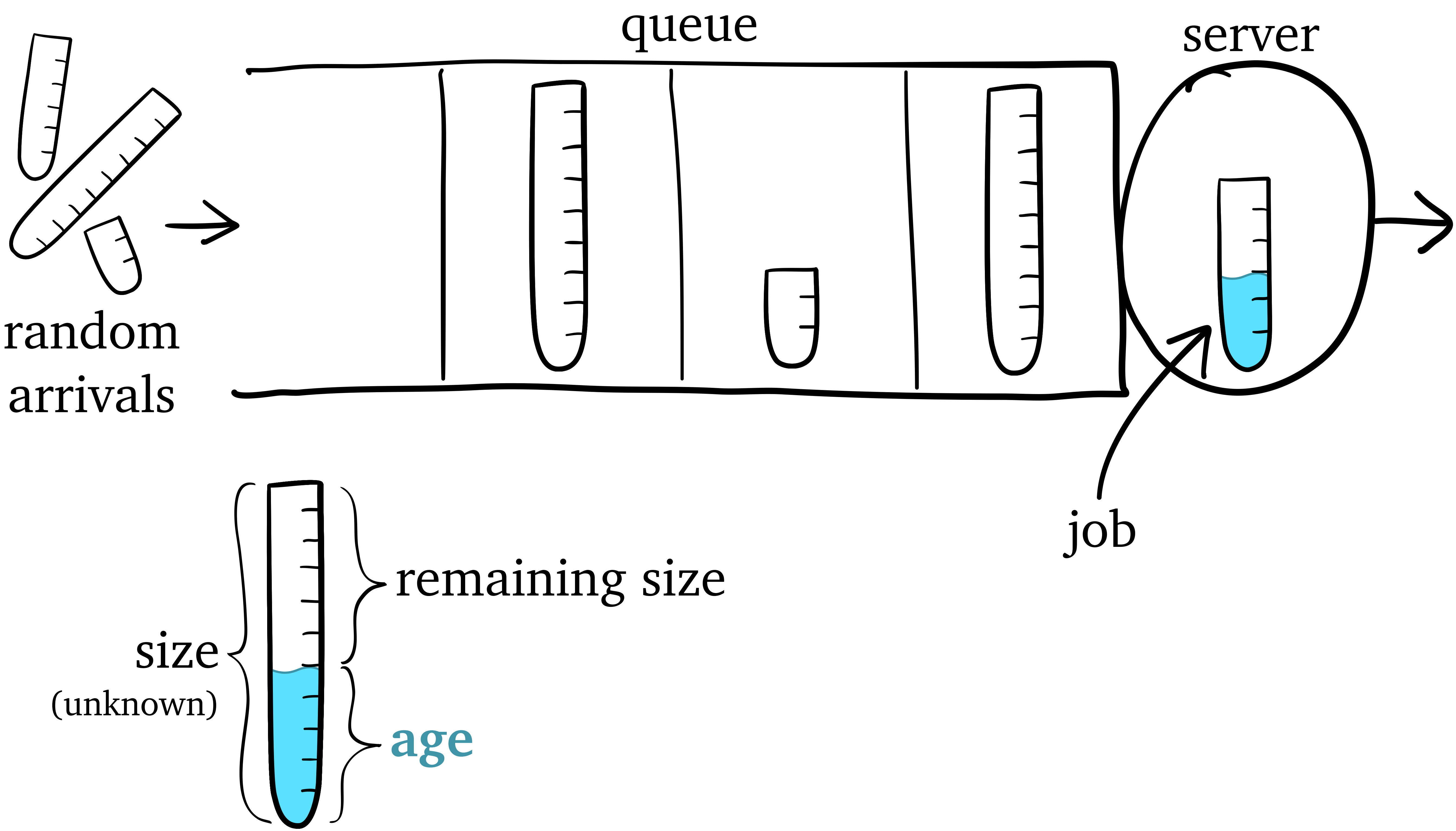
M/G/1 Queue



M/G/1 Queue

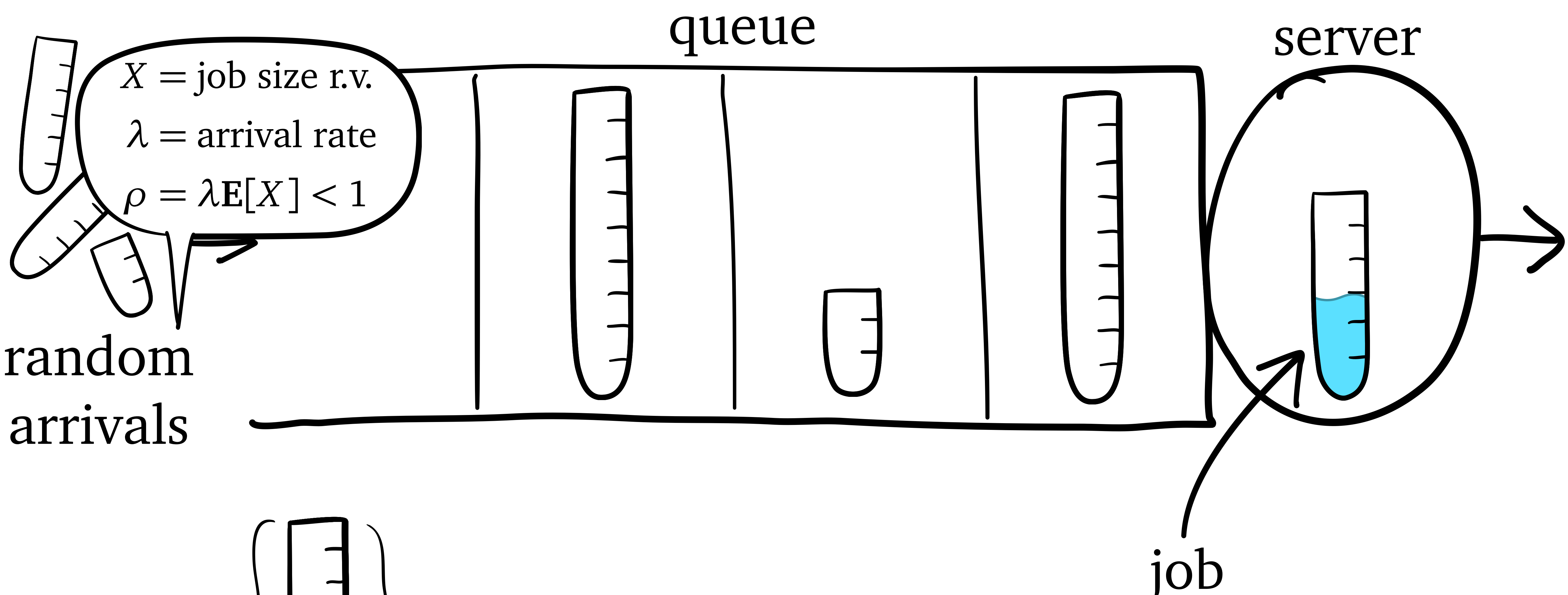

queue
server
random
arrivals
job
size
(unknown)
remaining size
age

M/G/1 Queue

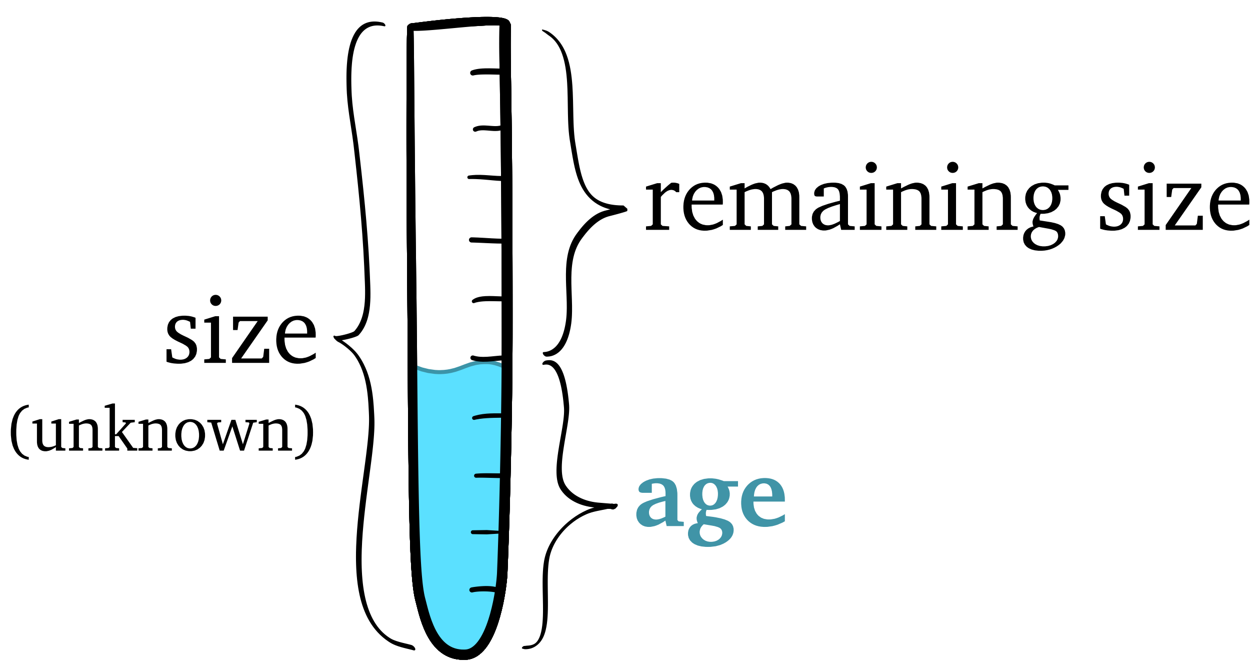


queue
server
random
arrivals
job
size
(unknown)
remaining size
age

M/G/1 Queue

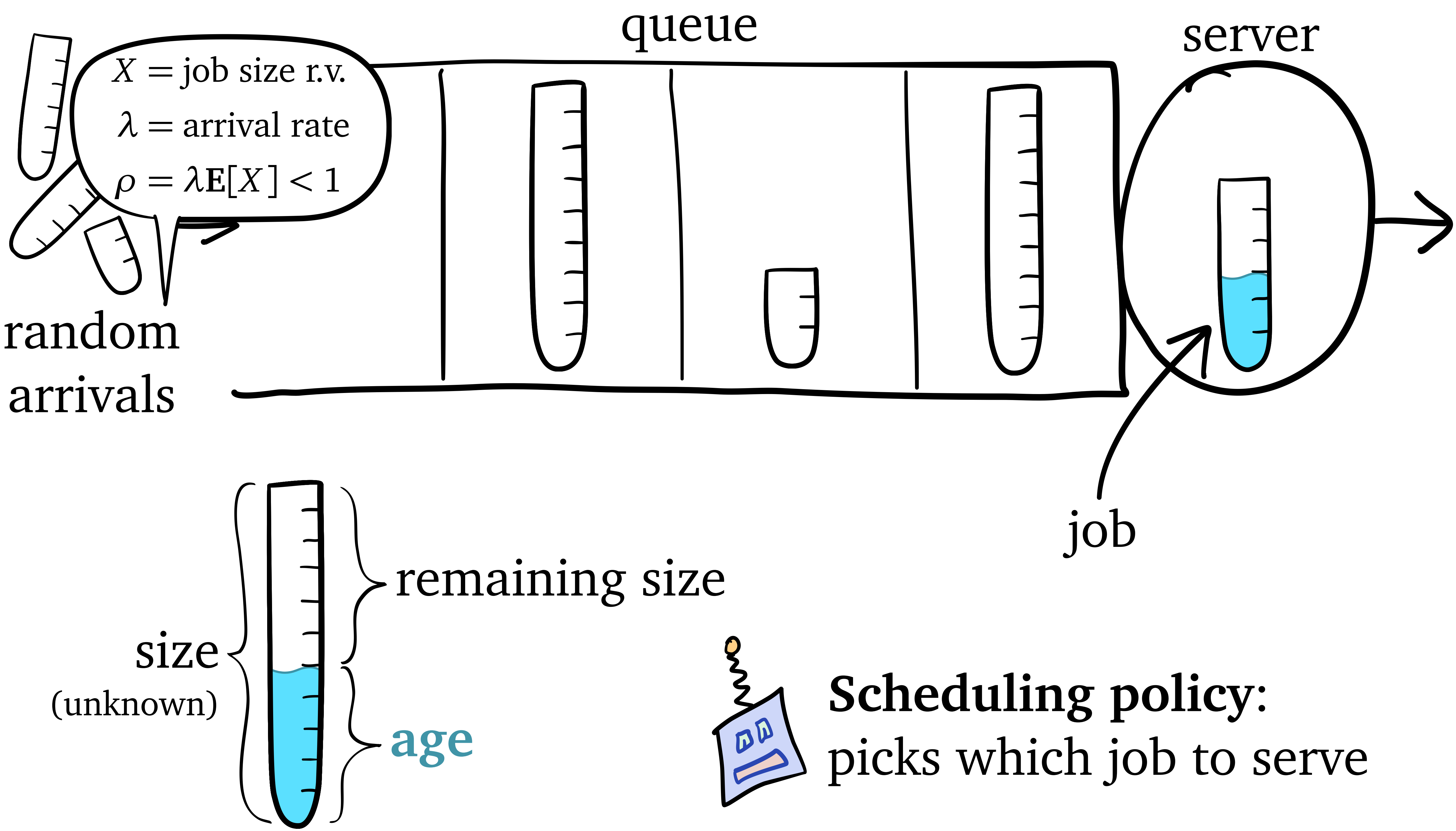


queue
server
X = job size r.v.
λ = arrival rate
$\rho = \lambda \mathbf{E}[X] < 1$
random arrivals
job



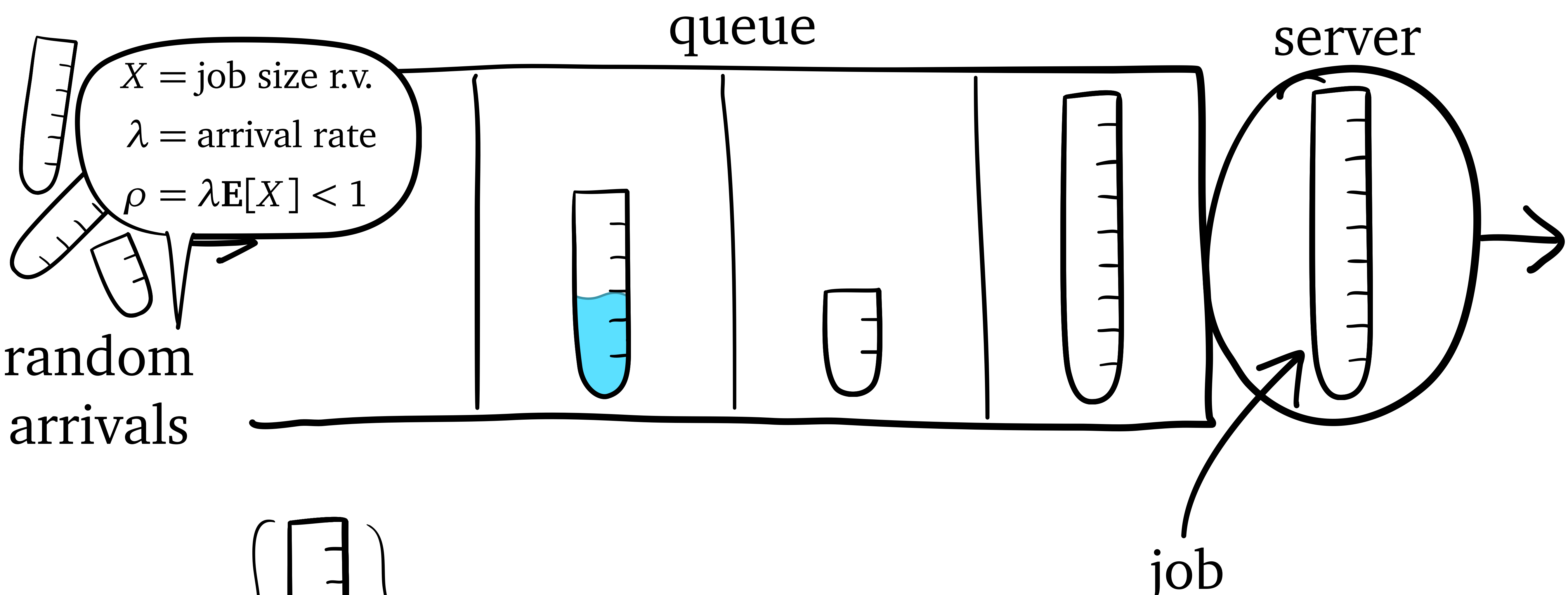
remaining size
size
(unknown)
age

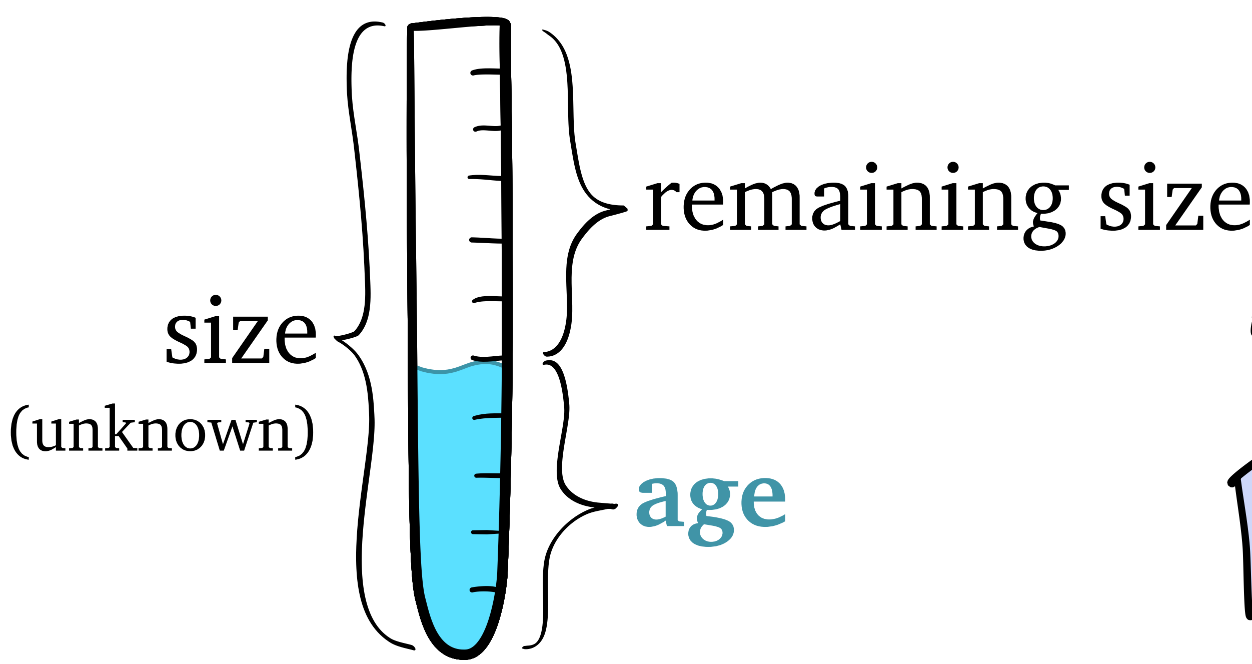
M/G/1 Queue

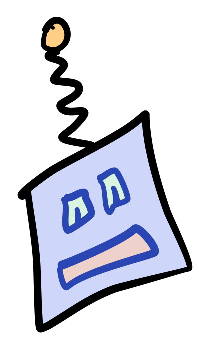


Scheduling policy:
picks which job to serve

M/G/1 Queue

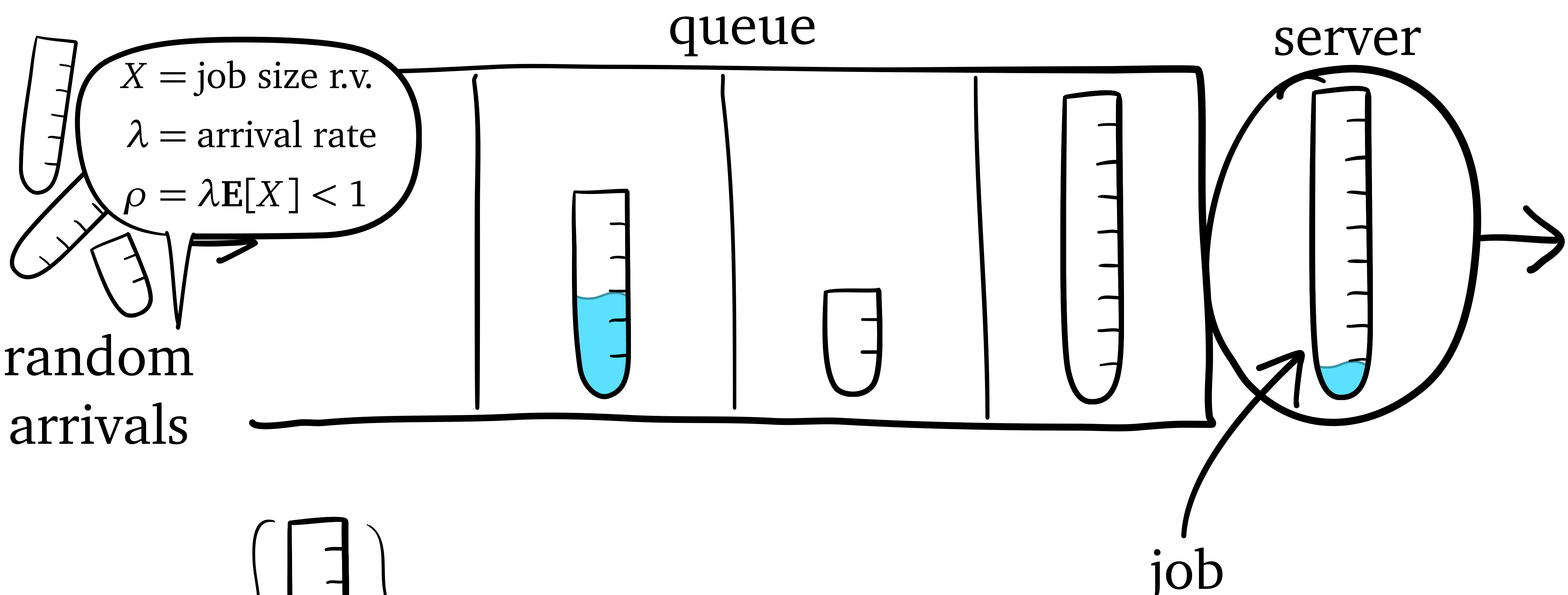


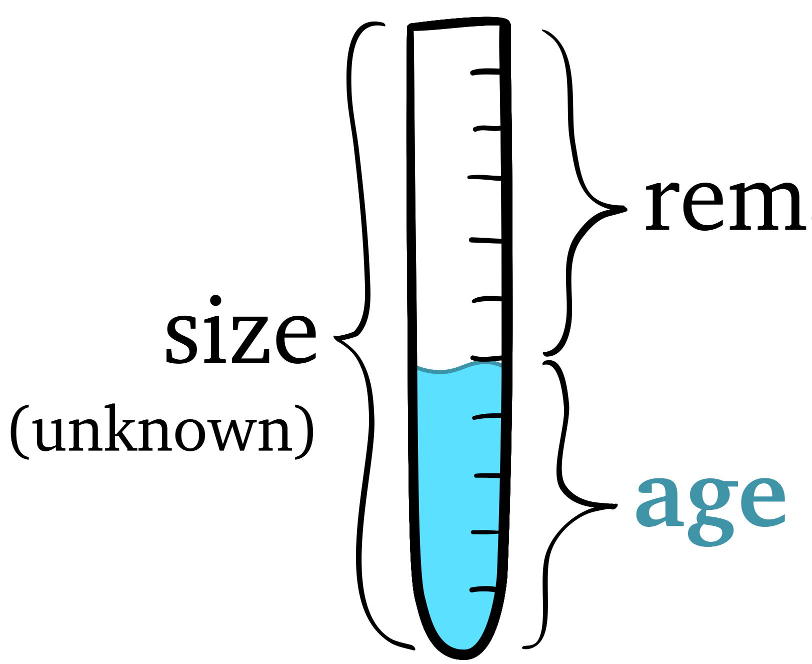


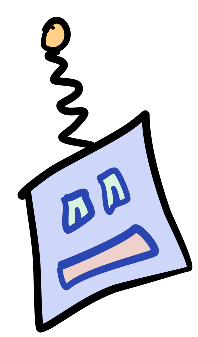


Scheduling policy:
picks which job to serve

M/G/1 Queue

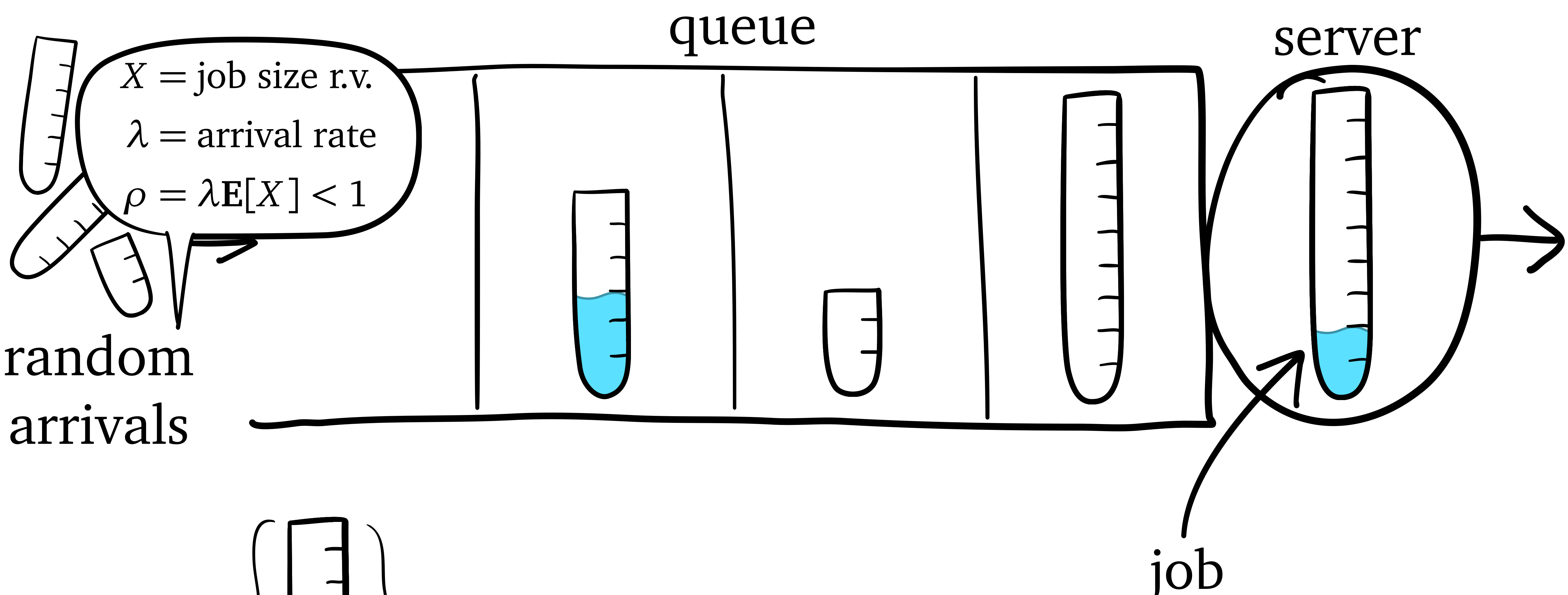


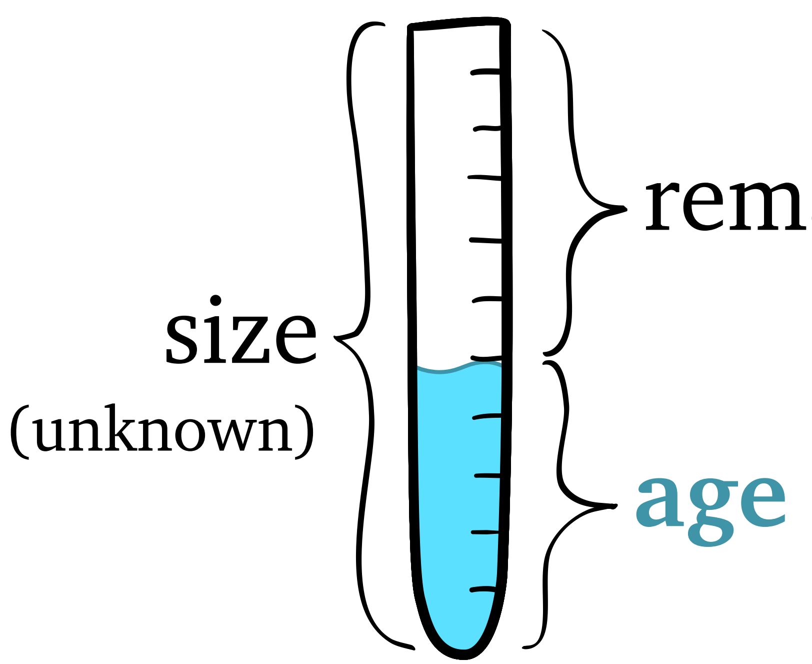


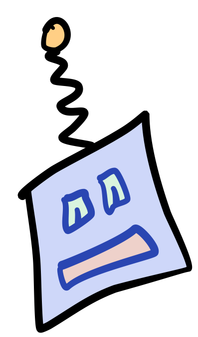


Scheduling policy:
picks which job to serve

M/G/1 Queue

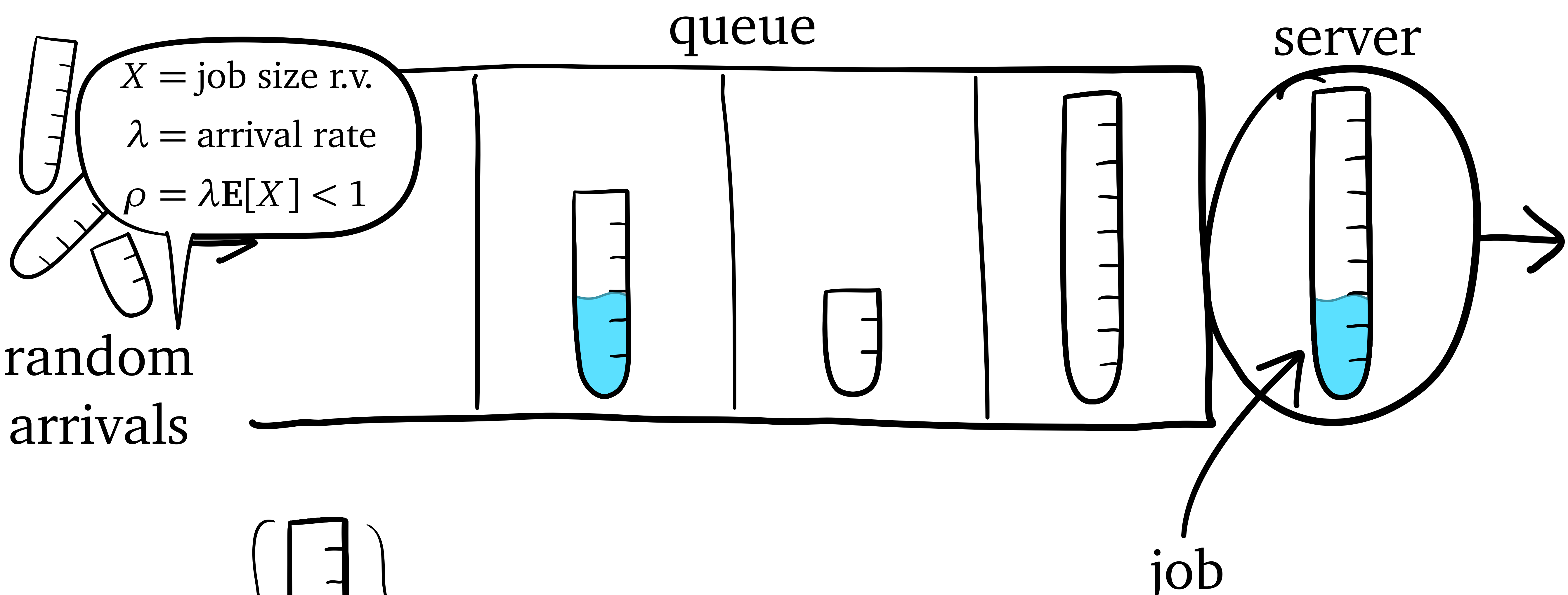


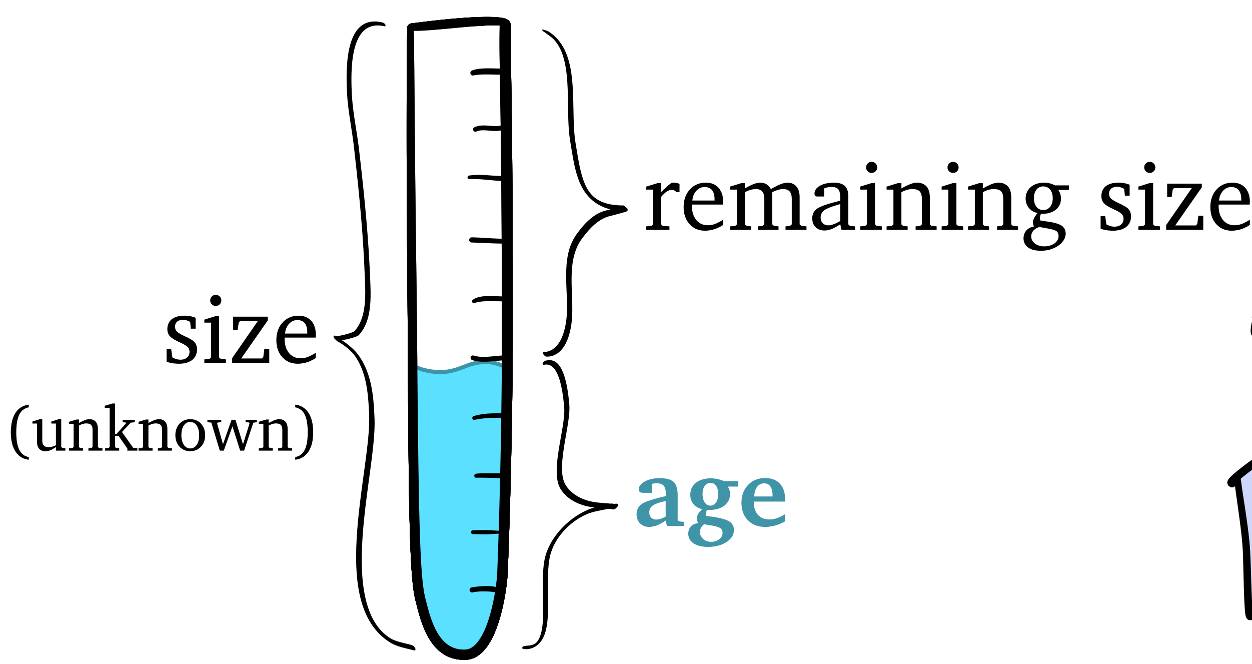


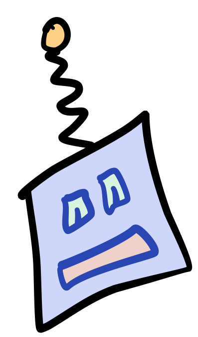


Scheduling policy:
picks which job to serve

M/G/1 Queue

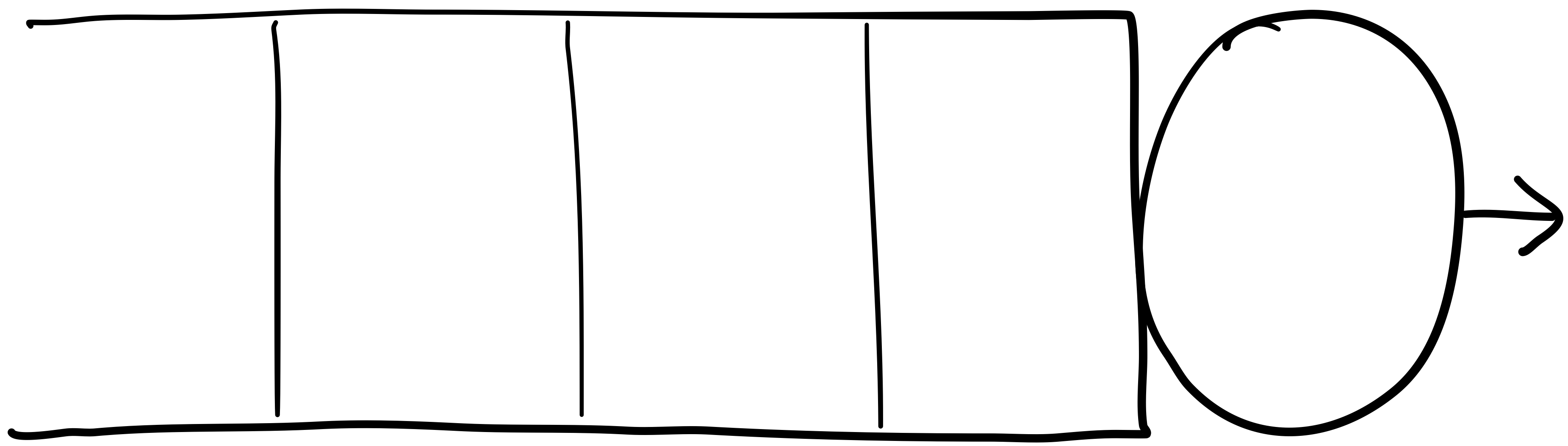




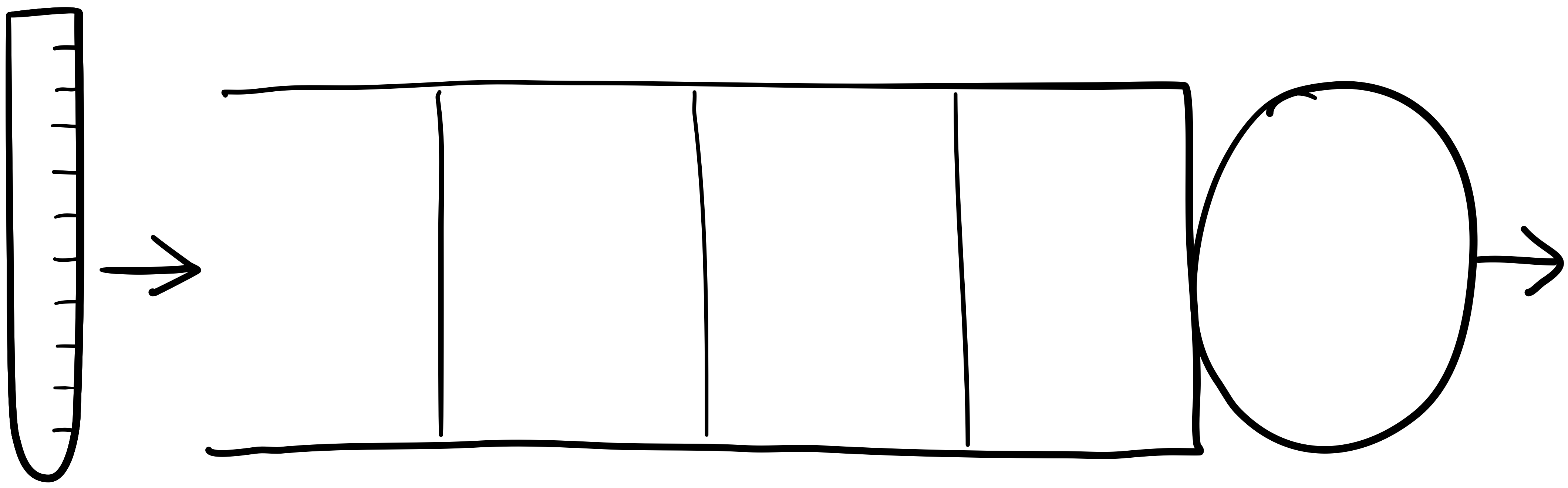


Scheduling policy:
picks which job to serve

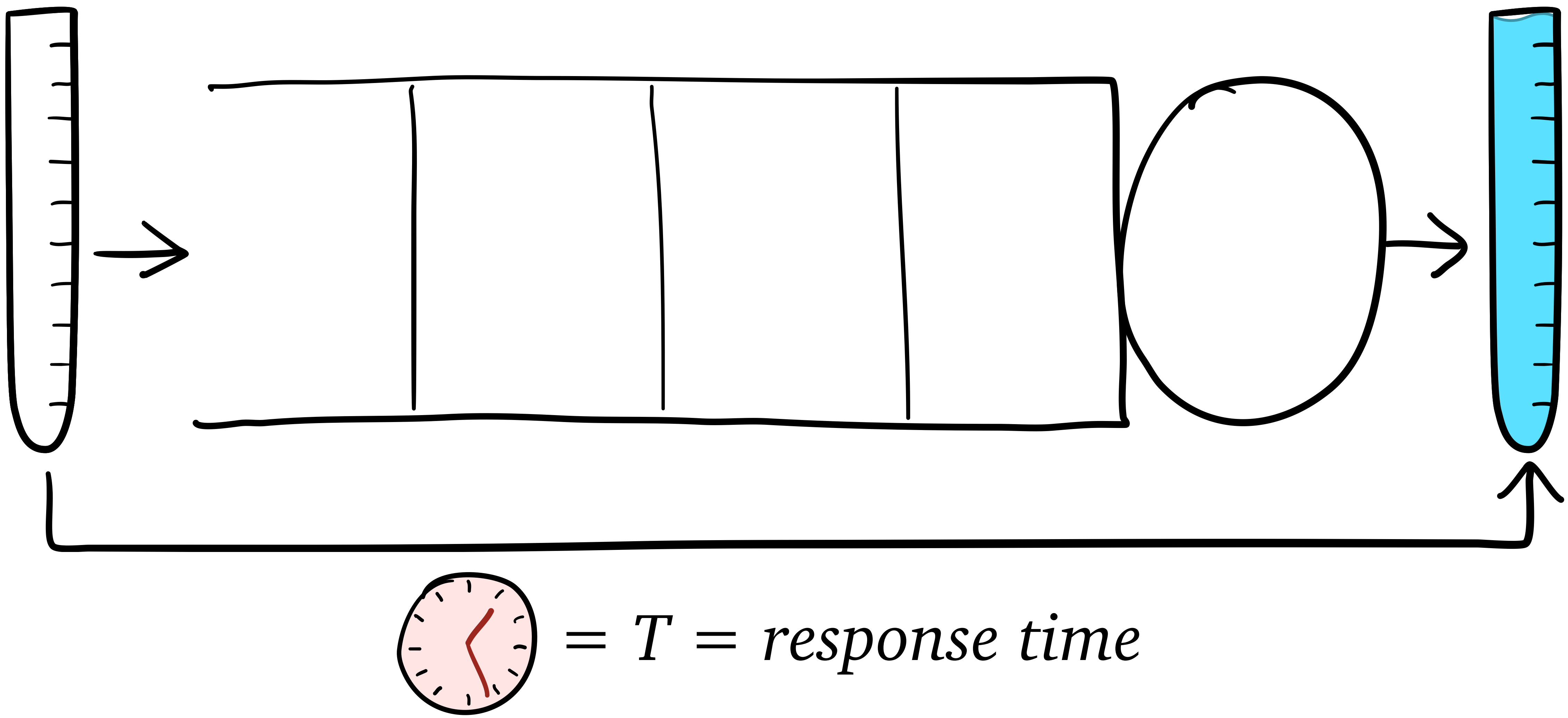
Response Time



Response Time

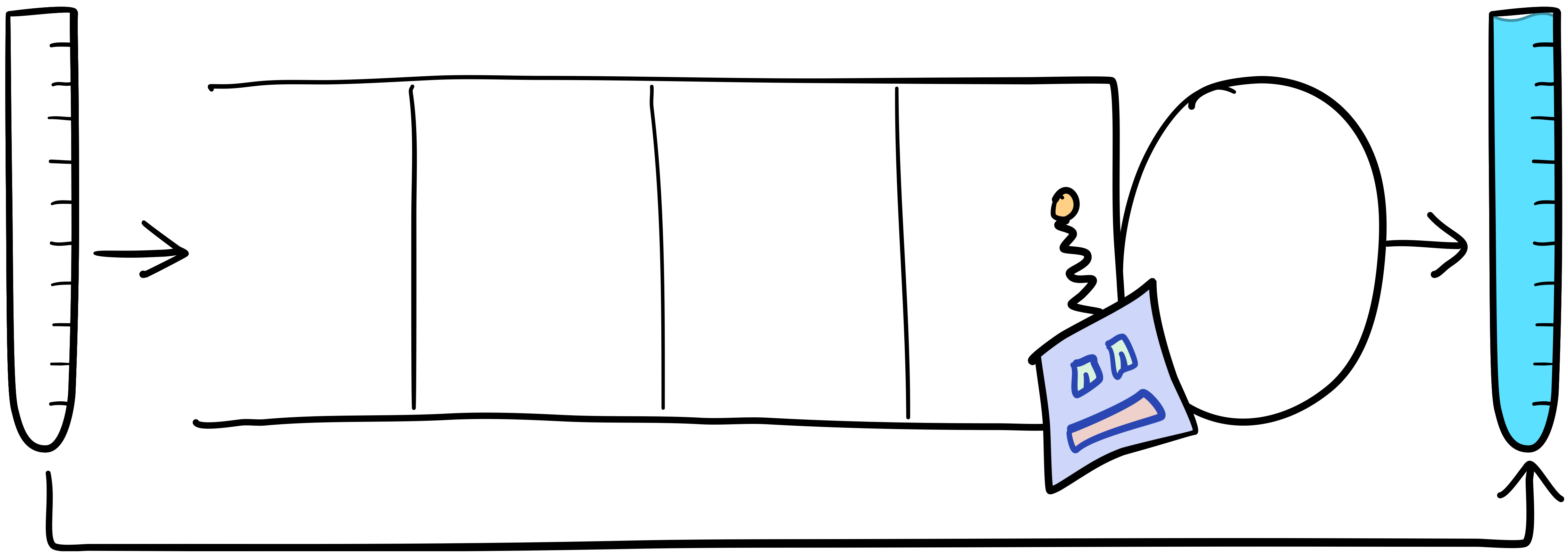


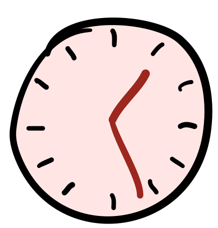
Response Time



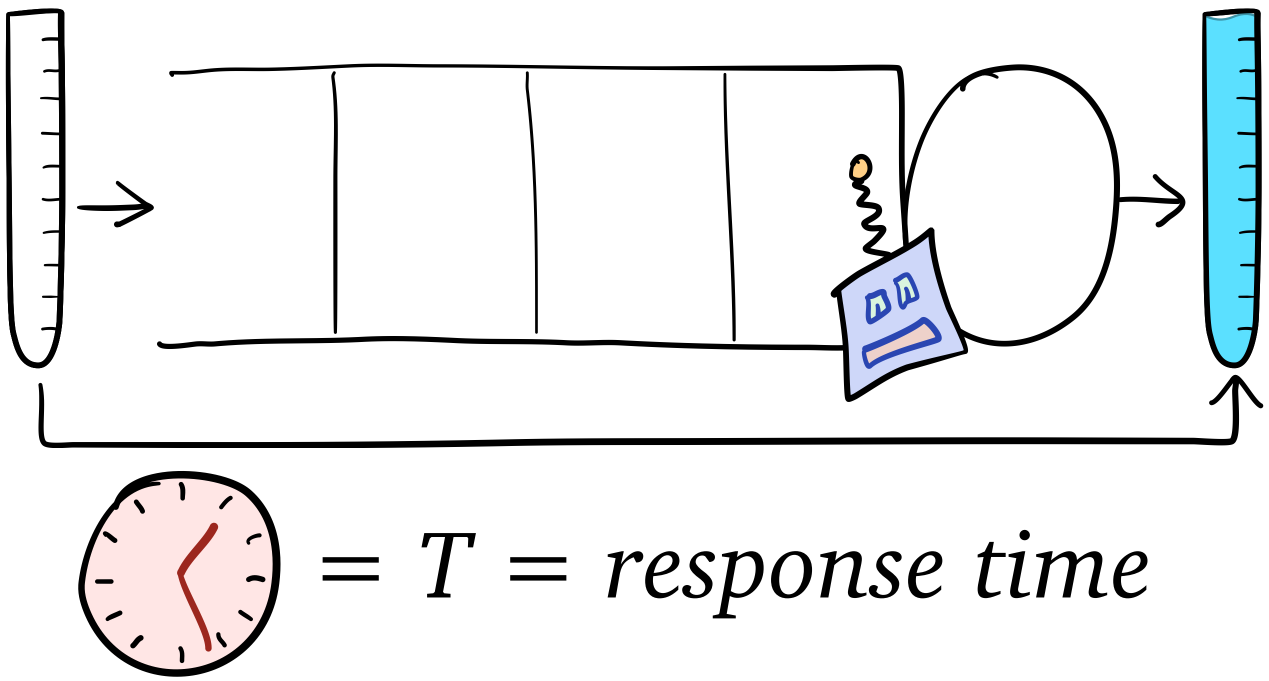
$= T =$ *response time*

Response Time



 $= T = \textit{response time}$

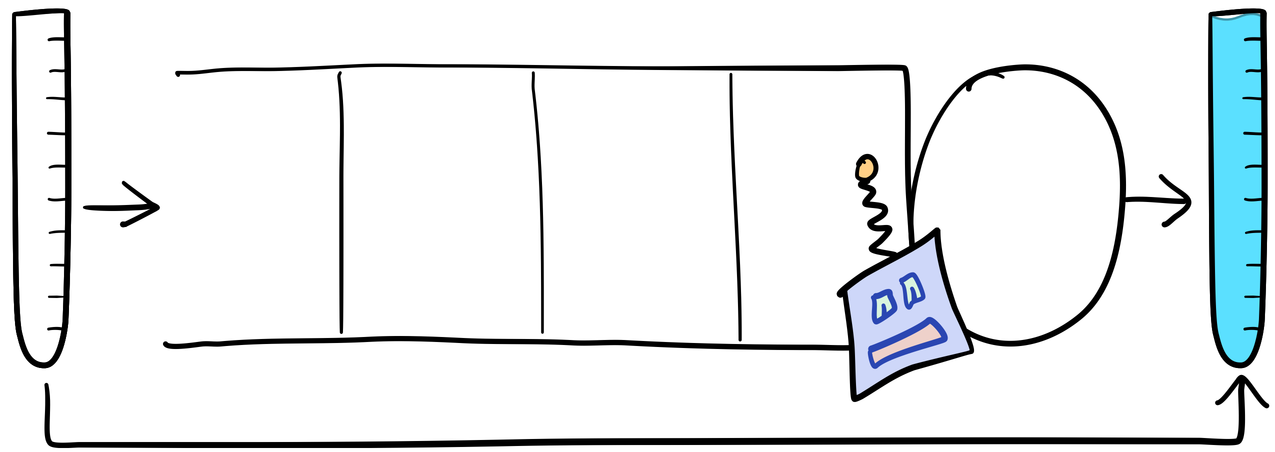
Response Time

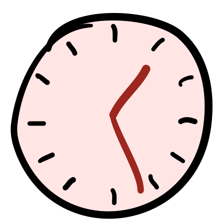


= T = *response time*

Goal: schedule to minimize two metrics

Response Time

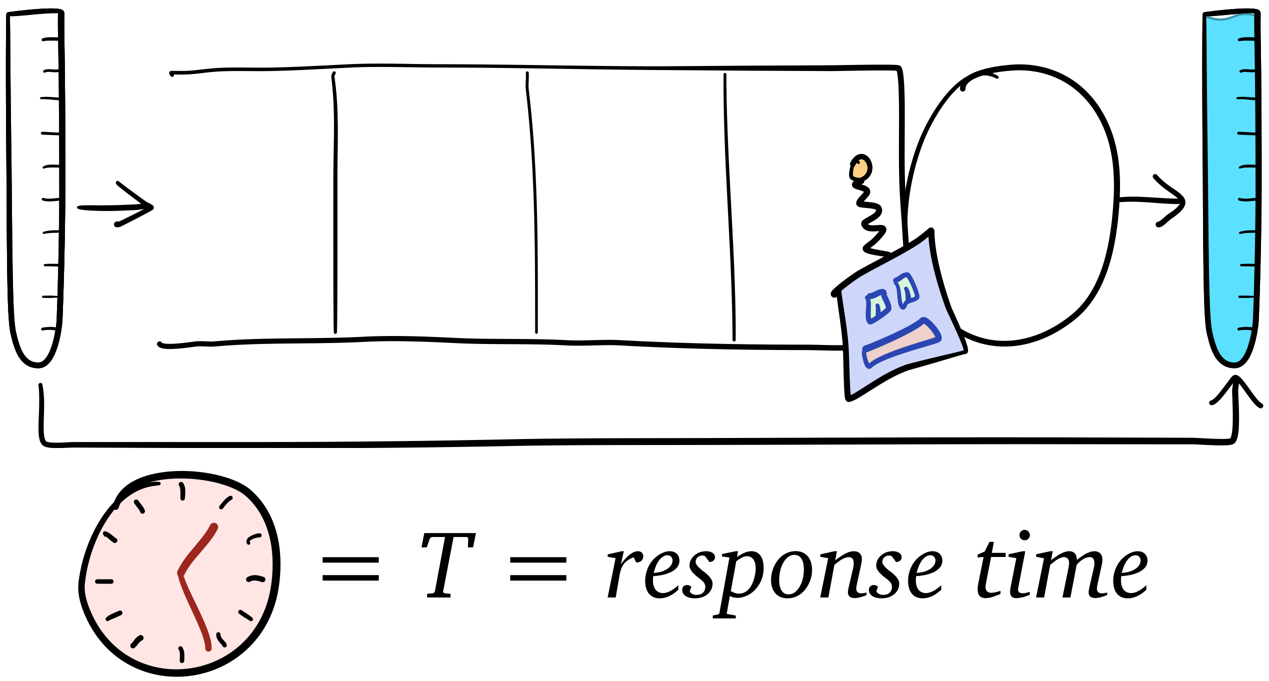


 $= T =$ *response time*

Goal: schedule to minimize two metrics

- *mean* response time $\mathbf{E}[T]$

Response Time

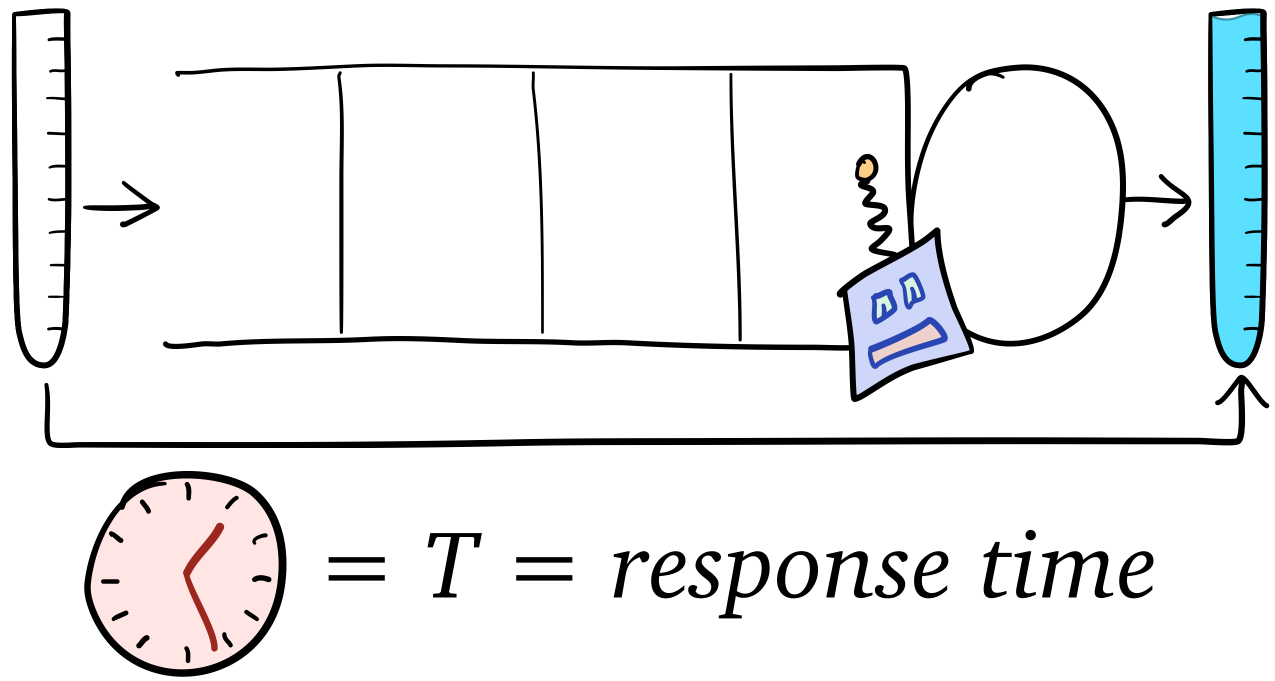


$= T =$ *response time*

Goal: schedule to minimize two metrics

- *mean* response time $\mathbf{E}[T]$
- *tail* of response time $\mathbf{P}[T > t]$

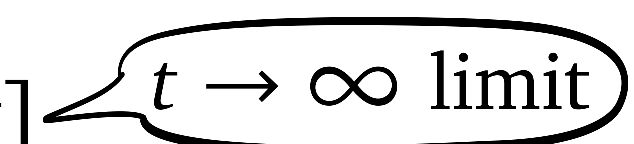
Response Time



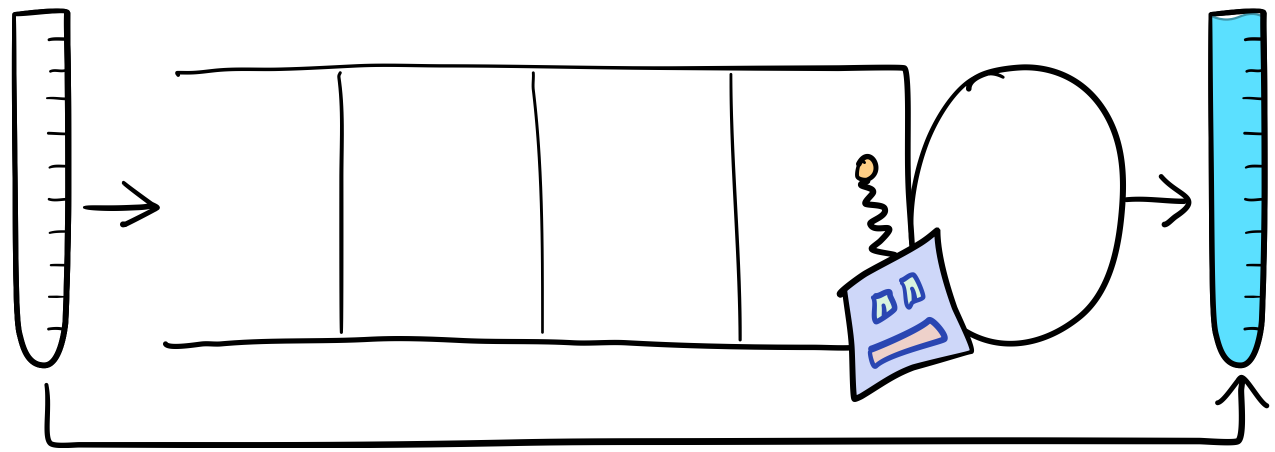
$= T =$ *response time*

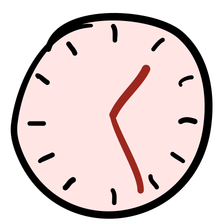
Goal: schedule to minimize two metrics

- *mean* response time $\mathbf{E}[T]$
- *tail* of response time $\mathbf{P}[T > t]$



Response Time



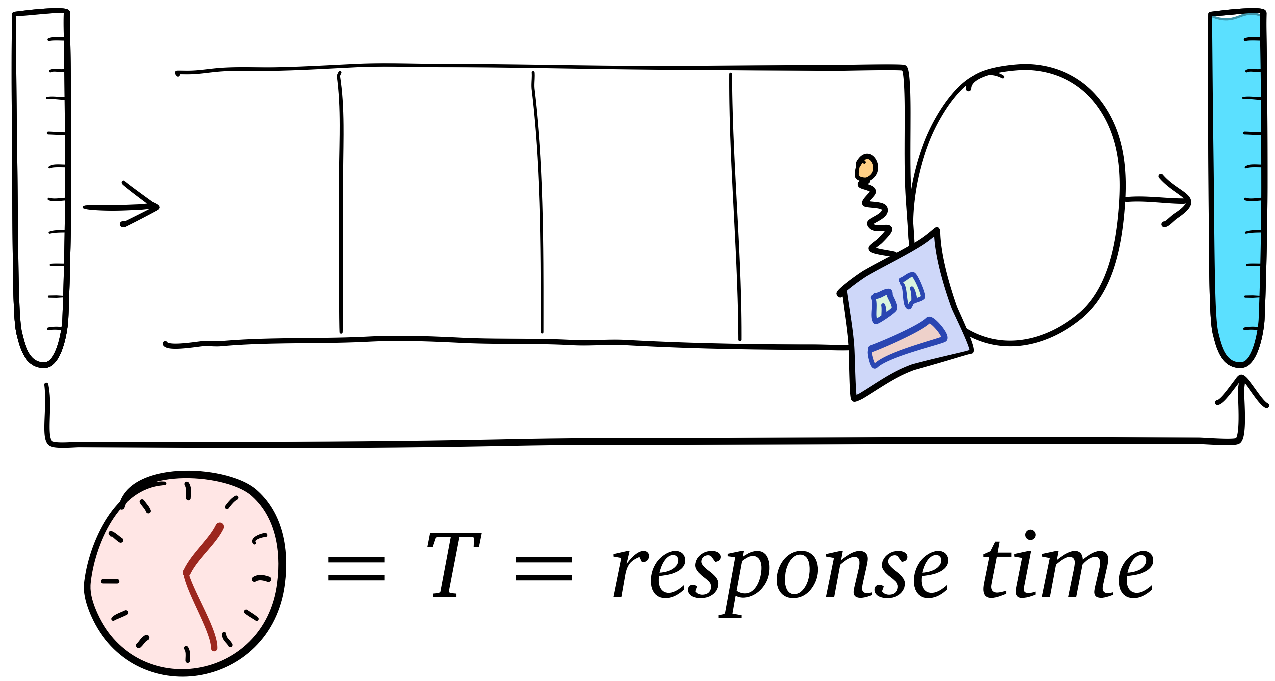
 $= T =$ *response time*

Goal: schedule to minimize two metrics

- *mean* response time $\mathbf{E}[T]$
- *tail* of response time $\mathbf{P}[T > t]$ — $t \to \infty$ limit

Setting: *heavy-tailed* job size X

Response Time



$= T =$ *response time*

Goal: schedule to minimize two metrics

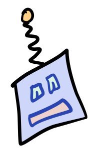
- *mean* response time $\mathbf{E}[T]$
- *tail* of response time $\mathbf{P}[T > t]$

Setting: *heavy-tailed* job size X

Scheduling with Heavy Tails

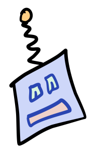
Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$

Scheduling with Heavy Tails

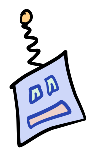
$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS		

First Come,
First Served

Scheduling with Heavy Tails

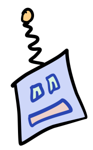
$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	

First Come, First Served

Scheduling with Heavy Tails

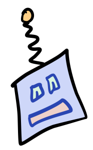
$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**

First Come, First Served

Scheduling with Heavy Tails

$t \to \infty$ limit

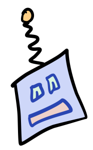
Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**

First Come, First Served

$\mathbf{P}[T > t] = \Theta(t) \cdot \mathbf{P}[X > t]$

Scheduling with Heavy Tails

$t \to \infty$ limit



Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS		

Processor Sharing

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	

Processor Sharing

Scheduling with Heavy Tails

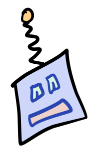
Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**

$t \to \infty$ limit

Processor Sharing

Scheduling with Heavy Tails

$t \to \infty$ limit

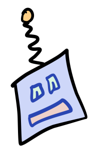
Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**

Processor Sharing

$\mathbf{P}[T > t] = \Theta(\mathbf{1}) \cdot \mathbf{P}[X > t]$

Scheduling with Heavy Tails

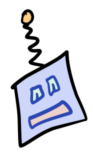
$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB		

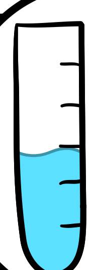
Foreground-Background

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB		

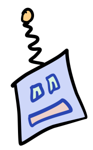
Foreground-Background



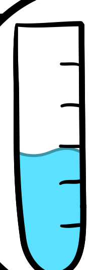
serves job of *least* ***age***

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	

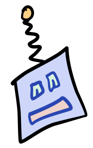
Foreground-Background



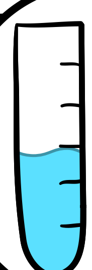
serves job of *least* ***age***

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**

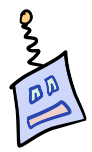
Foreground-Background



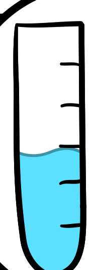
serves job of *least* ***age***

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**

Foreground-Background

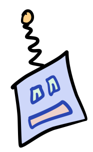


serves job of *least* ***age***

$\mathbf{P}[T > t] = \Theta(\mathbf{1}) \cdot \mathbf{P}[X > t]$

Scheduling with Heavy Tails

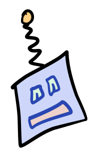
$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins		

prioritize by *Gittins rank*

Scheduling with Heavy Tails

$t \to \infty$ limit



Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	

prioritize by *Gittins rank*

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**

prioritize by *Gittins rank*

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT		

Monotonic Shortest Expected Remaining Processing Time

Scheduling with Heavy Tails

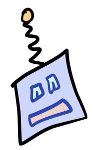
$t \to \infty$ limit

Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	

Monotonic Shortest Expected Remaining Processing Time

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**

Monotonic Shortest Expected Remaining Processing Time

Scheduling with Heavy Tails

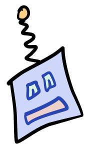
$t \to \infty$ limit

Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**
RMLF		

Randomized Multi-Level Feedback

Scheduling with Heavy Tails

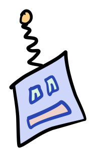
$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**
RMLF	**best** (X unknown)	

Randomized Multi-Level Feedback

Scheduling with Heavy Tails

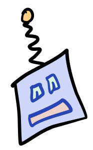
$t \to \infty$ limit

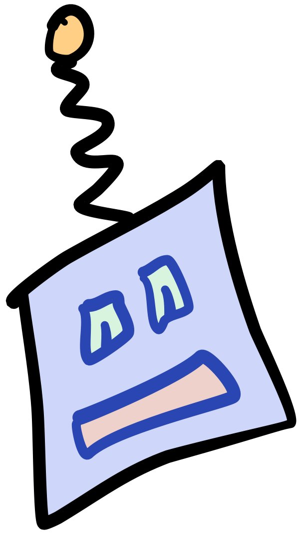
Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**
RMLF	**best** (X unknown)	**???**

Randomized Multi-Level Feedback

Scheduling with Heavy Tails

$t \to \infty$ limit

Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**
RMLF	**best** (X unknown)	**???**



Question:

can we optimize both *mean* and *tail* of response time?

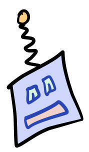
Scheduling with Heavy Tails

$t \to \infty$ limit

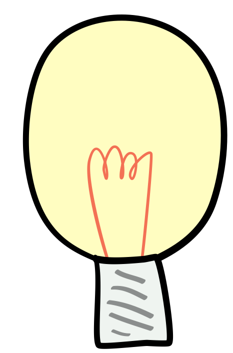
Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**
RMLF	**best** (X unknown)	**???**

Scheduling with Heavy Tails

$t \to \infty$ limit

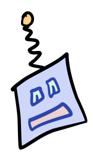


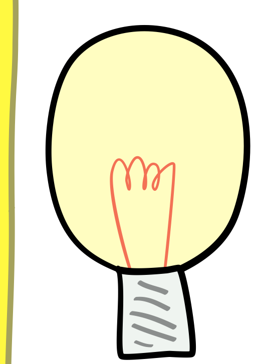
Policy	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**???**
M-SERPT	**5-approx.**	**???**
RMLF	**best** (X unknown)	**???**



Scheduling with Heavy Tails

$t \to \infty$ limit

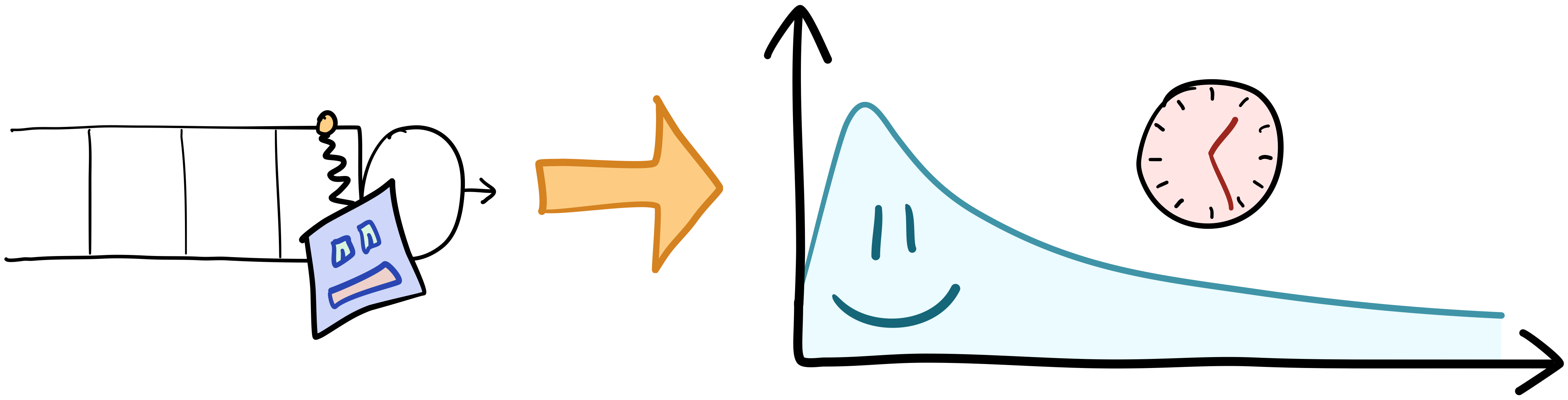
Policy 	*Mean* $\mathbf{E}[T]$	*Tail* $\mathbf{P}[T > t]$
FCFS	**bad**	**worst**
PS	**okay**	**best**
FB	**good**	**best**
Gittins	**best**	**best***
M-SERPT	**5-approx.**	**best**
RMLF	**best** (X unknown)	**best**



new!

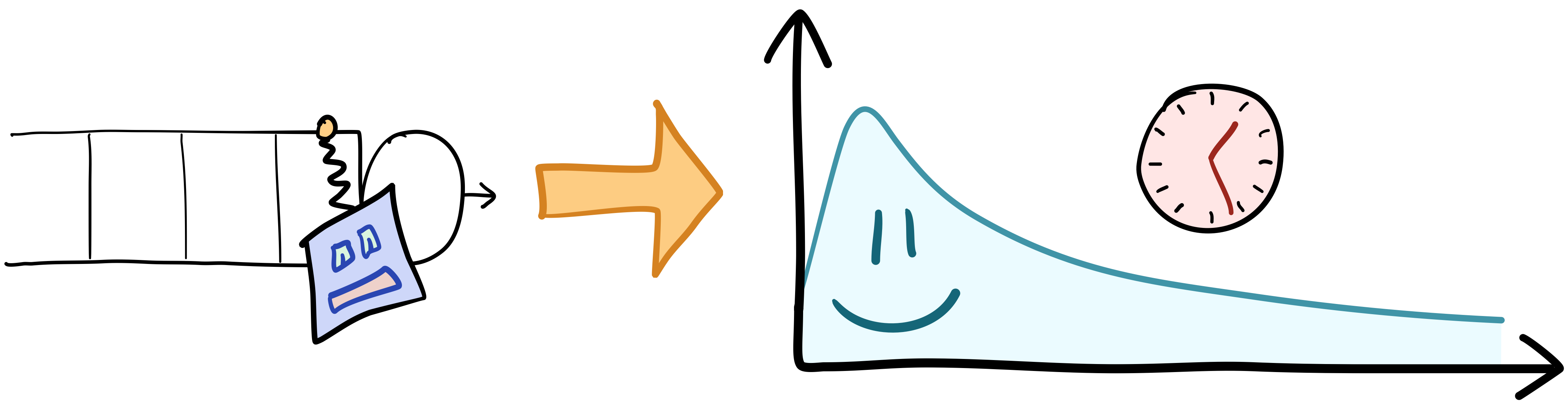
Our contribution:

a sufficient condition for
optimal response time tail



Our contribution:

a sufficient condition for
optimal response time tail

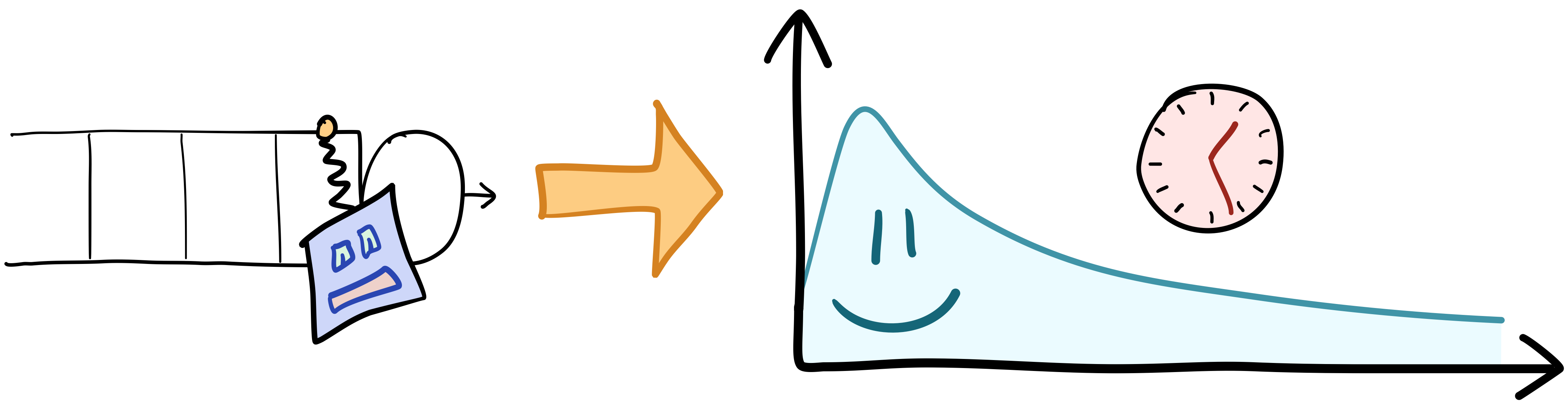


Gittins, M-SERPT,
RMLF, and more…

Our contribution:

a sufficient condition for

optimal response time tail

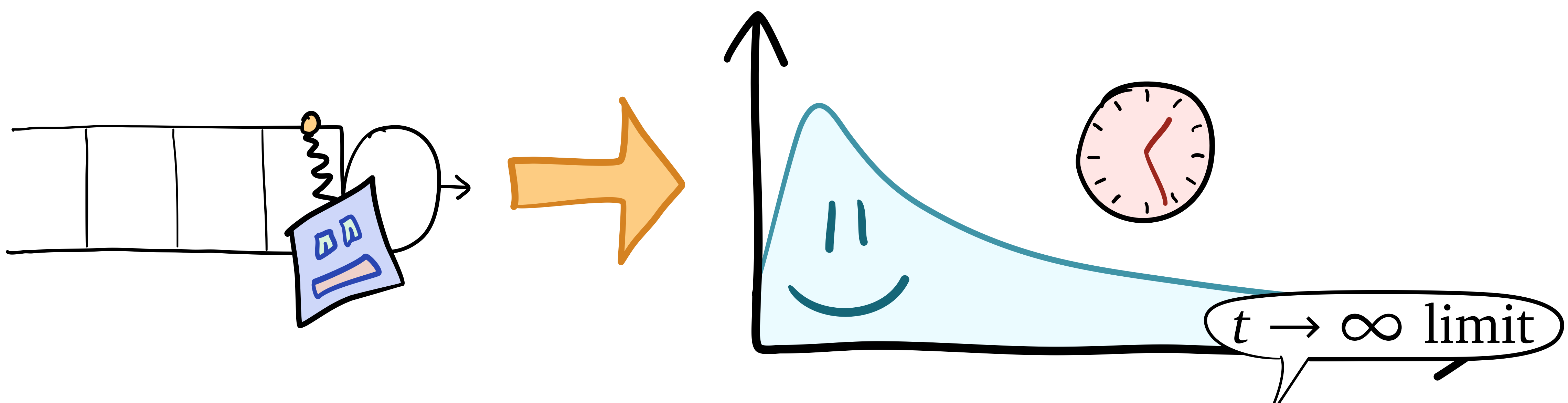


Gittins, M-SERPT, RMLF, and more…

… *all* asymptotically optimize $\mathbf{P}[T > t]$

Our contribution:

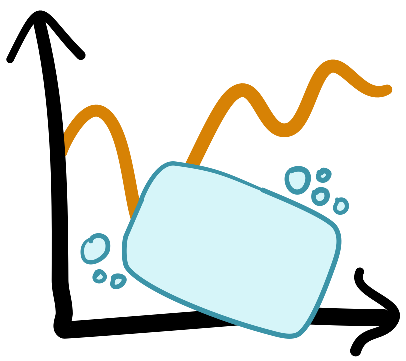
a sufficient condition for
optimal response time tail



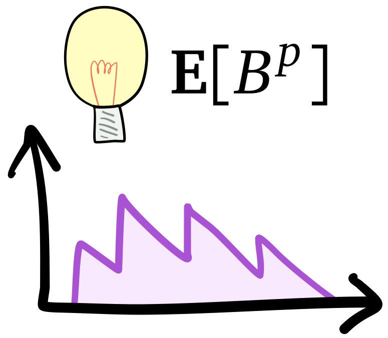
Gittins, M-SERPT, RMLF, and more…

… *all* asymptotically optimize $\mathbf{P}[T > t]$

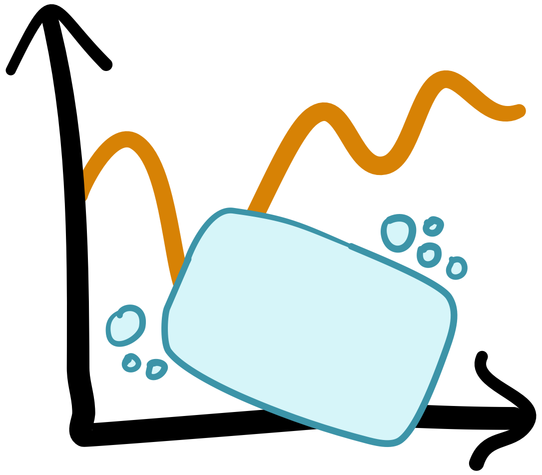
Outline



Part 1: formally state results

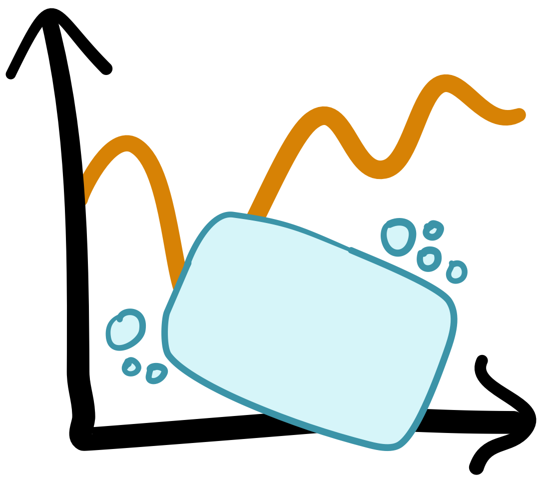


Part 2: sketch proof techniques



Part 1:

formally state results



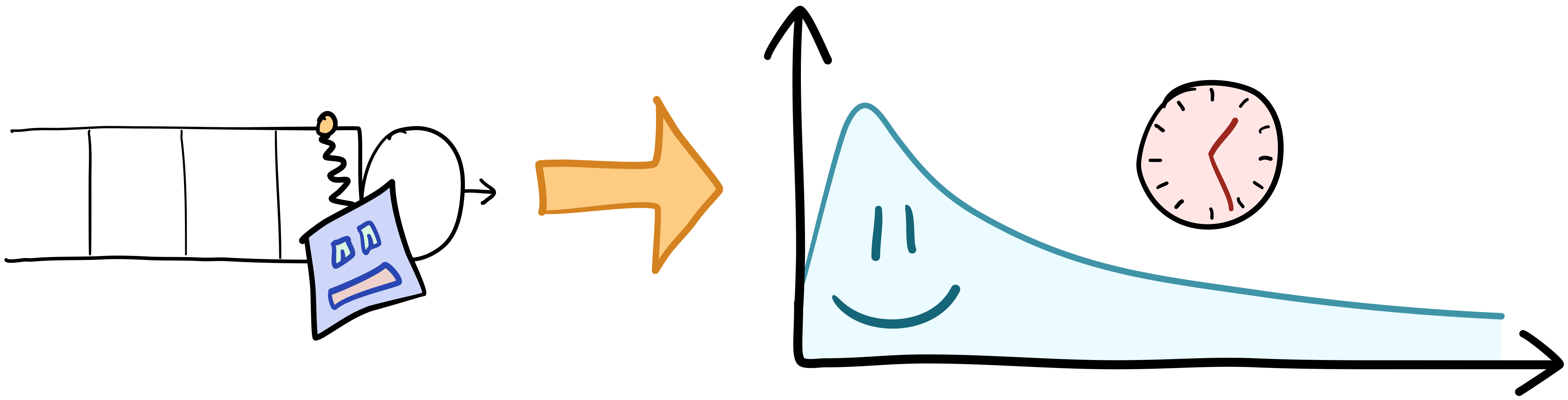
Part 1:

formally state results

easy version of

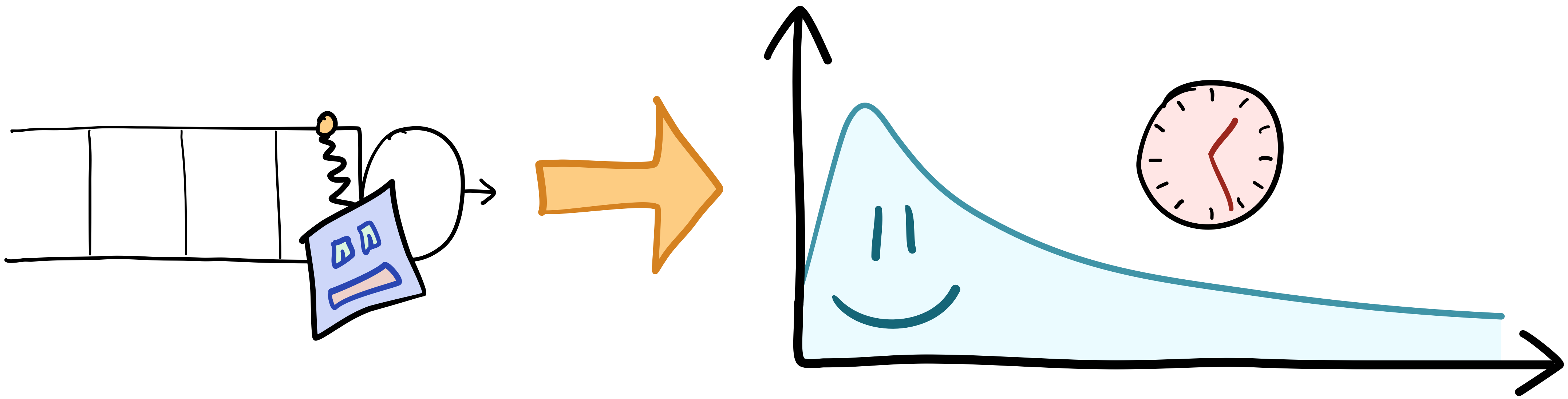
Our contribution:

a sufficient condition for
optimal response time tail



Our contribution:

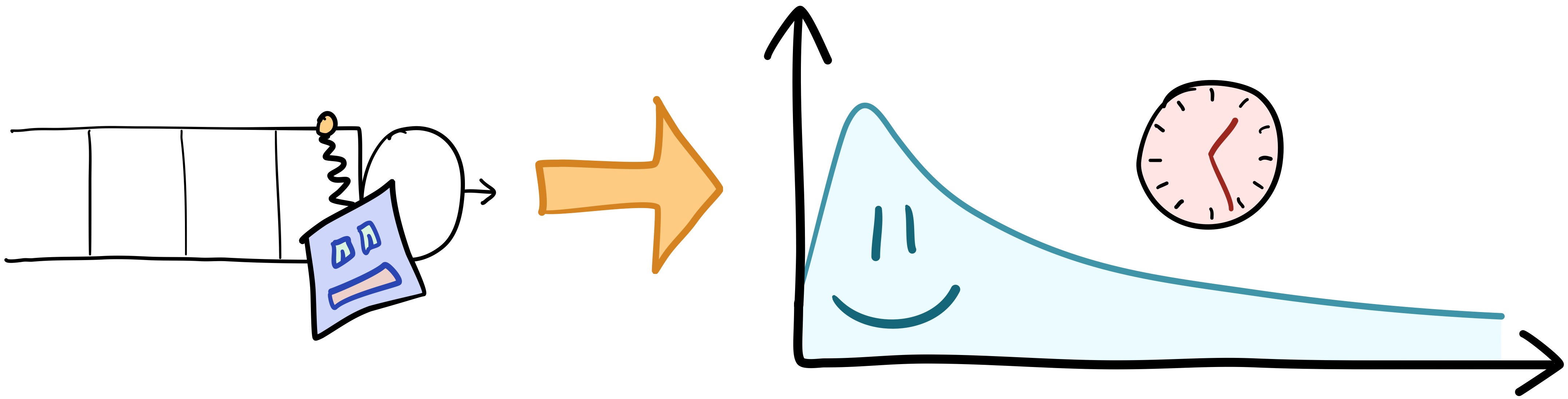
a sufficient condition for
optimal response time tail



Question: What does a sufficient condition look like?

Our contribution:

a sufficient condition for
optimal response time tail



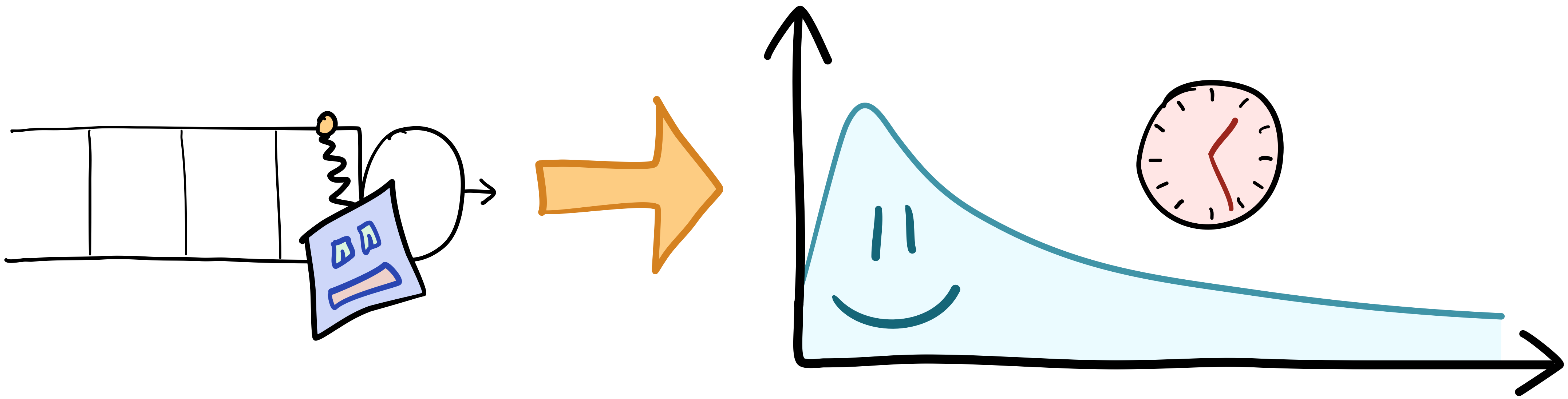
Question: What does a sufficient condition look like?

- "Don't let small jobs get stuck behind large jobs"

Our contribution:

a sufficient condition for

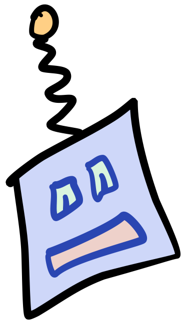
optimal response time tail



Question: What does a sufficient condition look like?

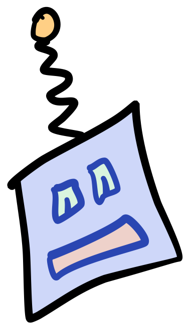
- "Don't let small jobs get stuck behind large jobs"
- How to formalize?

Describing Policies with **SOAP**

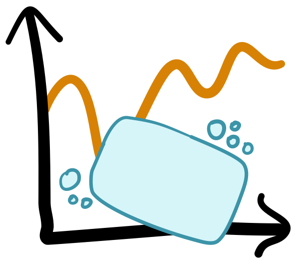


Scheduling policy:
picks which job to serve

Describing Policies with SOAP

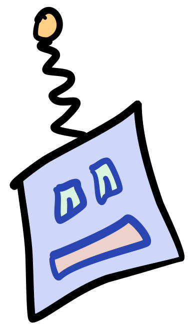


Scheduling policy:
picks which job to serve

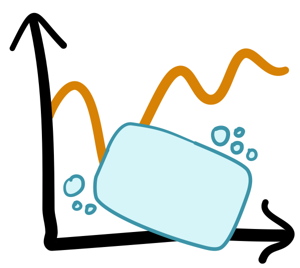


SOAP scheduling policy:
picks which job to serve
using a ***rank*** *function*

Describing Policies with SOAP



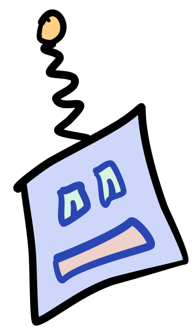
Scheduling policy:
picks which job to serve



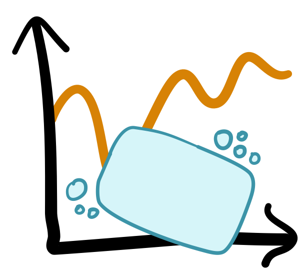
SOAP **scheduling policy**:
picks which job to serve
using a ***rank*** *function*

r : **age** → **rank**

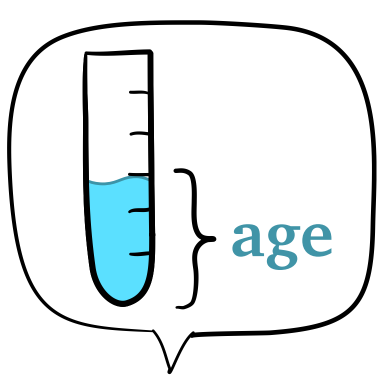
Describing Policies with SOAP



Scheduling policy:
picks which job to serve

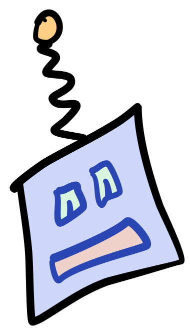


SOAP scheduling policy:
picks which job to serve
using a ***rank*** *function*

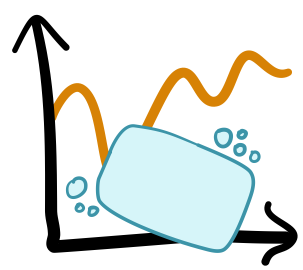


r : **age** → **rank**

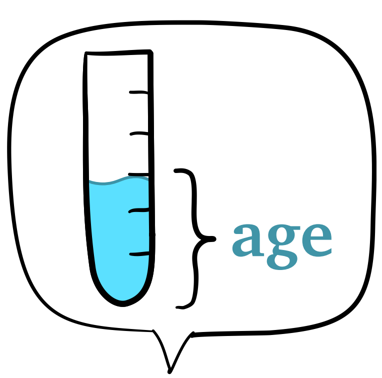
Describing Policies with SOAP



Scheduling policy:
picks which job to serve



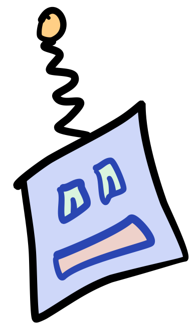
SOAP scheduling policy:
picks which job to serve
using a ***rank** function*



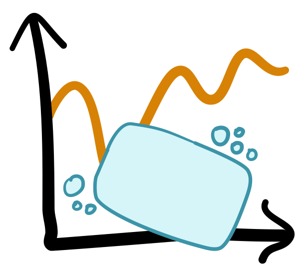
$r : \textbf{age} \to \textbf{rank}$

a job's *priority*
(lower is better)

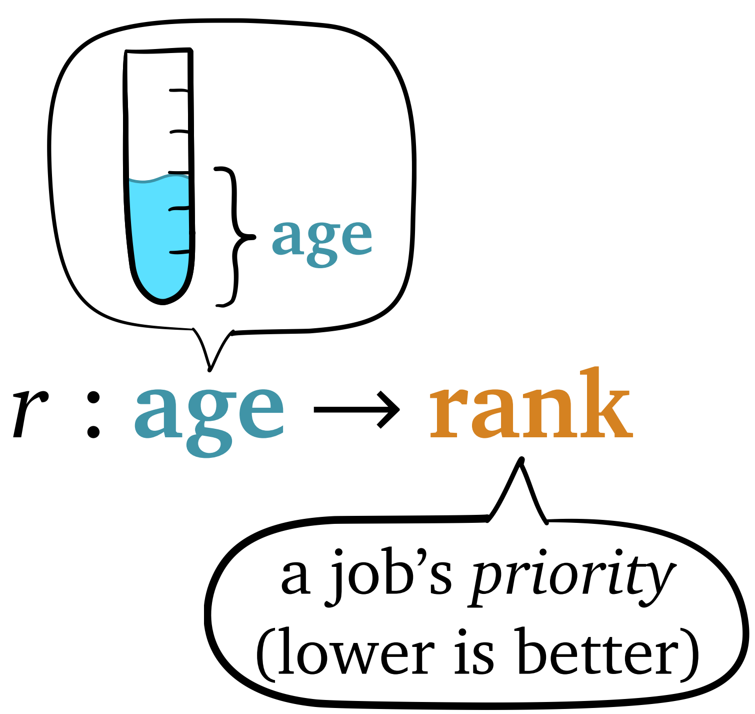
Describing Policies with SOAP



Scheduling policy:
picks which job to serve

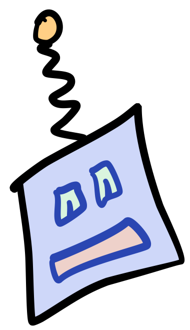


SOAP scheduling policy:
picks which job to serve
using a ***rank*** *function*

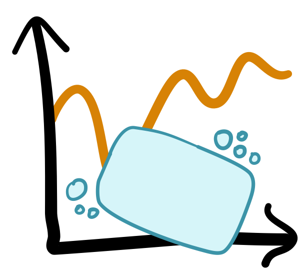


FB

Describing Policies with SOAP

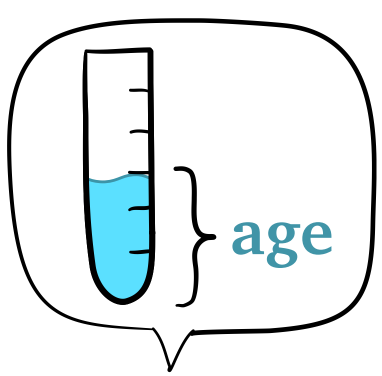


Scheduling policy:
picks which job to serve



SOAP scheduling policy:
picks which job to serve
using a ***rank*** *function*

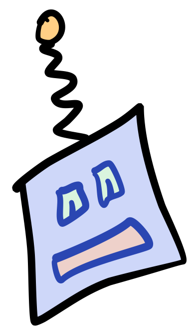
FB — serves job of least age



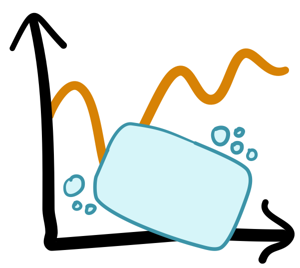
$r : \textbf{age} \to \textbf{rank}$

a job's *priority*
(lower is better)

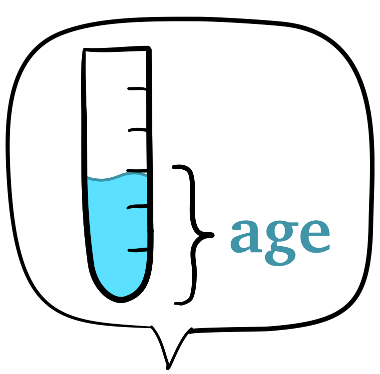
Describing Policies with SOAP



Scheduling policy:
picks which job to serve



SOAP **scheduling policy:**
picks which job to serve
using a ***rank*** *function*



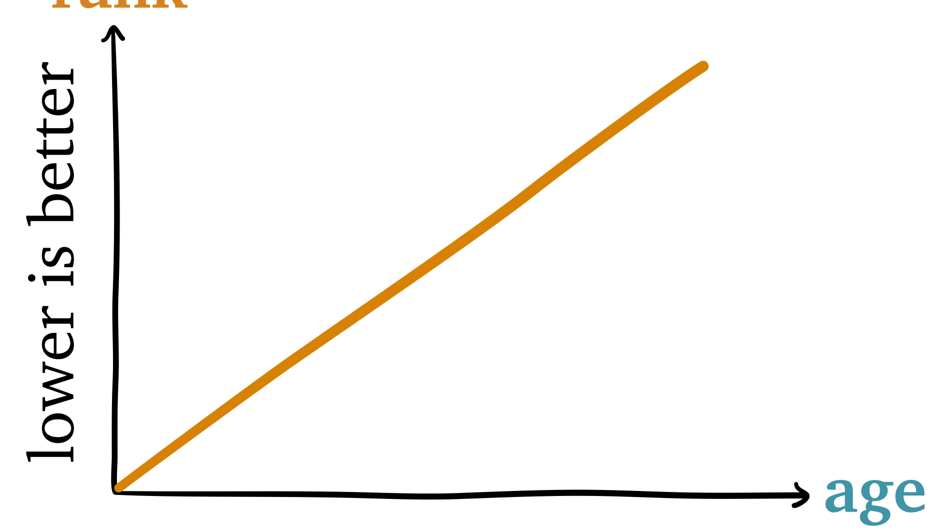
r : **age** $\rightarrow$ **rank**

a job's *priority*
(lower is better)

FB

serves job of least age

$r(a) = a$



Wide Range of SOAP Policies

One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

Wide Range of **SOAP** Policies

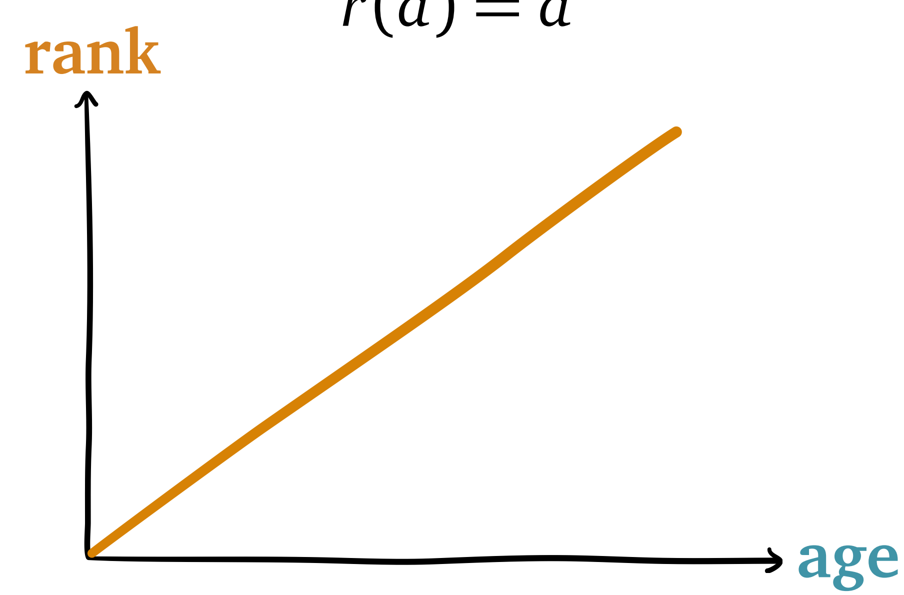
One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

FB

$r(a) = a$



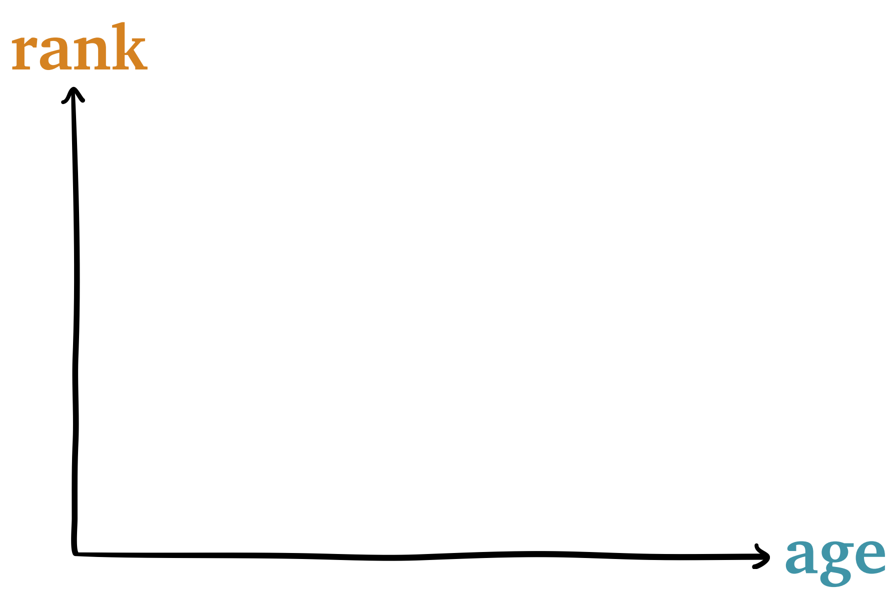
Wide Range of SOAP Policies

One rule of SOAP:

always serve job of *minimum* ***rank***

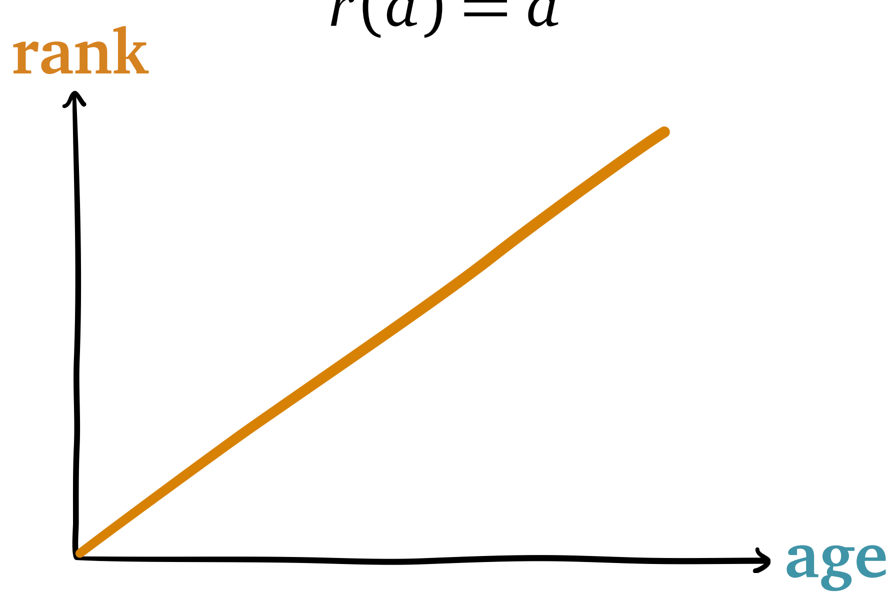
(break ties FCFS)

FCFS



FB

$r(a) = a$



Wide Range of SOAP Policies

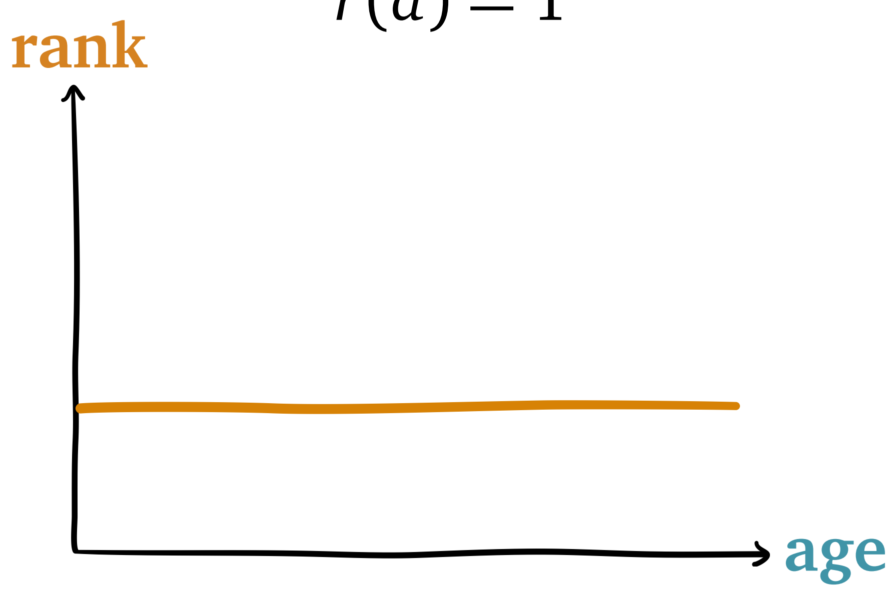
One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

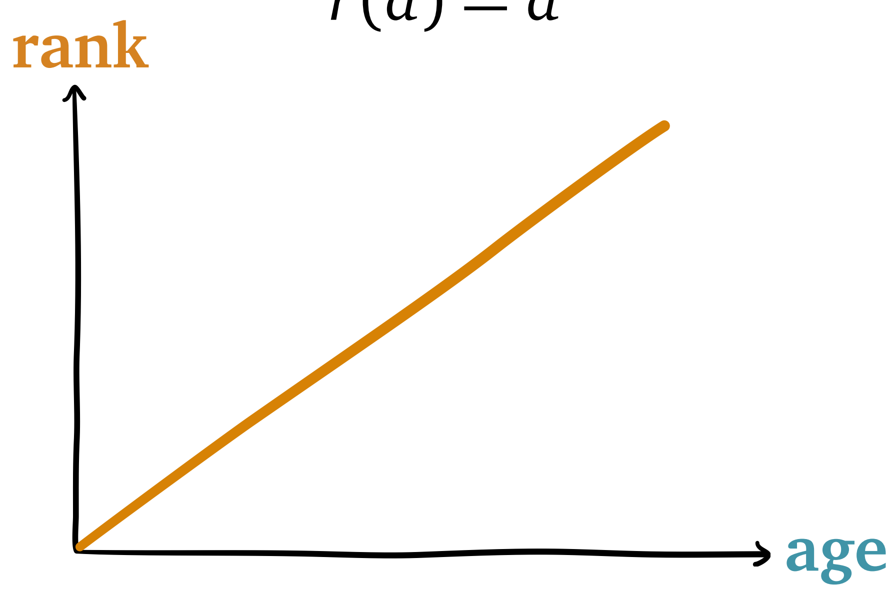
FCFS

$r(a) = 1$



FB

$r(a) = a$



Wide Range of SOAP Policies

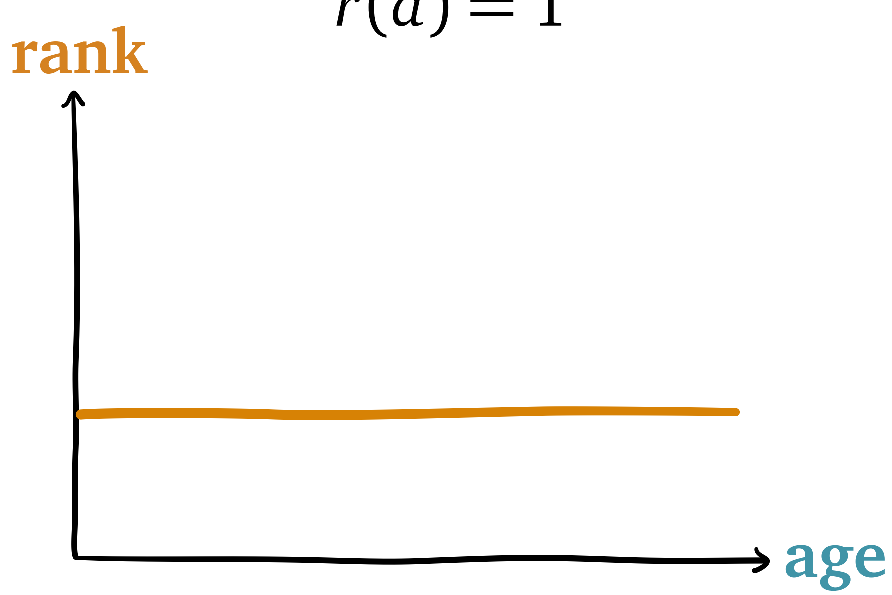
One rule of SOAP:

always serve job of *minimum* ***rank***

(break ties FCFS)

FCFS

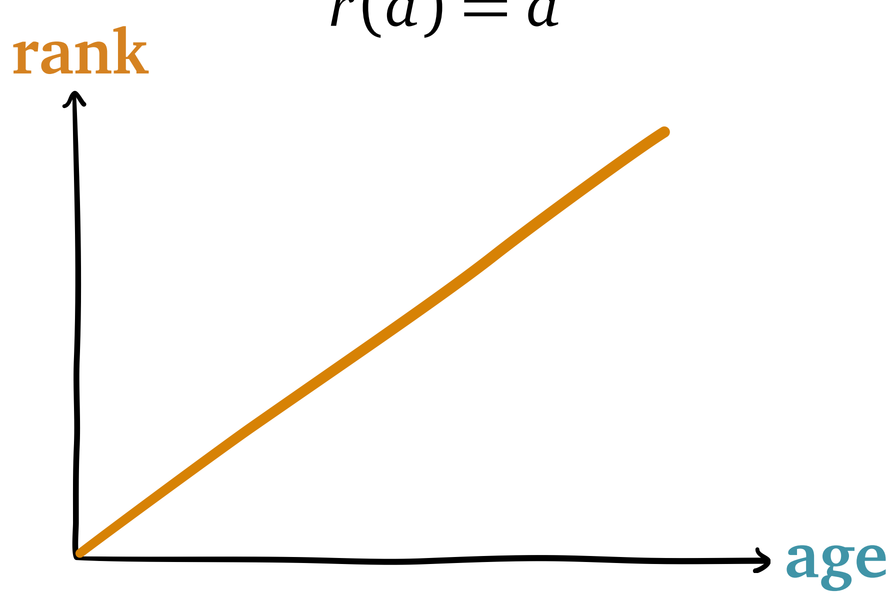
$r(a) = 1$



worst tail

FB

$r(a) = a$



best tail

Wide Range of SOAP Policies

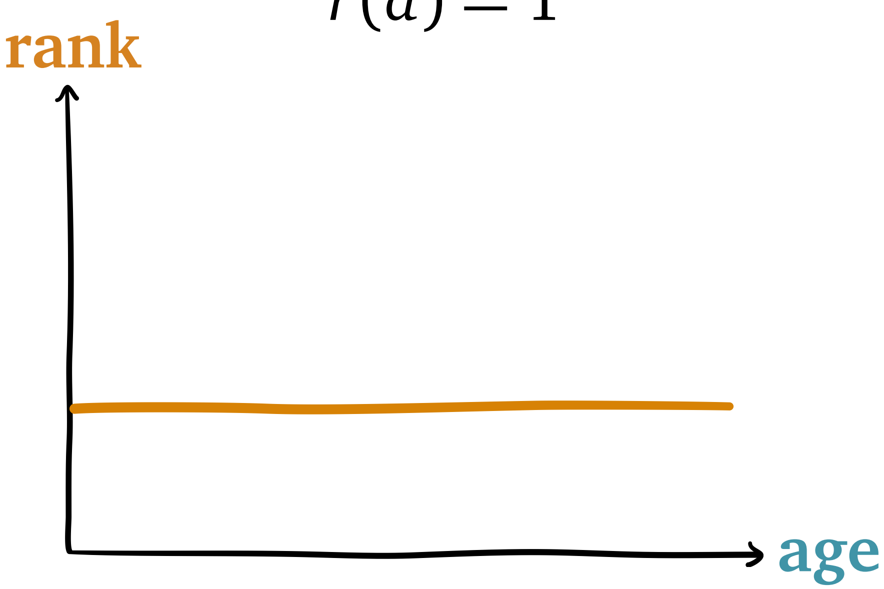
One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

FCFS

$r(a) = 1$

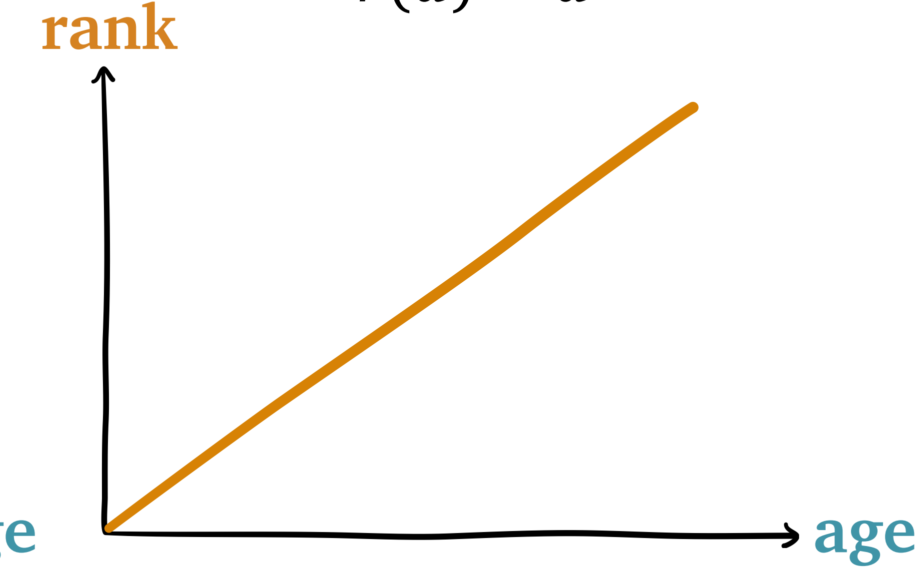


worst tail

RMLF

FB

$r(a) = a$



best tail

Wide Range of **SOAP** Policies

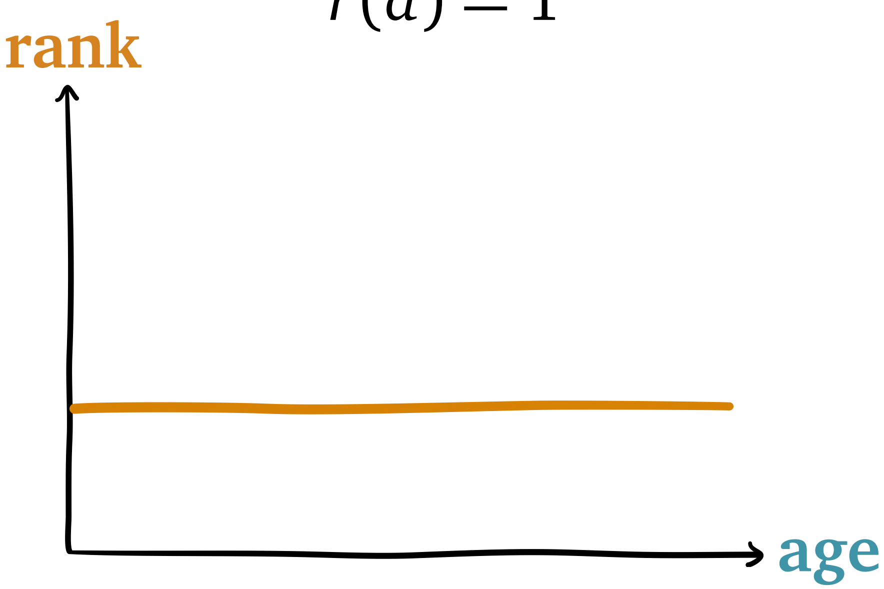
One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

FCFS

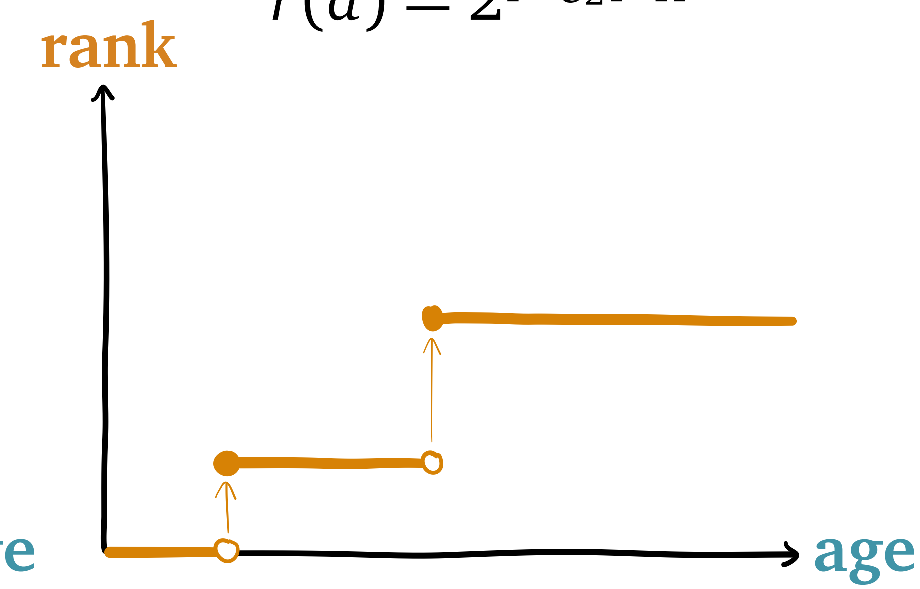
$r(a) = 1$



worst tail

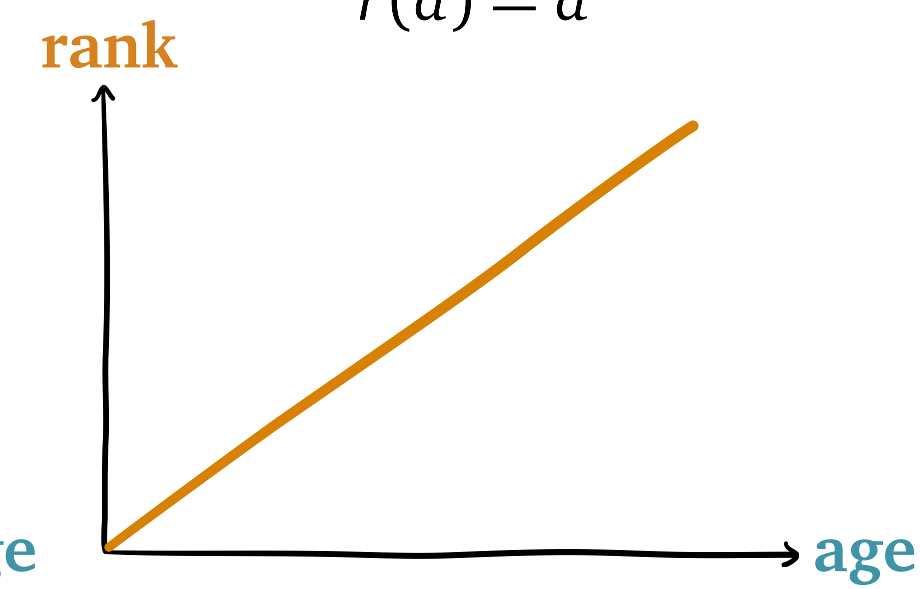
RMLF

$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$



FB

$r(a) = a$



best tail

Wide Range of **SOAP** Policies

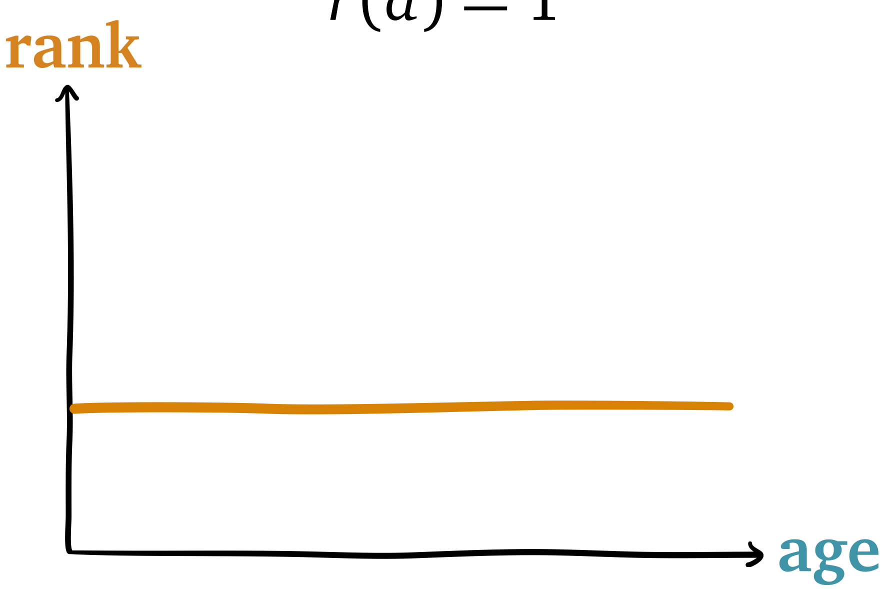
One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

FCFS

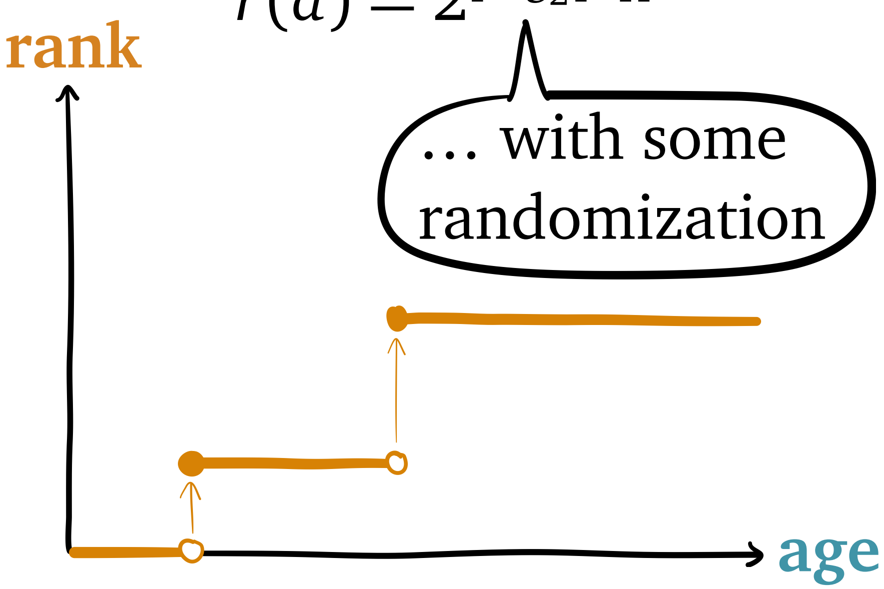
$r(a) = 1$



worst tail

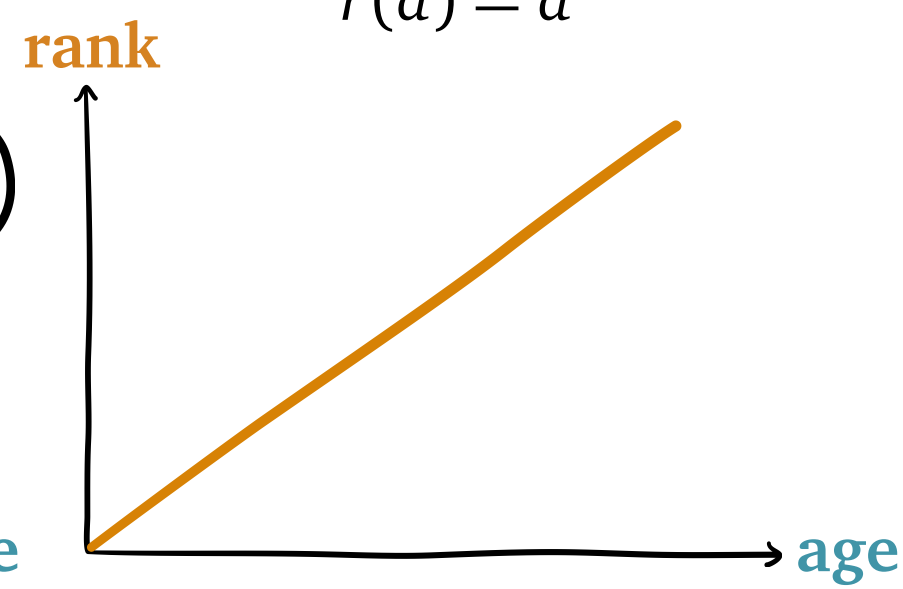
RMLF

$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$



FB

$r(a) = a$



best tail

Wide Range of SOAP Policies

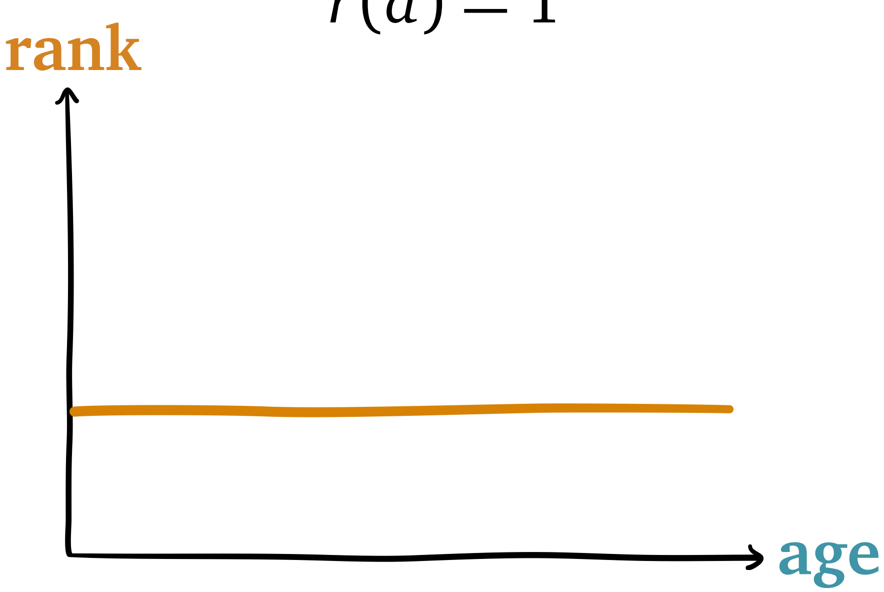
One rule of **SOAP**:

always serve job of *minimum* ***rank***

(break ties FCFS)

FCFS

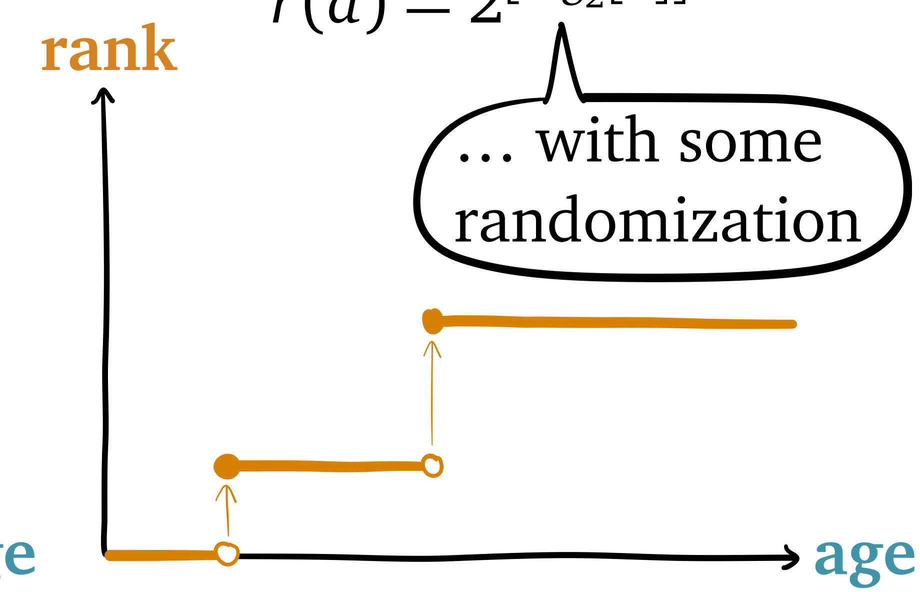
$r(a) = 1$



worst tail

RMLF

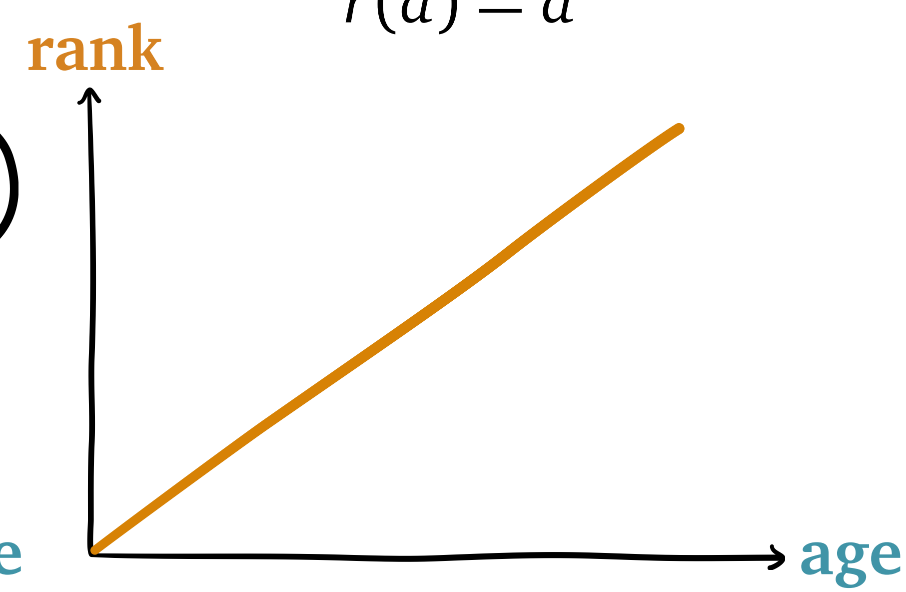
$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$



???

FB

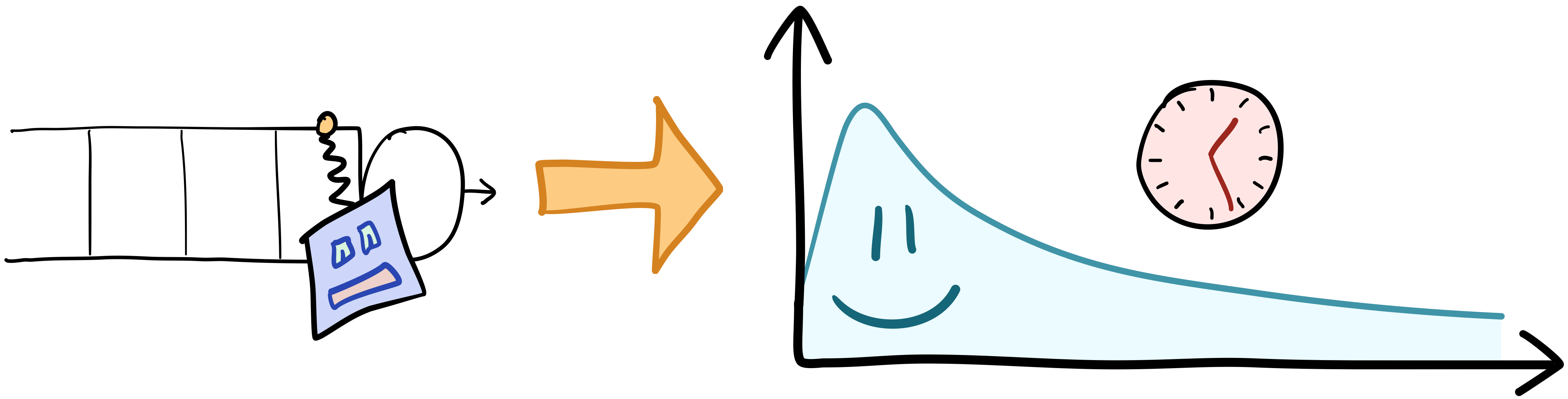
$r(a) = a$



best tail

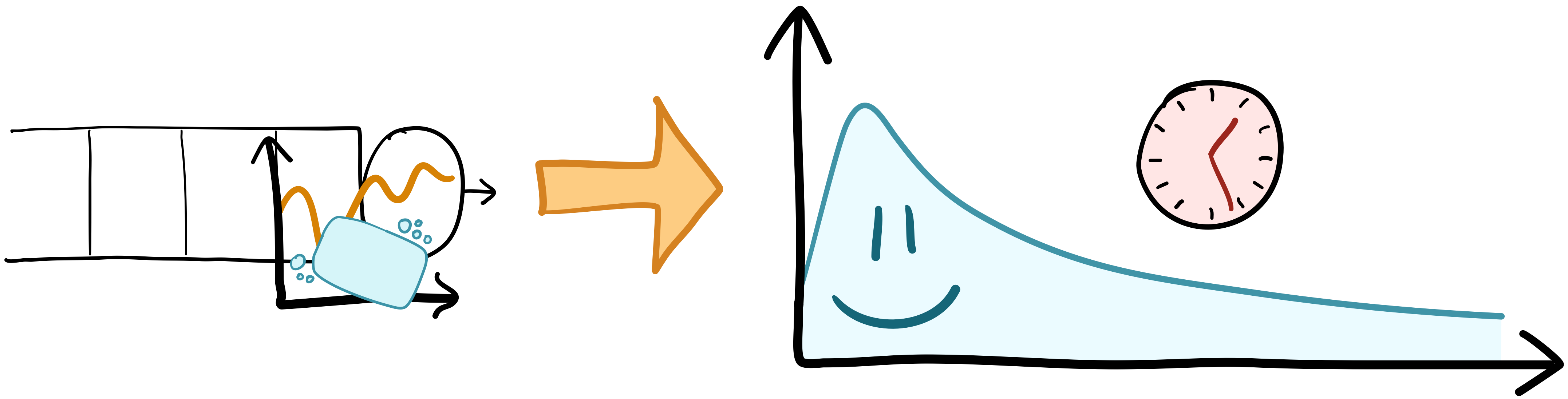
Our contribution:

a sufficient condition for
optimal response time tail



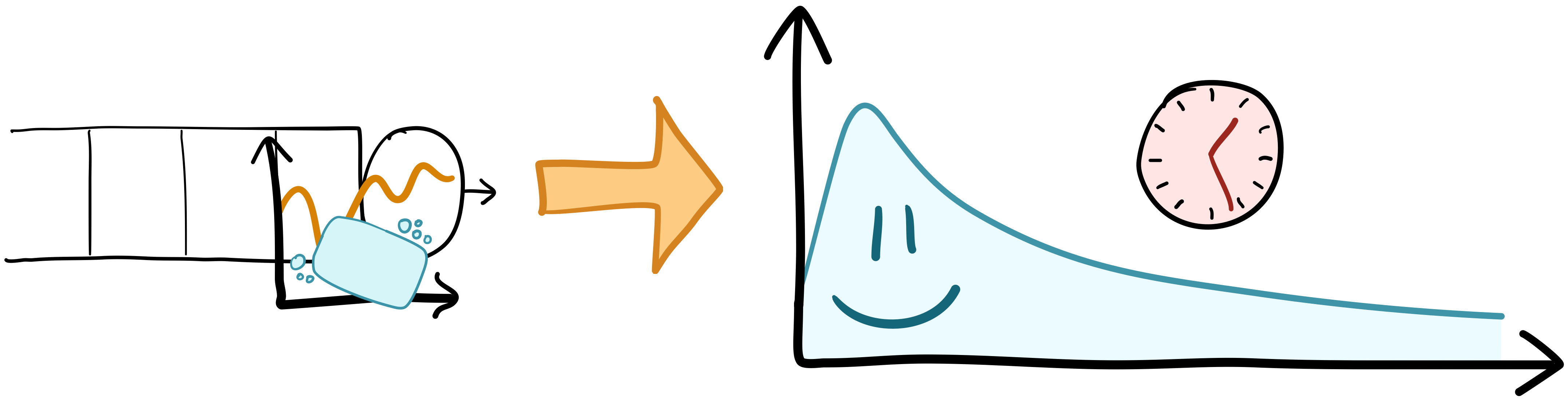
Our contribution:

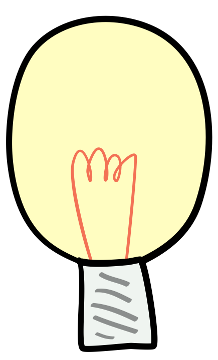
a sufficient condition for
optimal response time tail



Our contribution:

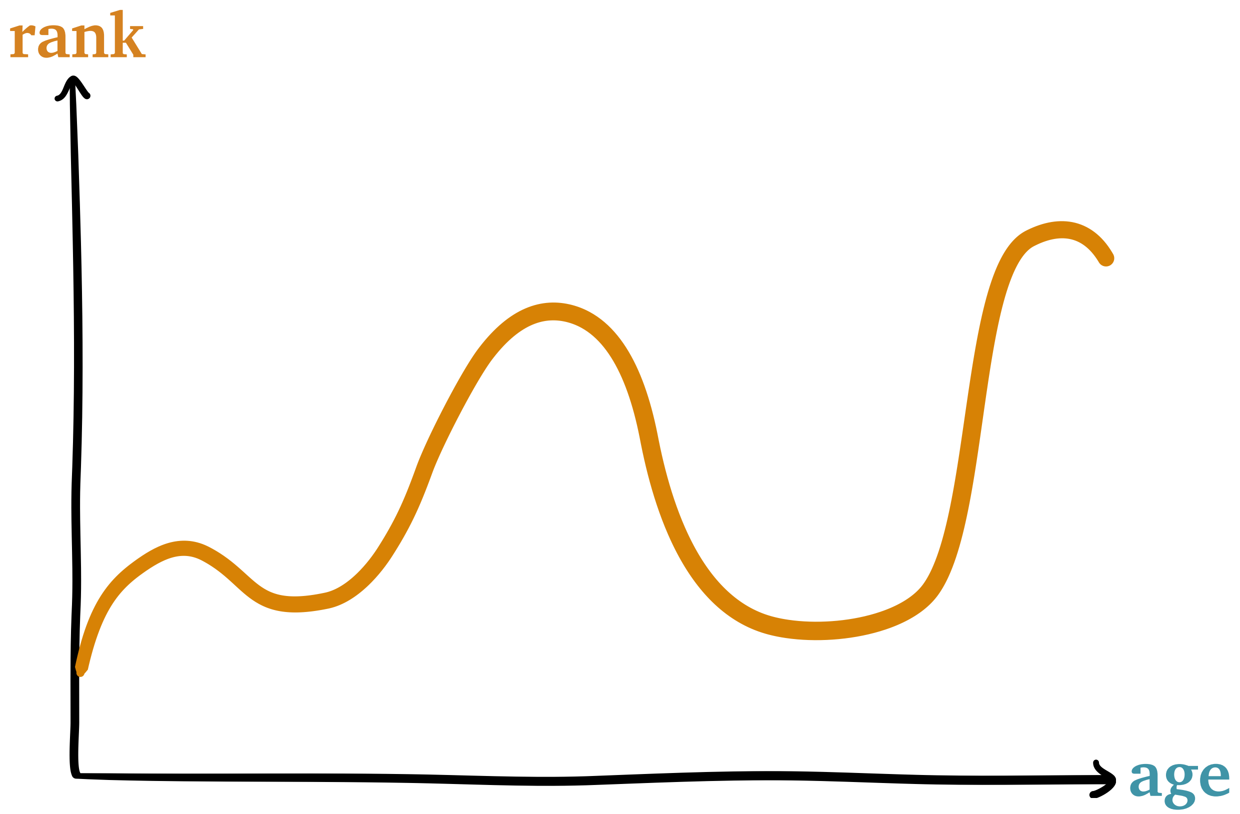
a sufficient condition for
optimal response time tail



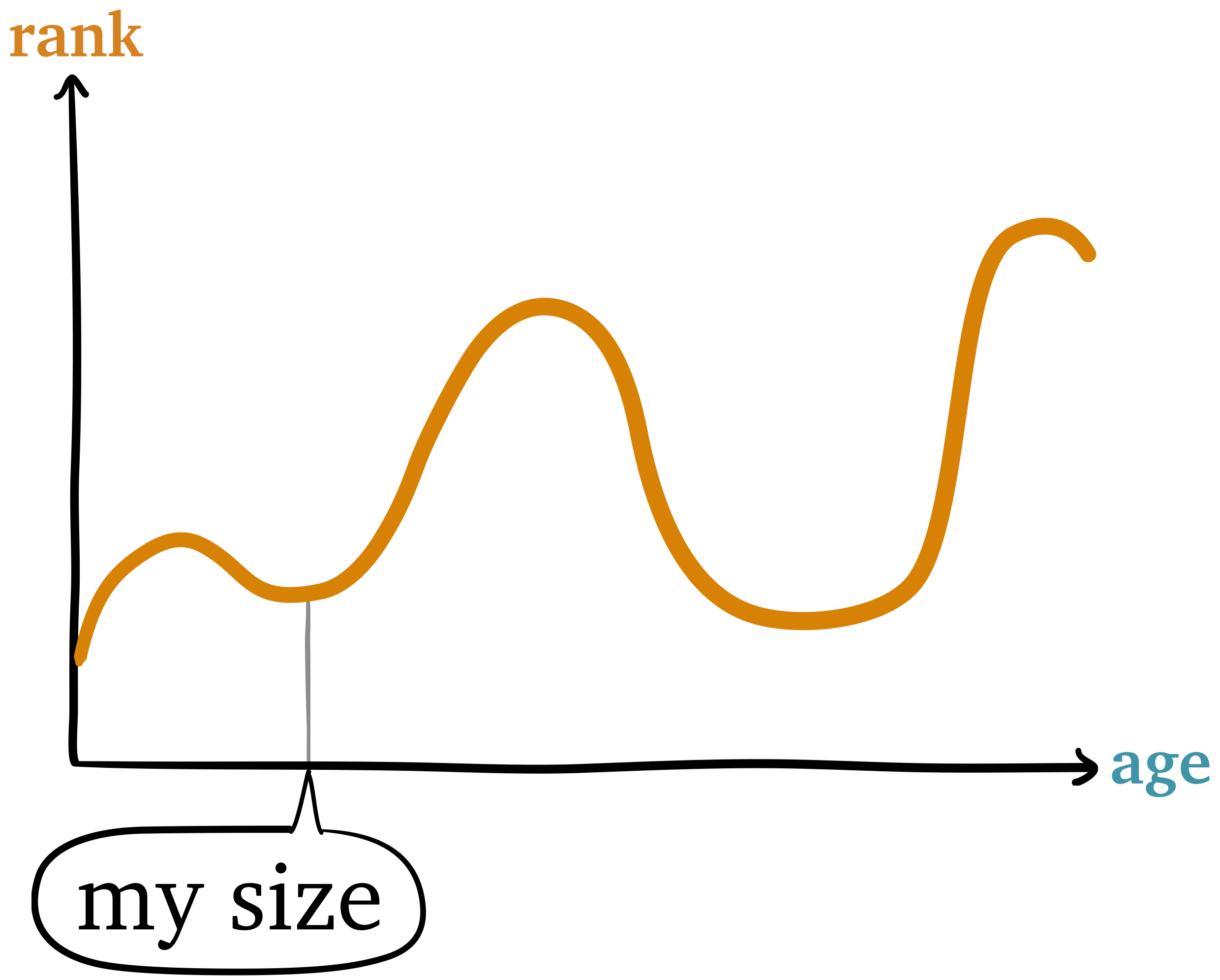


For **SOAP** policies: want a
condition on the ***rank*** *function*

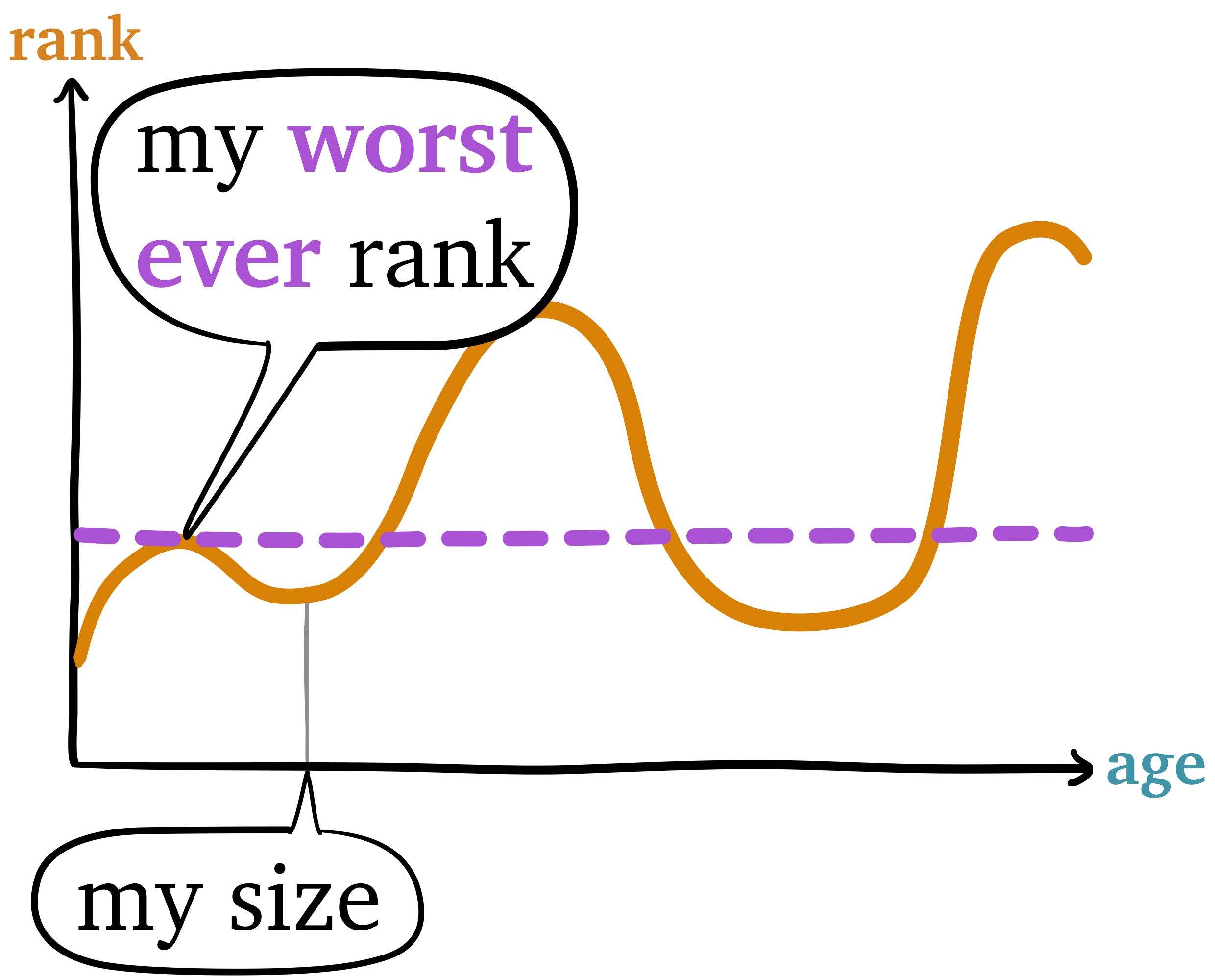
Sufficient Condition



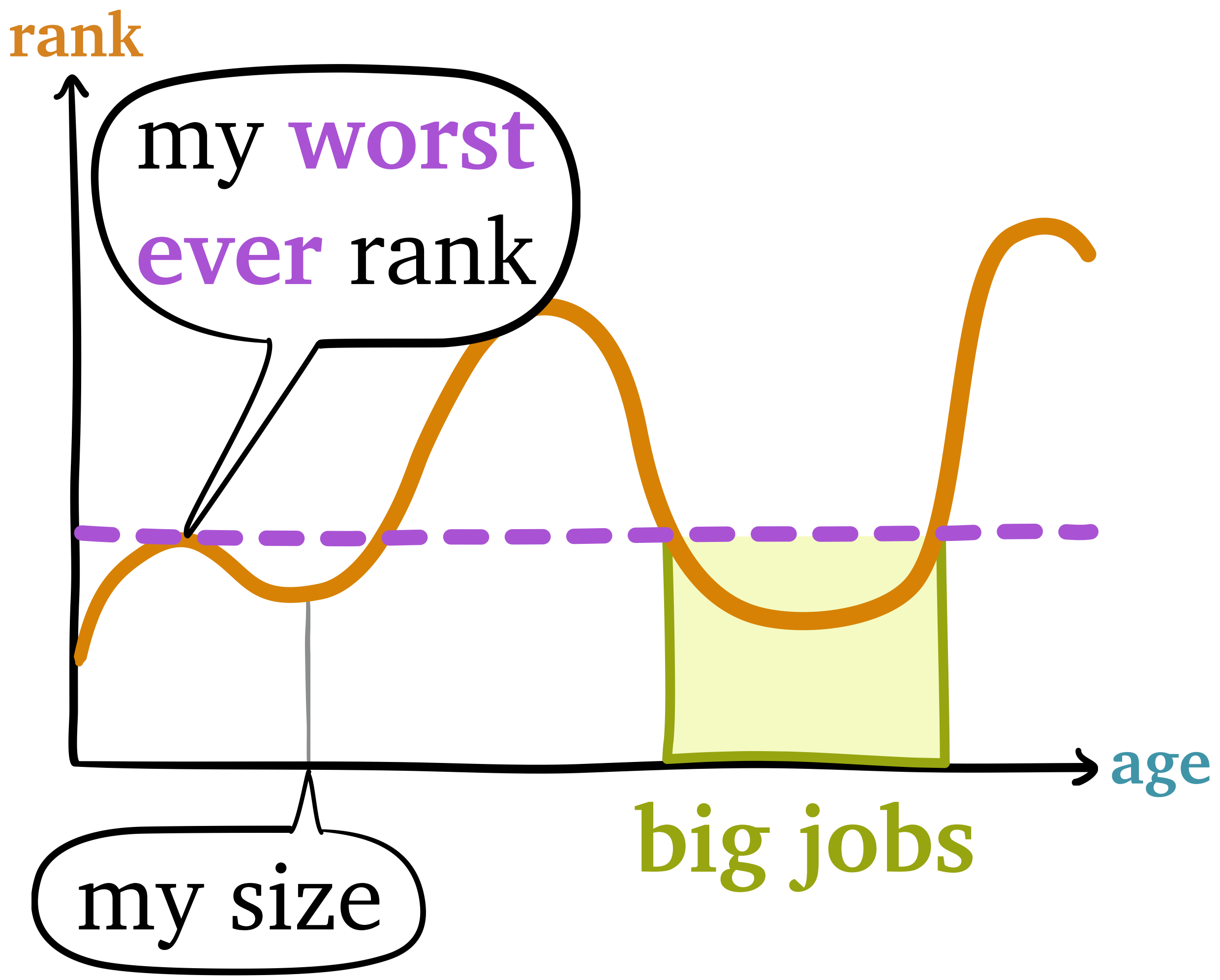
Sufficient Condition



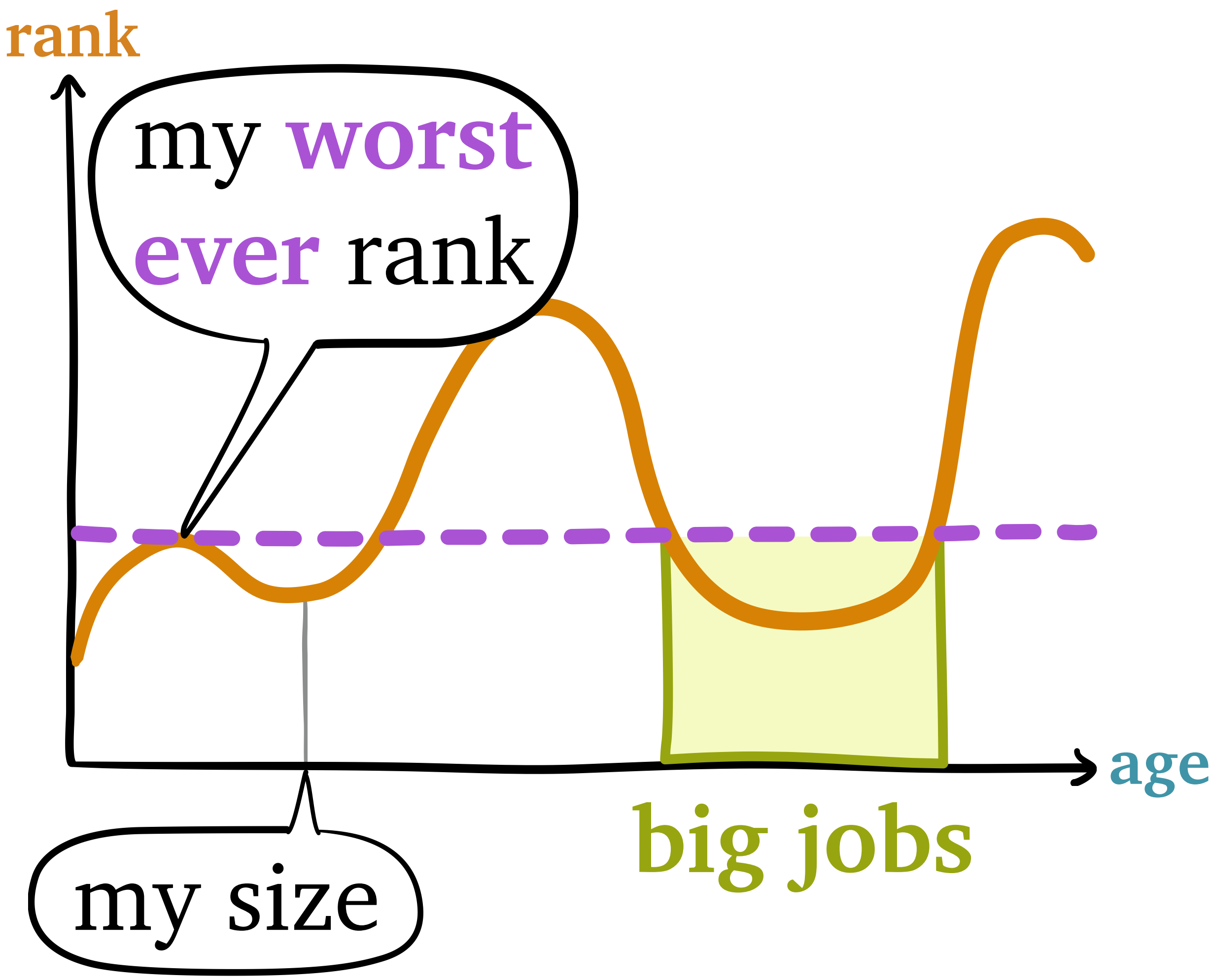
Sufficient Condition

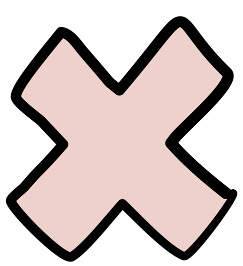


Sufficient Condition



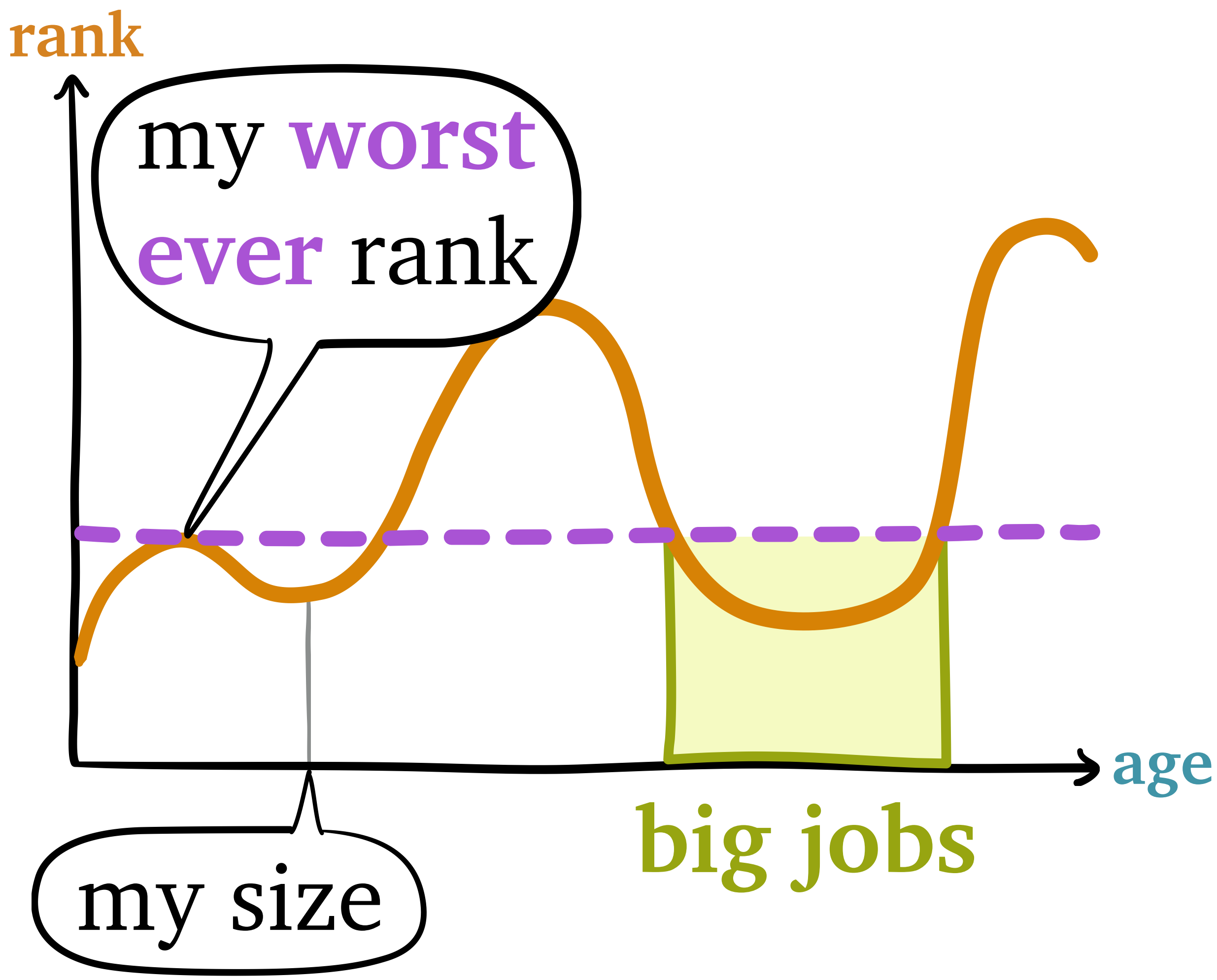
Sufficient Condition

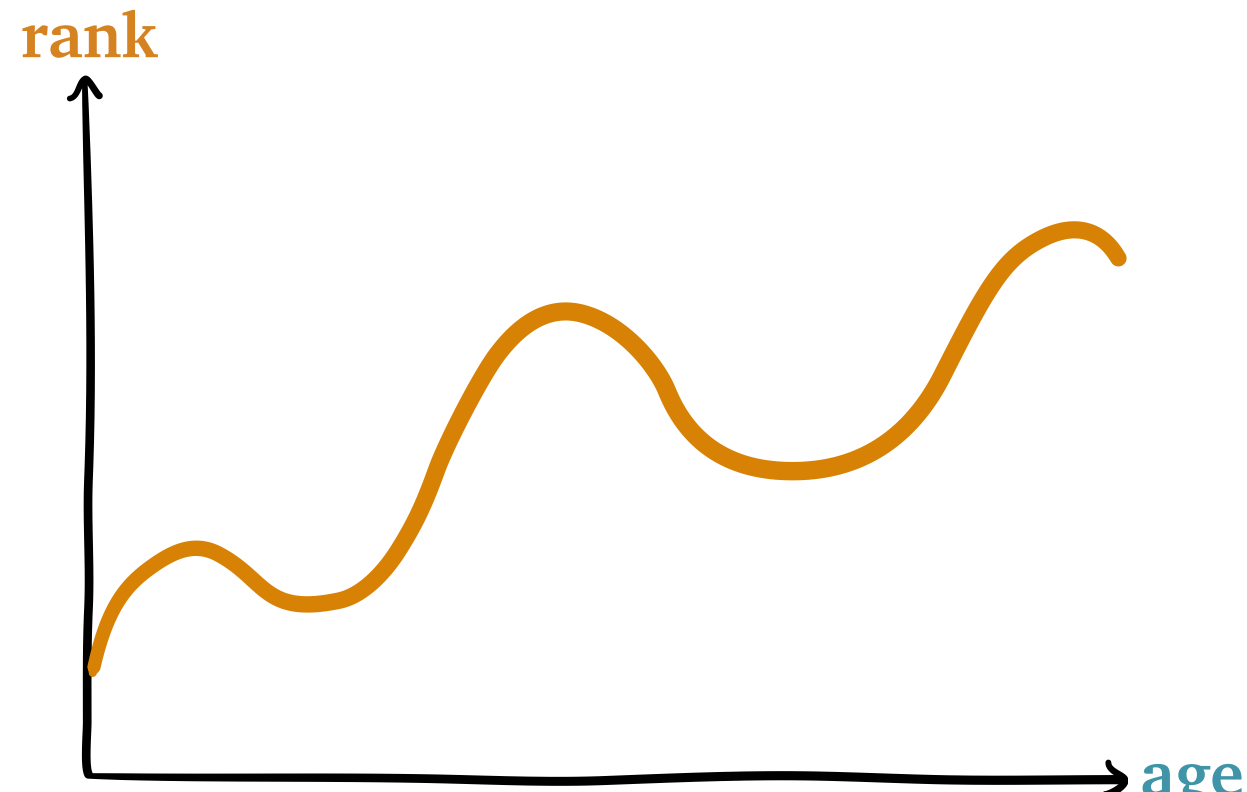




Big jobs get in my way!

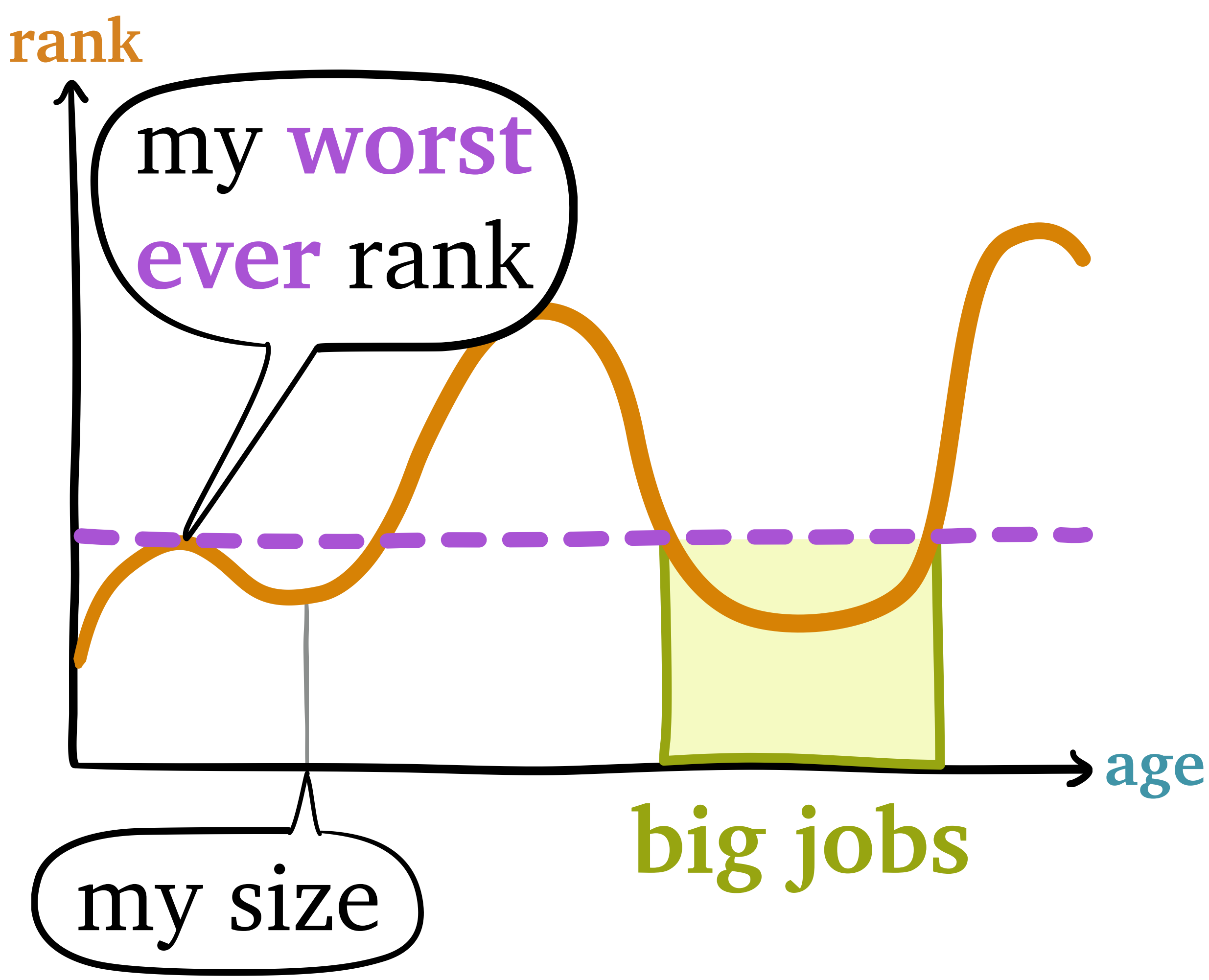
Sufficient Condition

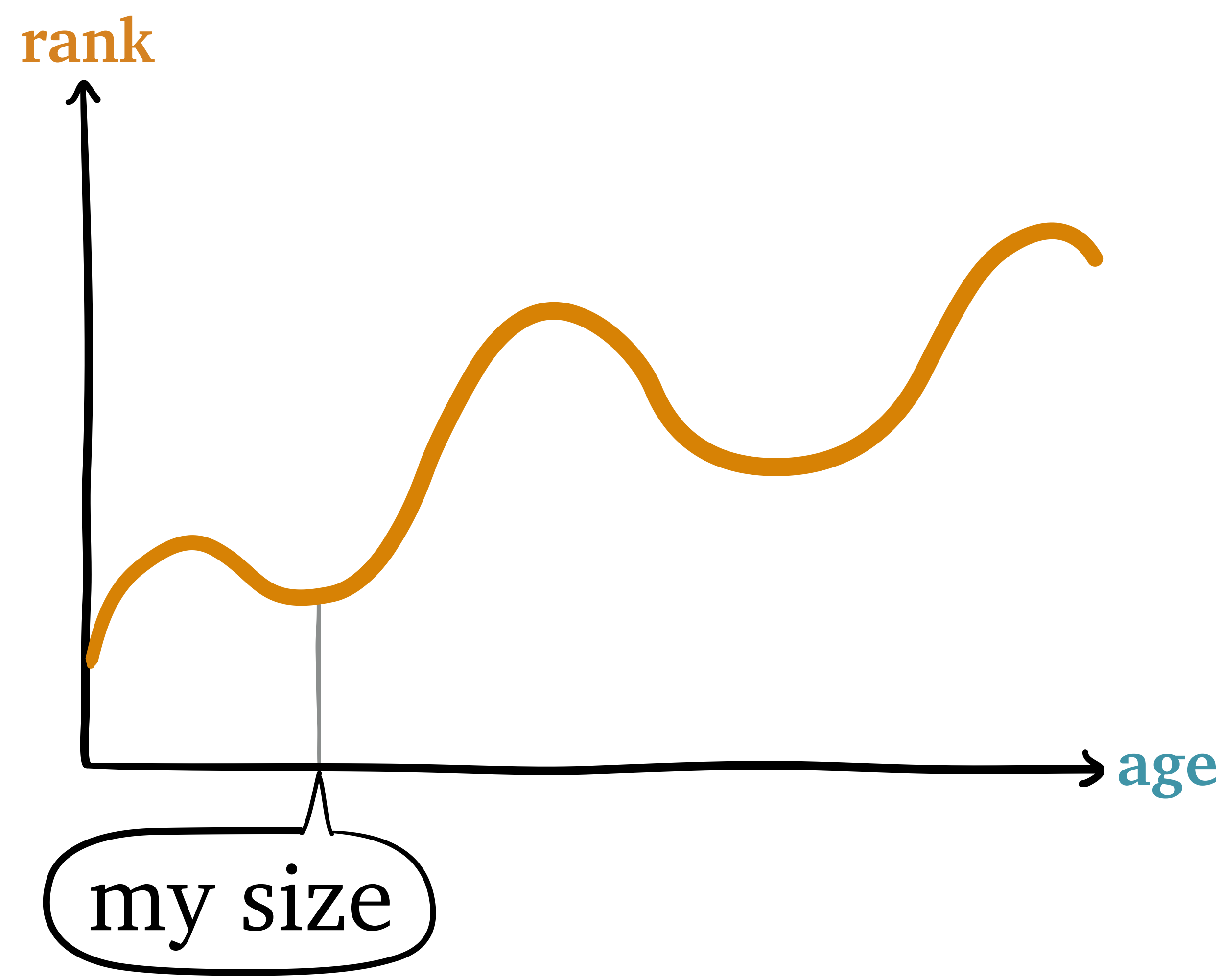


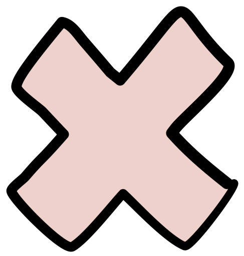


Big jobs get in my way!

Sufficient Condition

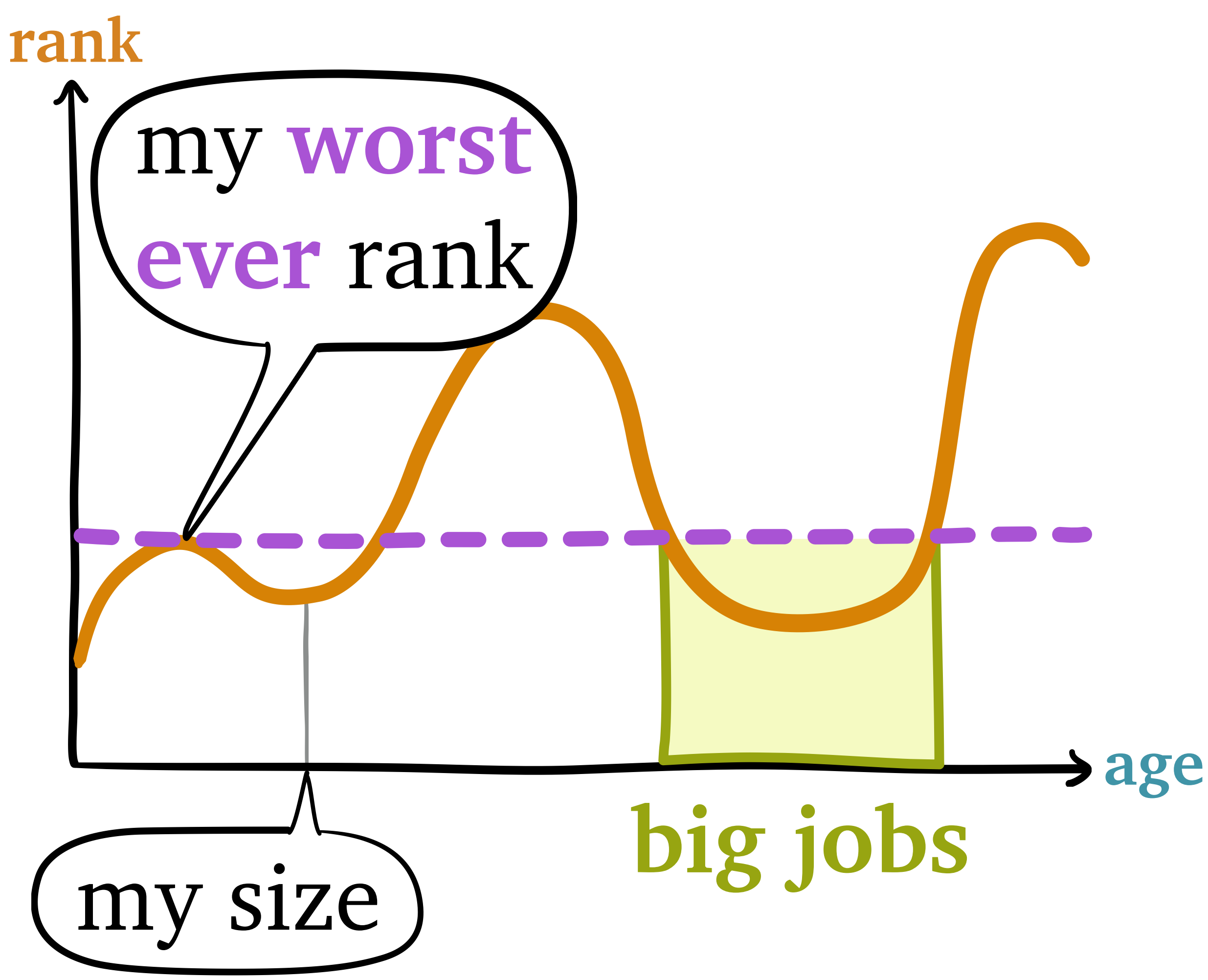


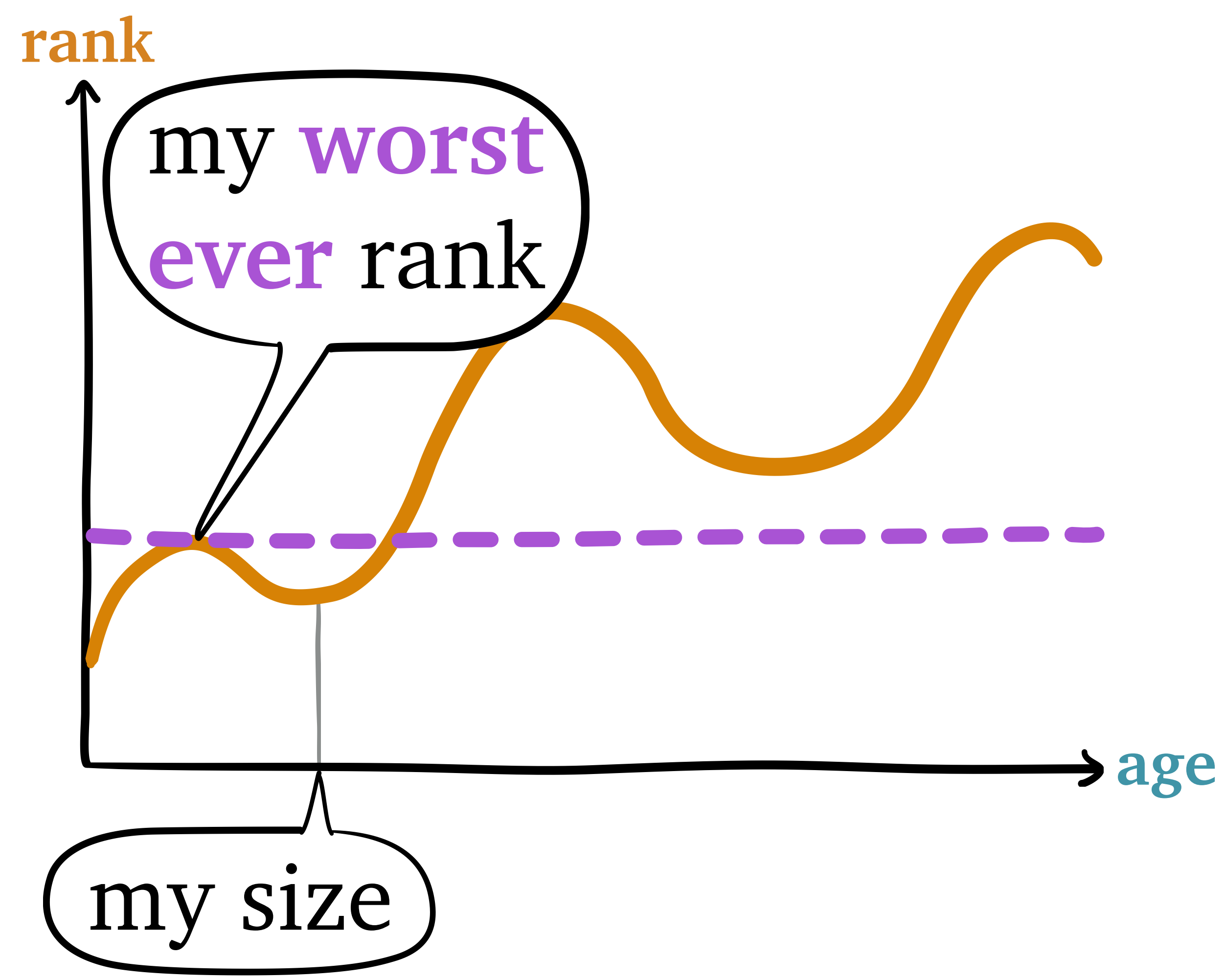




Big jobs get in my way!

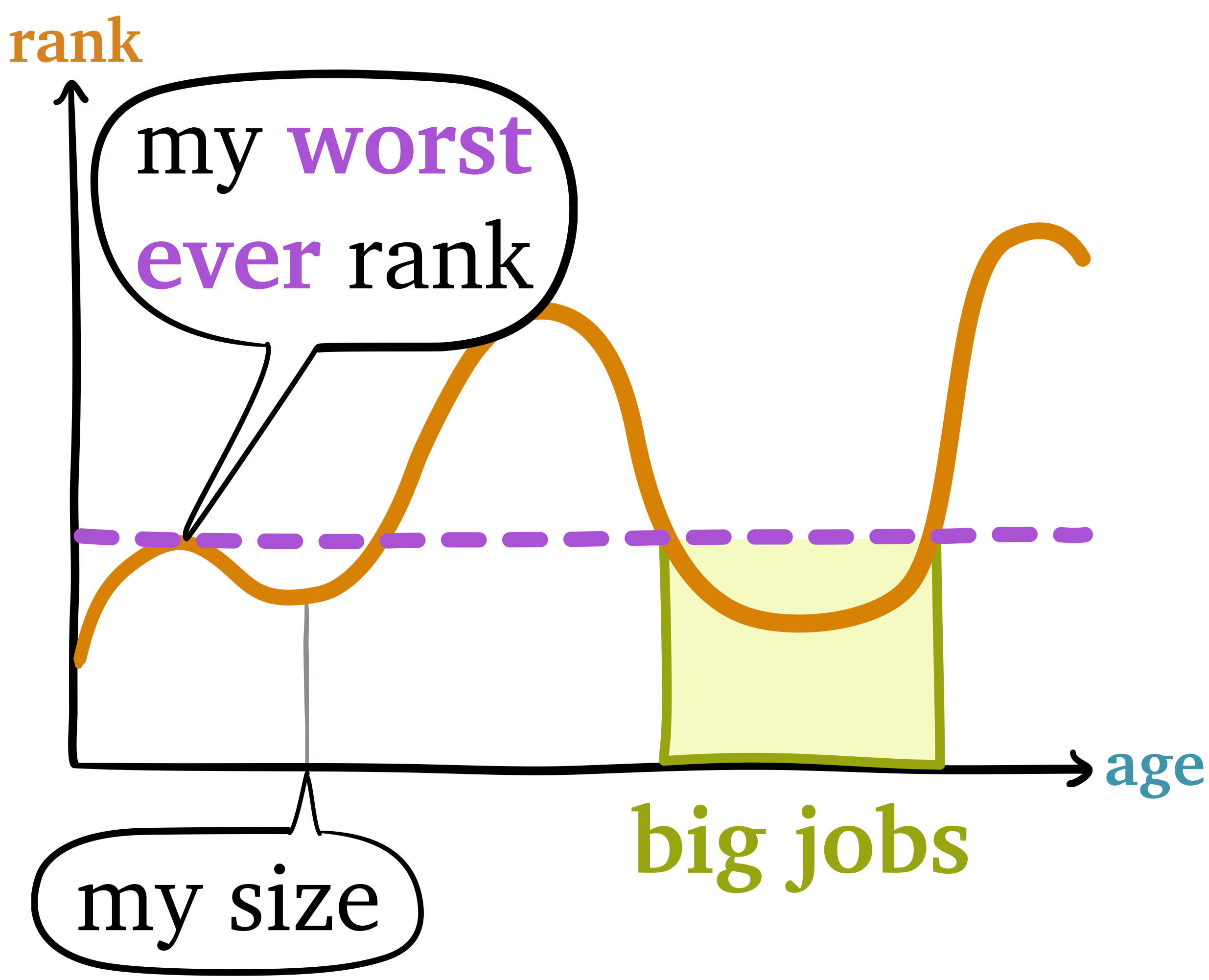
Sufficient Condition

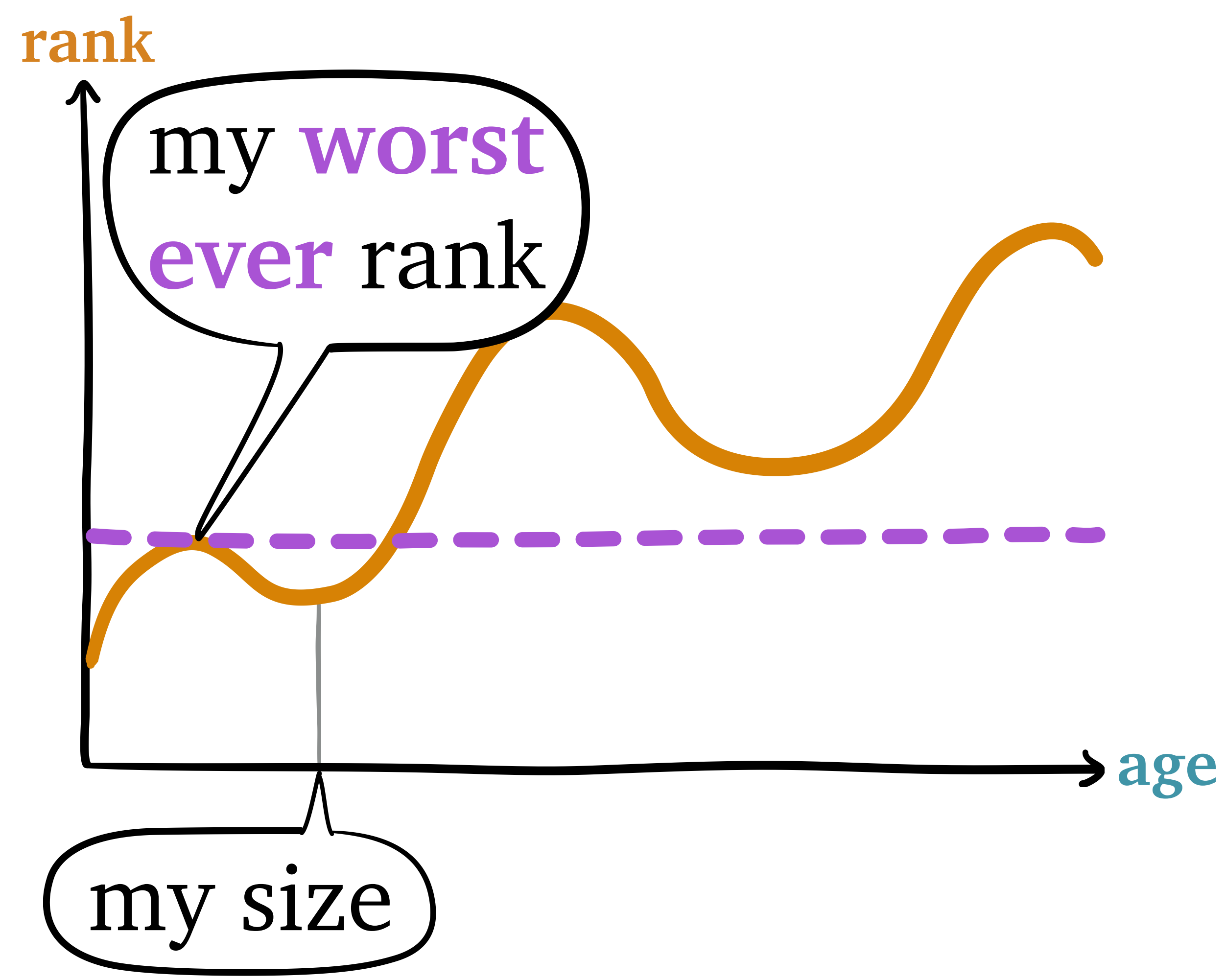




Big jobs get
in my way!

Sufficient Condition

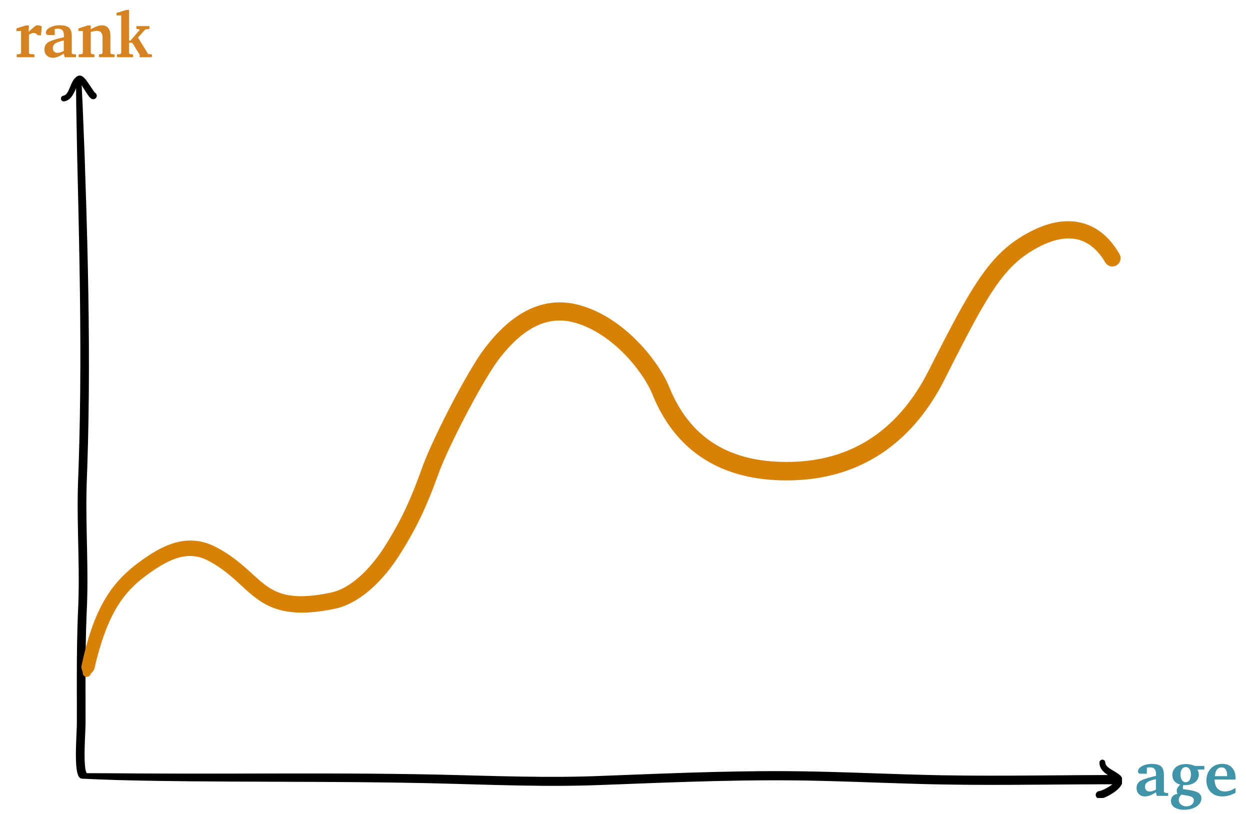


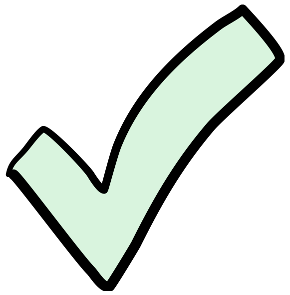


Big jobs get in my way!

No big jobs bothering me

Sufficient Condition



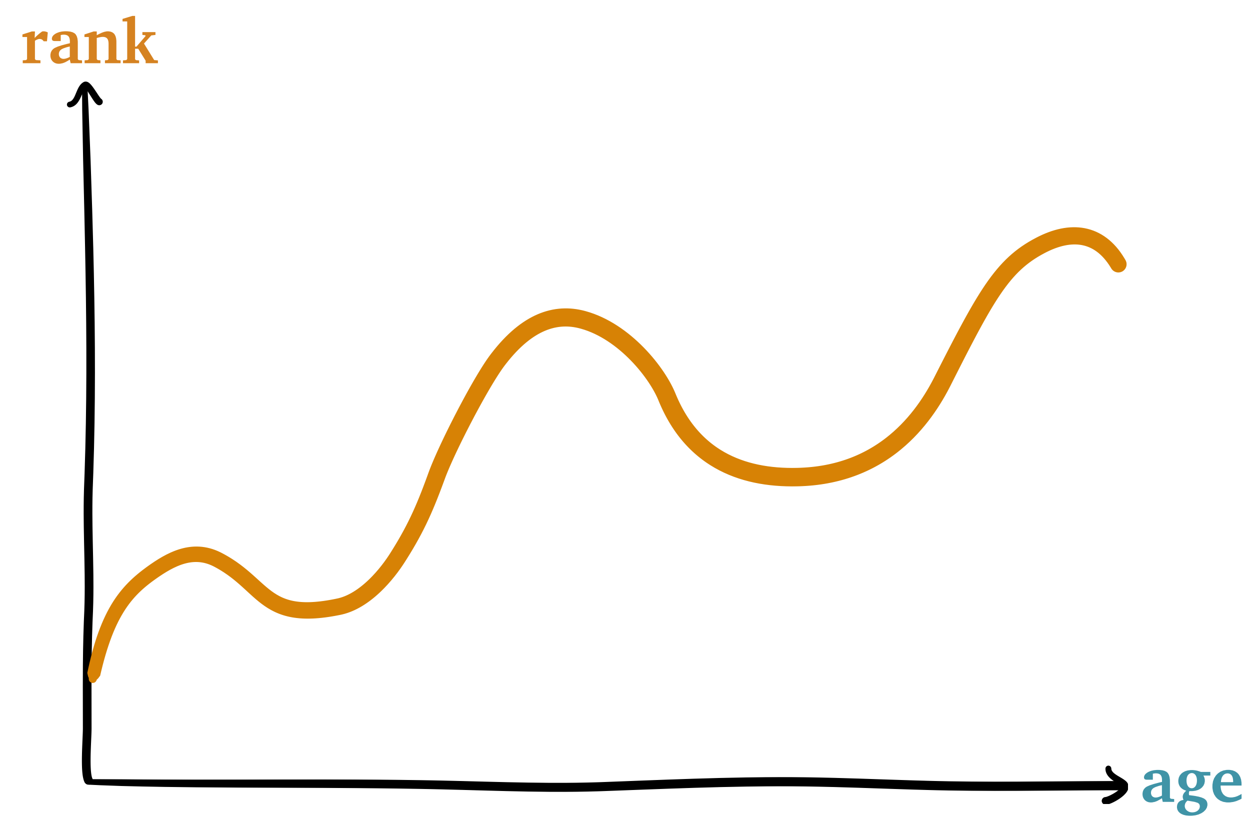


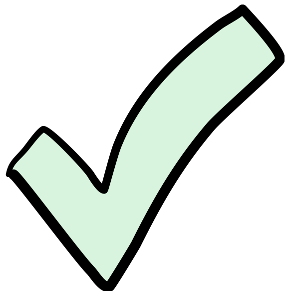
No **big jobs** bothering me

Sufficient Condition

Suppose for some $\delta \geq \gamma > 0$:

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta})$$



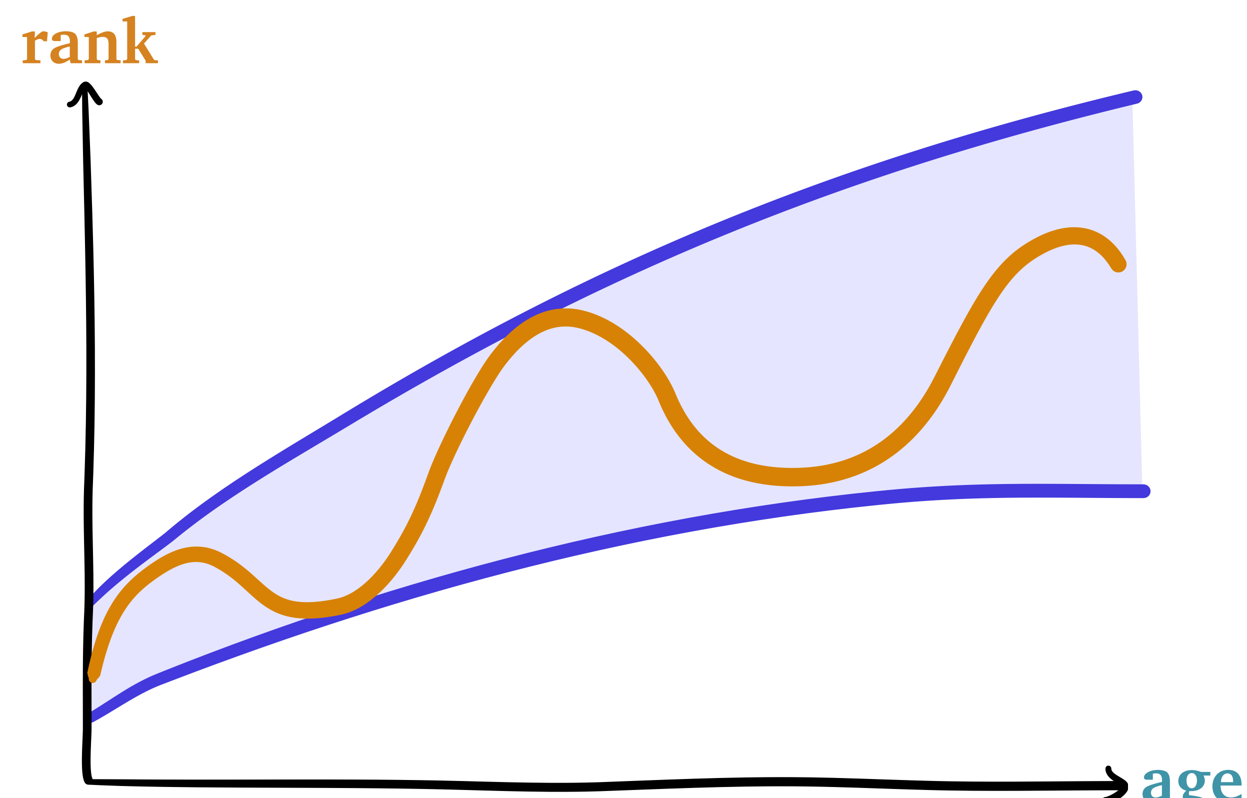


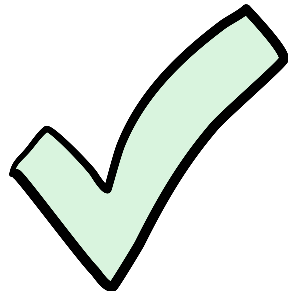
No **big jobs** bothering me

Sufficient Condition

Suppose for some $\delta \geq \gamma > 0$:

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta})$$



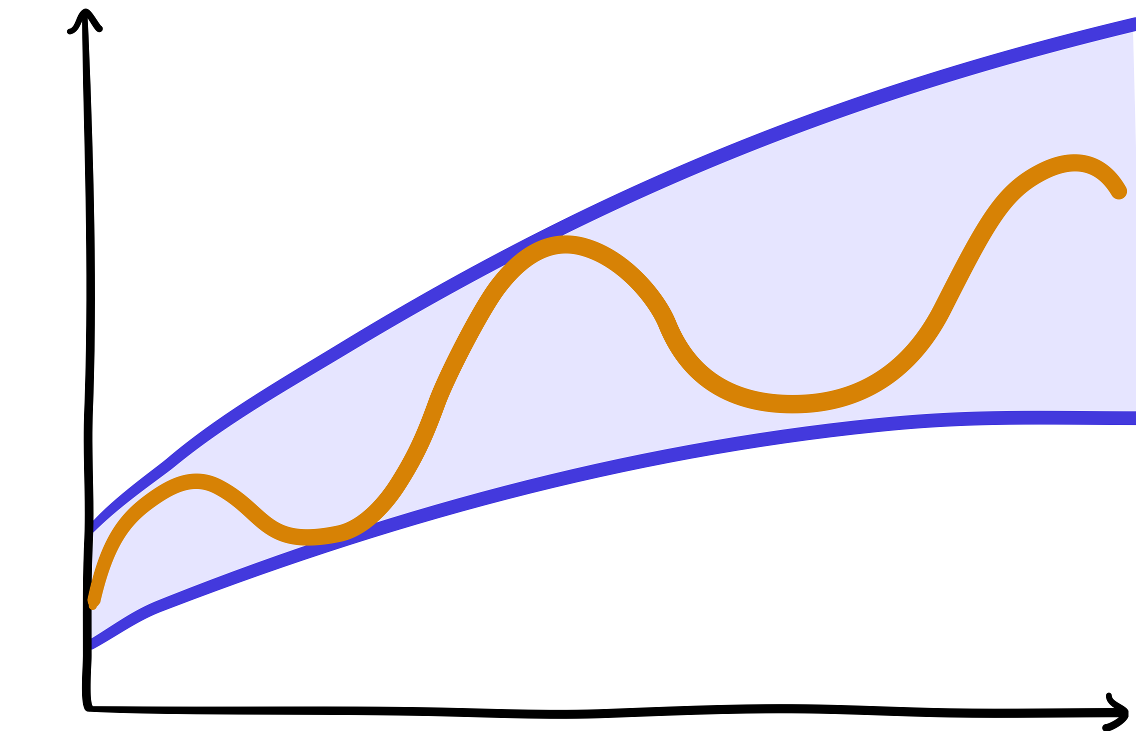


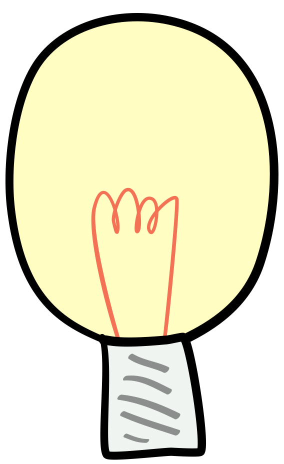
No **big jobs** bothering me

Sufficient Condition

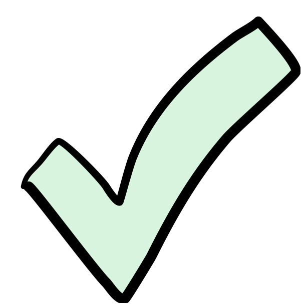
Suppose for some $\delta \geq \gamma > 0$:

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta})$$





Want γ and δ to be close



No **big jobs** bothering me

Main theorem: Consider an M/G/1 queue whose job size distribution X is *intermediate regularly varying* and satisfies

$$\Omega(k^{-\beta}) \leq \frac{\mathbf{P}[X > kx]}{\mathbf{P}[X > x]} \leq O(k^{-\alpha}), \qquad (\beta \geq \alpha > 1, a \to \infty)$$

Main theorem: Consider an M/G/1 queue whose job size distribution X is *intermediate regularly varying* and satisfies

$$\Omega(k^{-\beta}) \leq \frac{\mathbf{P}[X > kx]}{\mathbf{P}[X > x]} \leq O(k^{-\alpha}), \qquad (\beta \geq \alpha > 1, a \to \infty)$$

and suppose a **SOAP** policy with **rank** function r satisfies

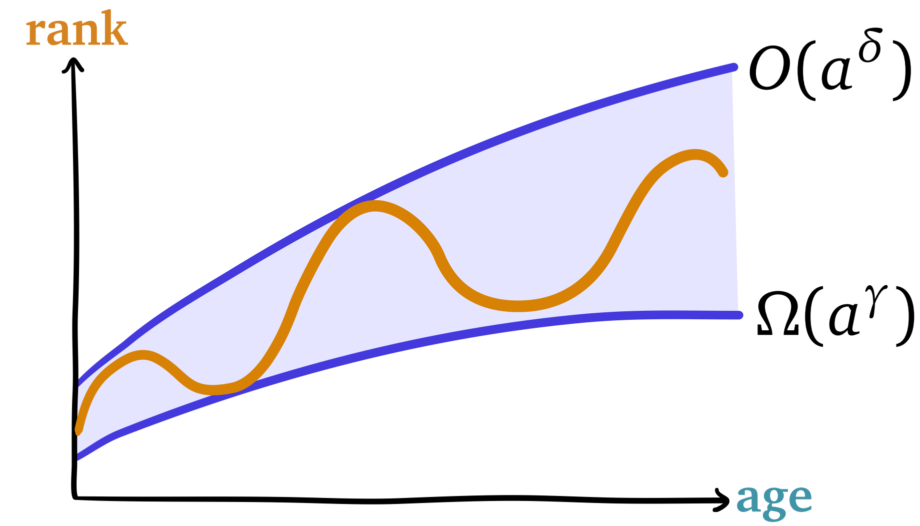
$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta}), \qquad (\delta \geq \gamma > 0, a \to \infty)$$

Main theorem: Consider an M/G/1 queue whose job size distribution X is *intermediate regularly varying* and satisfies

$$\Omega(k^{-\beta}) \leq \frac{\mathbf{P}[X > kx]}{\mathbf{P}[X > x]} \leq O(k^{-\alpha}), \qquad (\beta \geq \alpha > 1, a \to \infty)$$

and suppose a **SOAP** policy with **rank** function r satisfies

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta}), \qquad (\delta \geq \gamma > 0, a \to \infty)$$



Main theorem: Consider an M/G/1 queue whose job size distribution X is *intermediate regularly varying* and satisfies

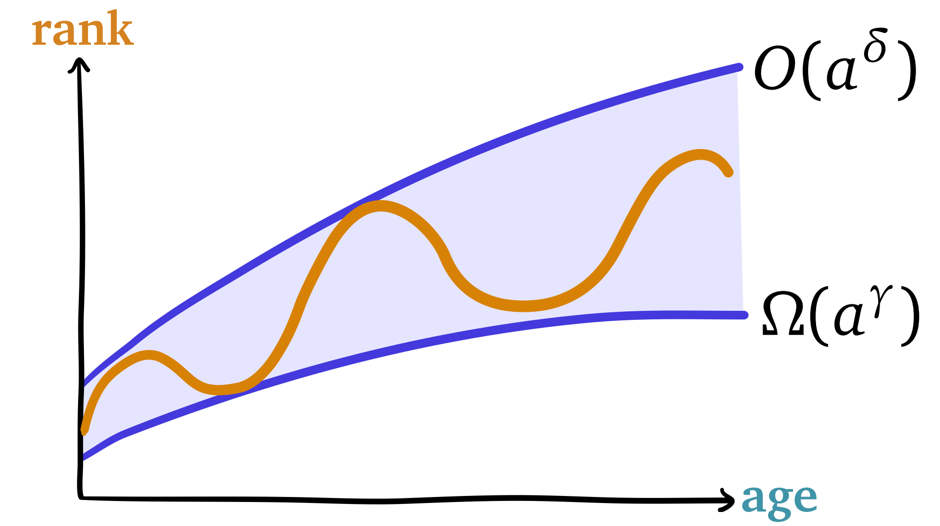
$$\Omega(k^{-\beta}) \leq \frac{\mathbf{P}[X > kx]}{\mathbf{P}[X > x]} \leq O(k^{-\alpha}), \qquad (\beta \geq \alpha > 1, a \to \infty)$$

and suppose a **SOAP** policy with **rank** function r satisfies

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta}), \qquad (\delta \geq \gamma > 0, a \to \infty)$$

Then if

$$\frac{\delta}{\gamma} < \frac{\alpha - 1}{2\beta} + \sqrt{1 + \left(\frac{\alpha - 1}{2\beta}\right)^2},$$



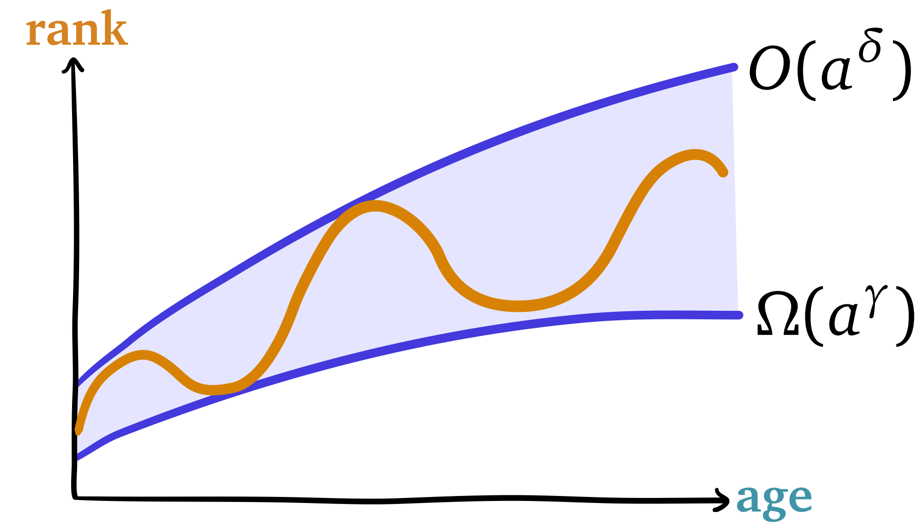
Main theorem: Consider an M/G/1 queue whose job size distribution X is *intermediate regularly varying* and satisfies

$$\Omega(k^{-\beta}) \leq \frac{\mathbf{P}[X > kx]}{\mathbf{P}[X > x]} \leq O(k^{-\alpha}), \qquad (\beta \geq \alpha > 1, a \to \infty)$$

and suppose a **SOAP** policy with **rank** function r satisfies

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta}), \qquad (\delta \geq \gamma > 0, a \to \infty)$$

Then if



$$\frac{\delta}{\gamma} < \frac{\alpha - 1}{2\beta} + \sqrt{1 + \left(\frac{\alpha - 1}{2\beta}\right)^2},$$

the **SOAP** policy is ***tail-optimal*** for X, meaning

$$\mathbf{P}\left[T > \frac{x}{1-\rho}\right] \sim \mathbf{P}[X > x]. \qquad (x \to \infty)$$

Main theorem: Consider an M/G/1 queue whose job size distribution X is *intermediate regularly varying* and satisfies

$$\Omega(k^{-\beta}) \leq \frac{\mathbf{P}[X > kx]}{\mathbf{P}[X > x]} \leq O(k^{-\alpha}), \qquad (\beta \geq \alpha > 1, a \to \infty)$$

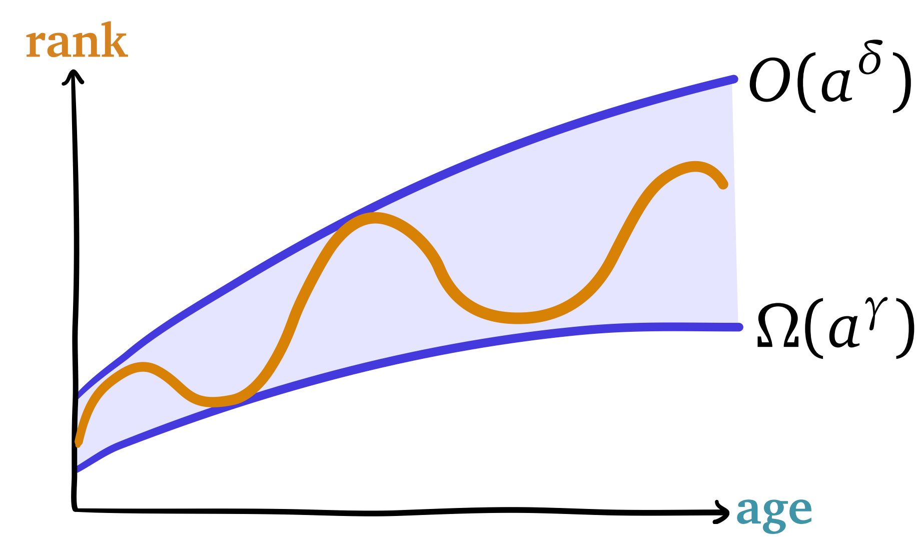
and suppose a **SOAP** policy with **rank** function r satisfies

$$\Omega(a^{\gamma}) \leq r(a) \leq O(a^{\delta}), \qquad (\delta \geq \gamma > 0, a \to \infty)$$

Then if

$\gamma = \delta$ suffices

$$\frac{\delta}{\gamma} < \frac{\alpha - 1}{2\beta} + \sqrt{1 + \left(\frac{\alpha - 1}{2\beta}\right)^2},$$



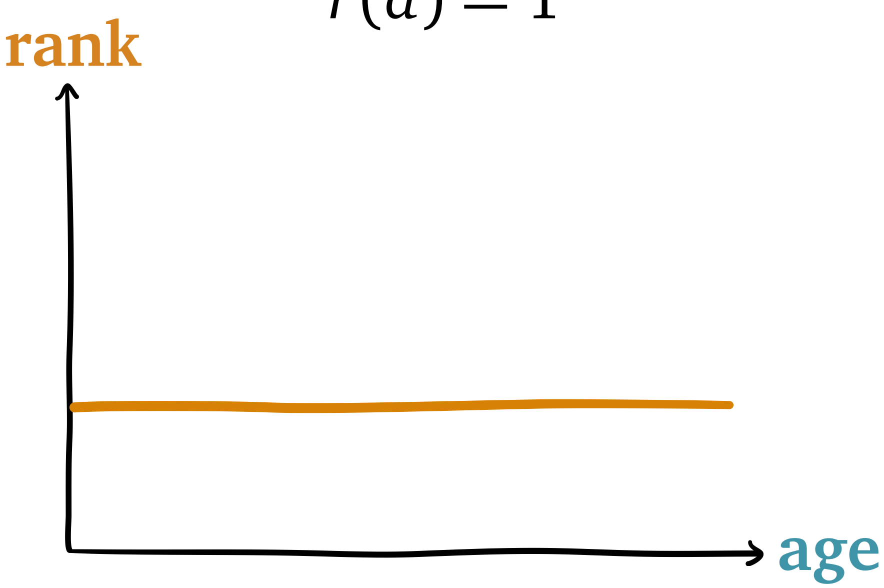
the **SOAP** policy is ***tail-optimal*** for X, meaning

$$\mathbf{P}\left[T > \frac{x}{1-\rho}\right] \sim \mathbf{P}[X > x]. \qquad (x \to \infty)$$

Applying the Condition

FCFS

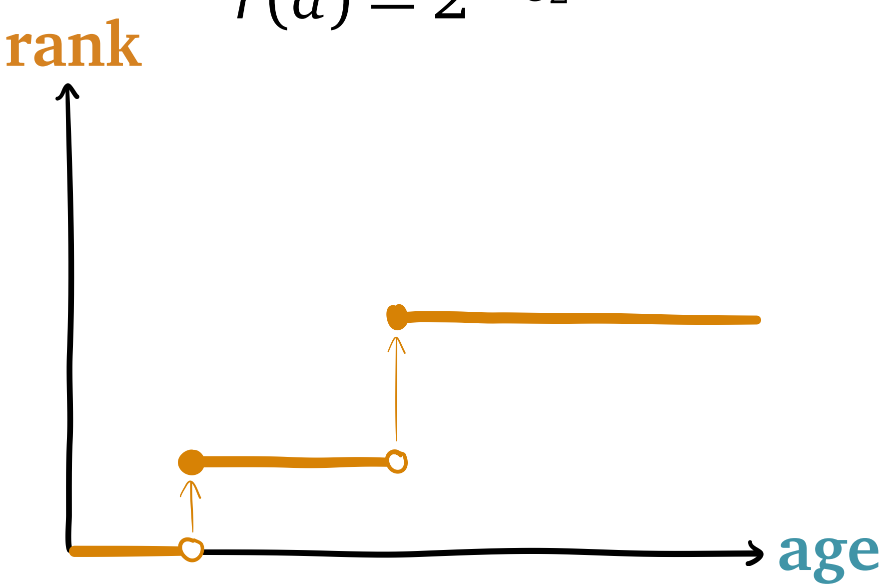
$r(a) = 1$



worst tail

RMLF

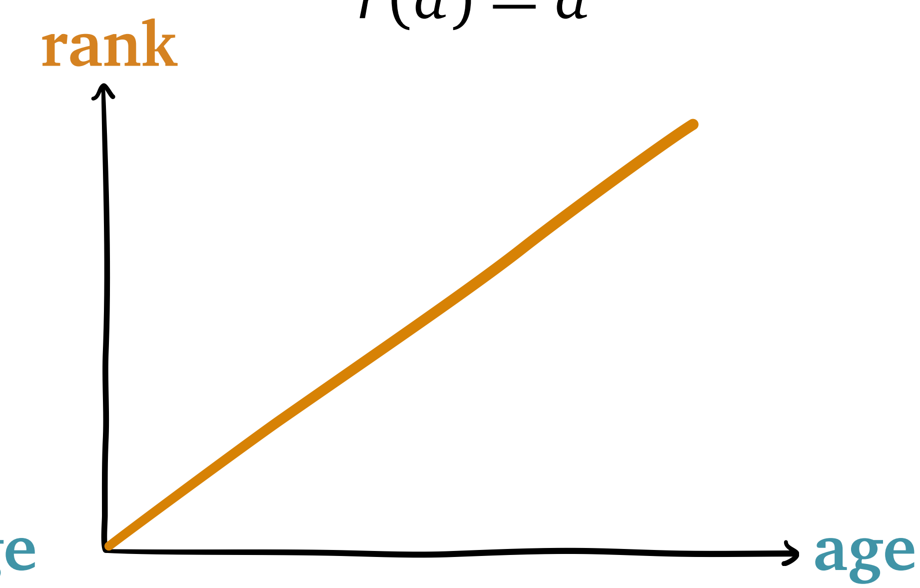
$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$



???

FB

$r(a) = a$

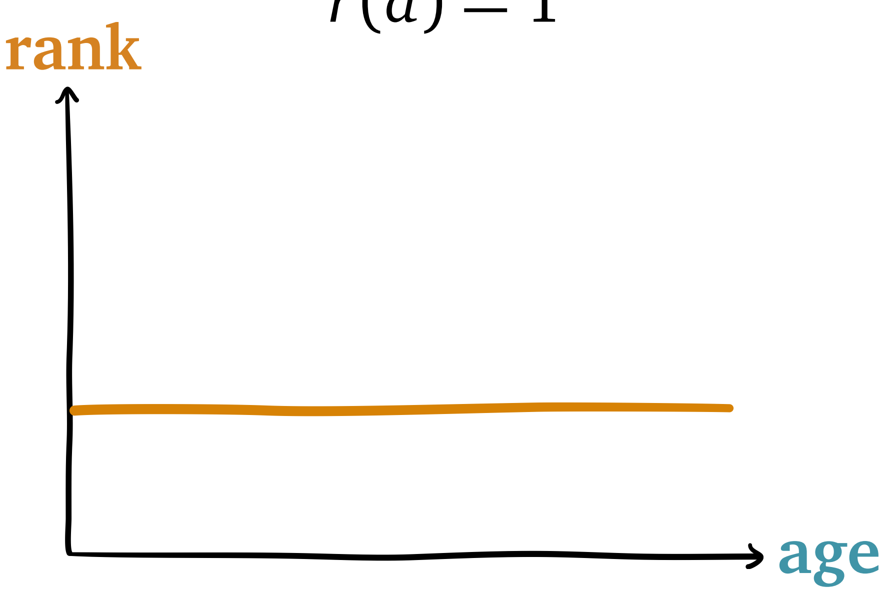


best tail

Applying the Condition

FCFS

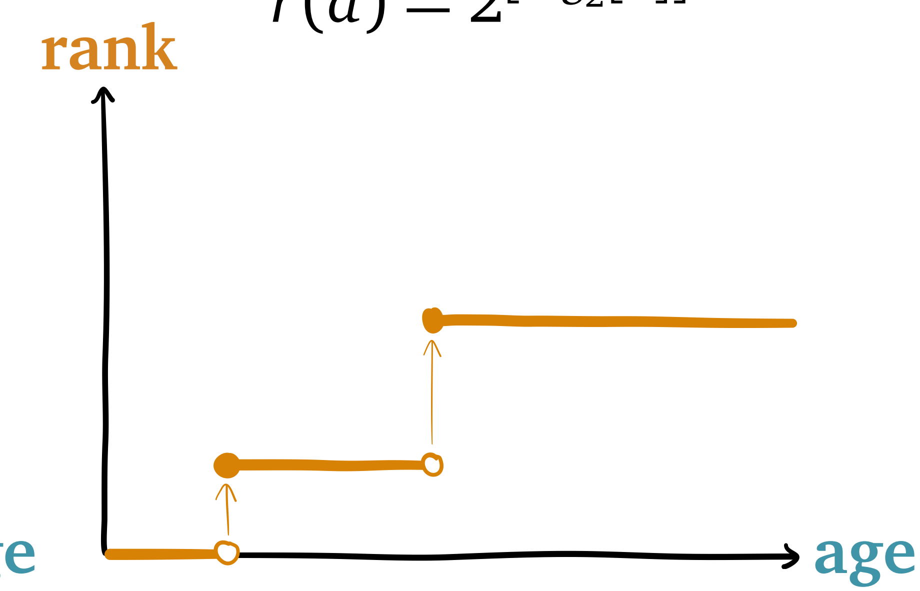
$r(a) = 1$



worst tail

RMLF

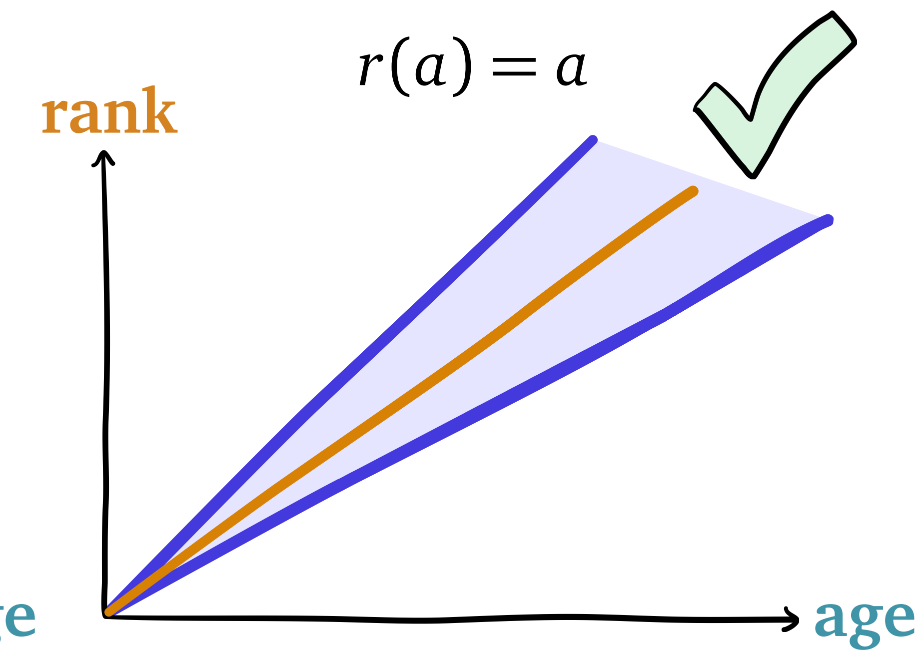
$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$



???

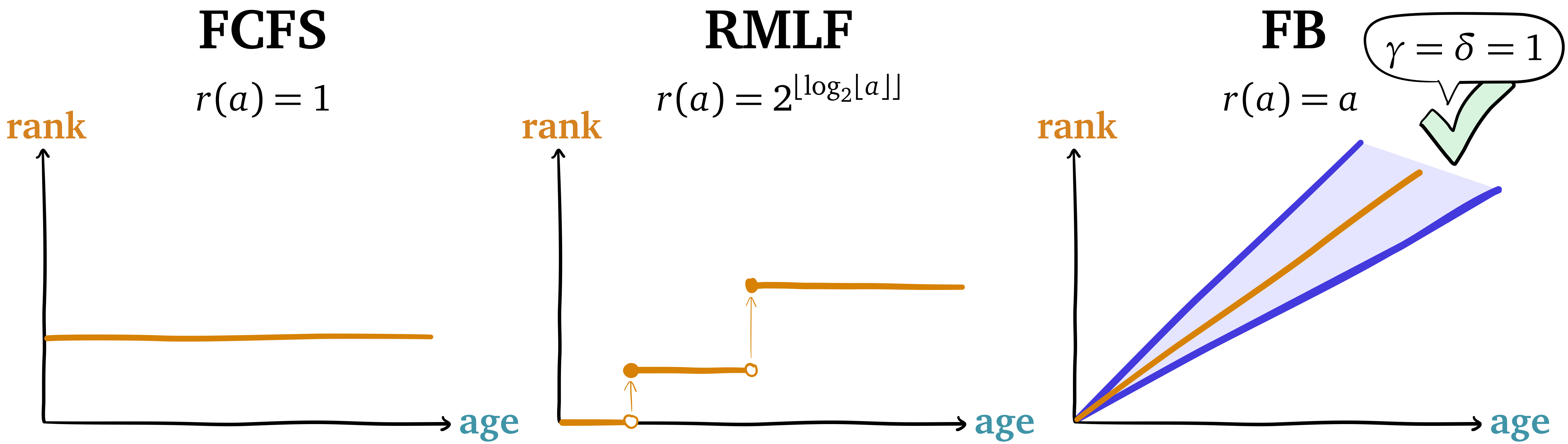
FB

$r(a) = a$



best tail

Applying the Condition

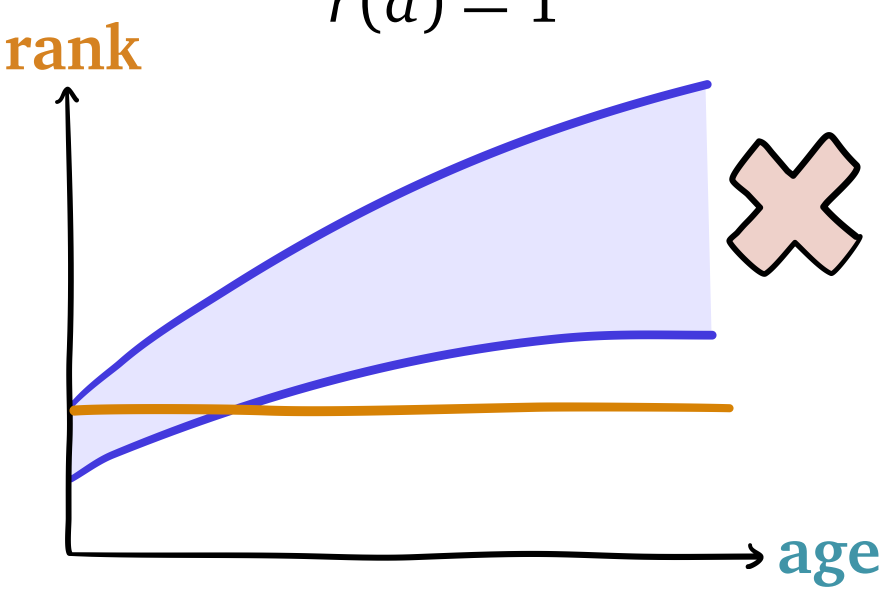


worst tail | ??? | **best** tail

Applying the Condition

FCFS

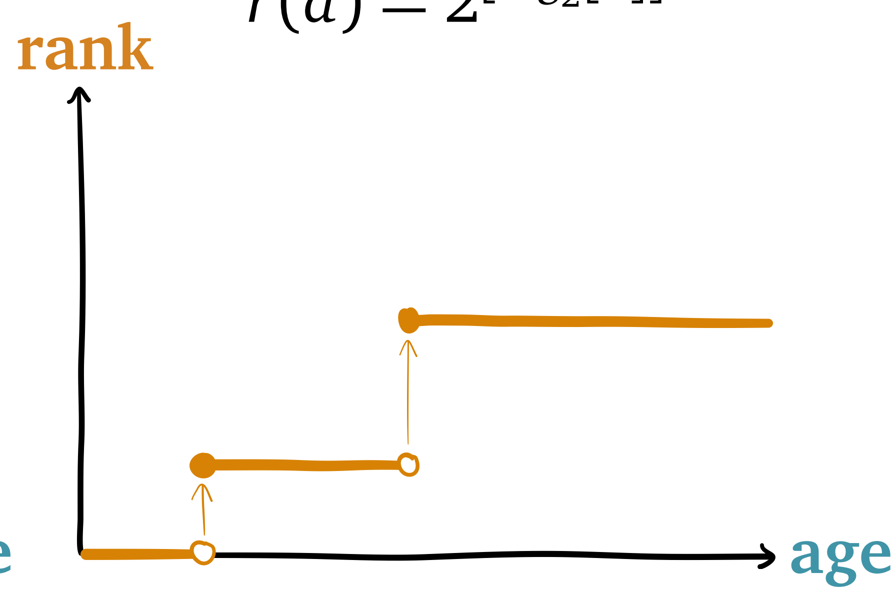
$r(a) = 1$



worst tail

RMLF

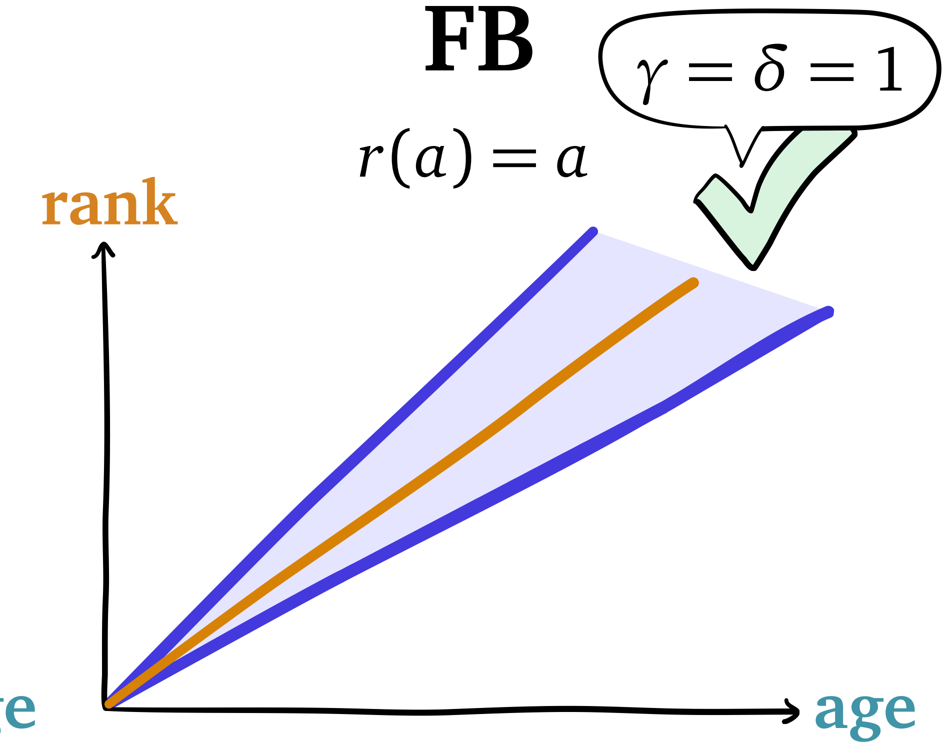
$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$



???

FB

$r(a) = a$

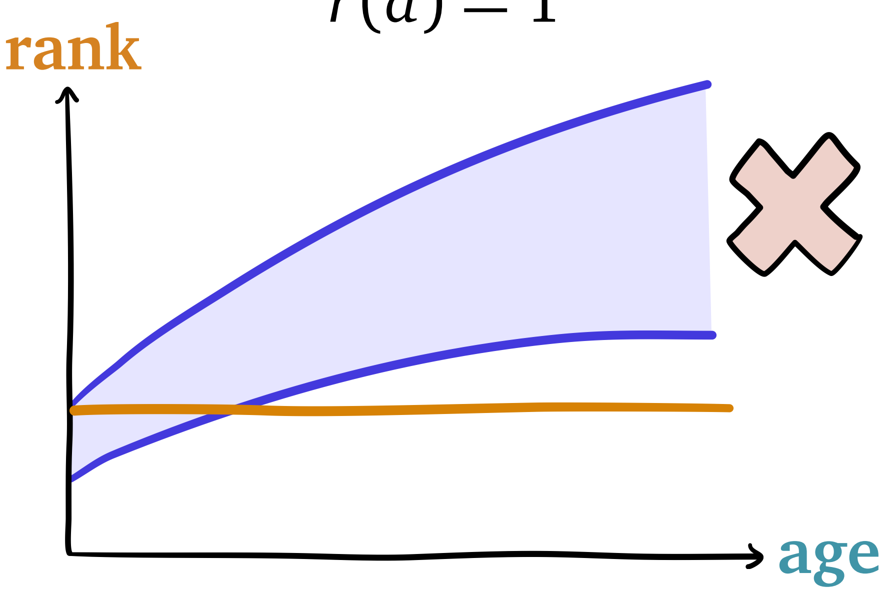


best tail

Applying the Condition

FCFS

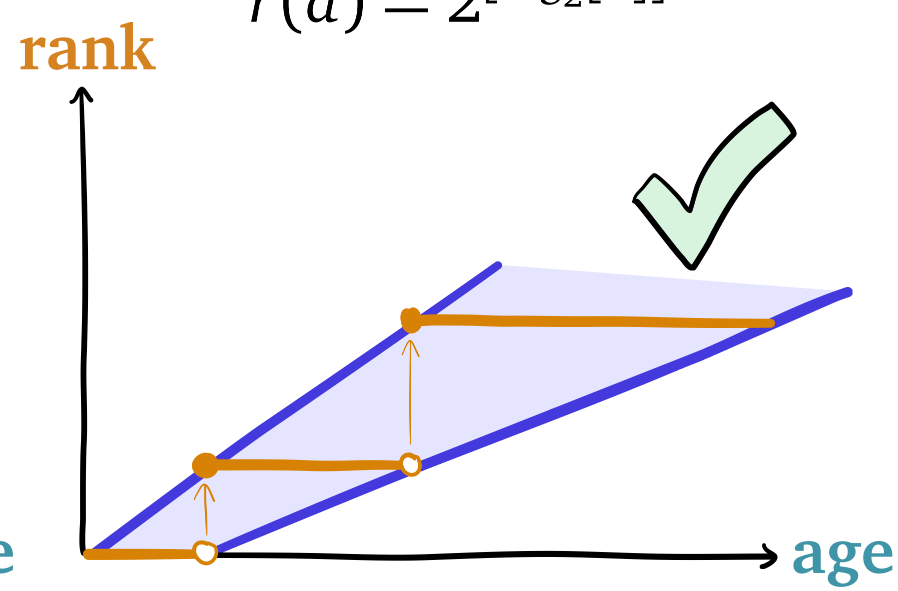
$r(a) = 1$



worst tail

RMLF

$r(a) = 2^{\lfloor \log_2 \lfloor a \rfloor \rfloor}$

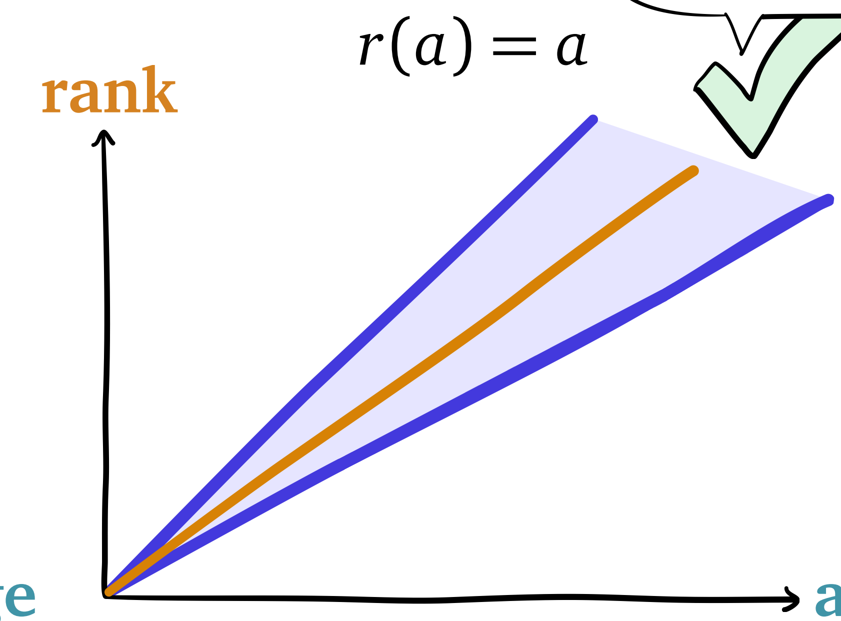


???

FB

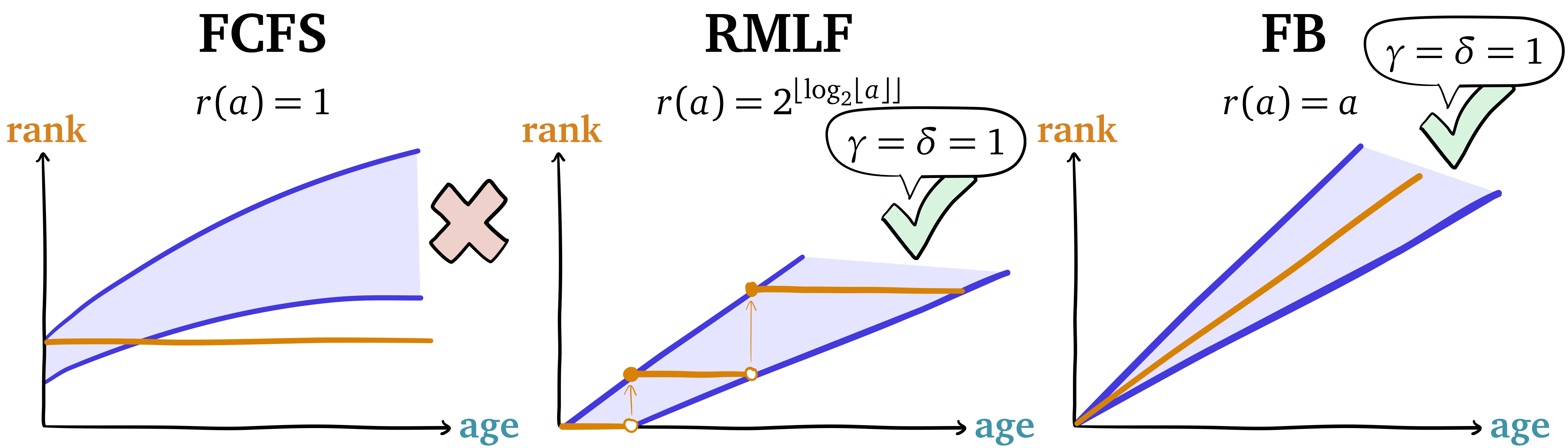
$\gamma = \delta = 1$

$r(a) = a$



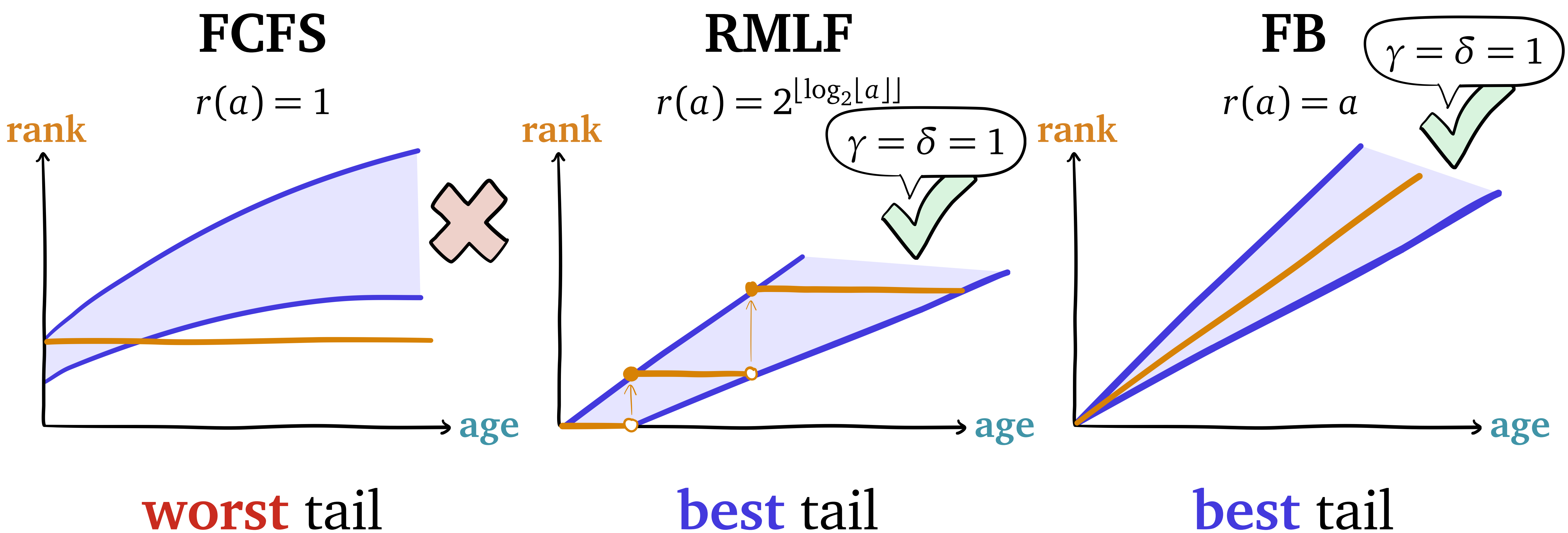
best tail

Applying the Condition



worst tail | **???** | **best** tail

Applying the Condition

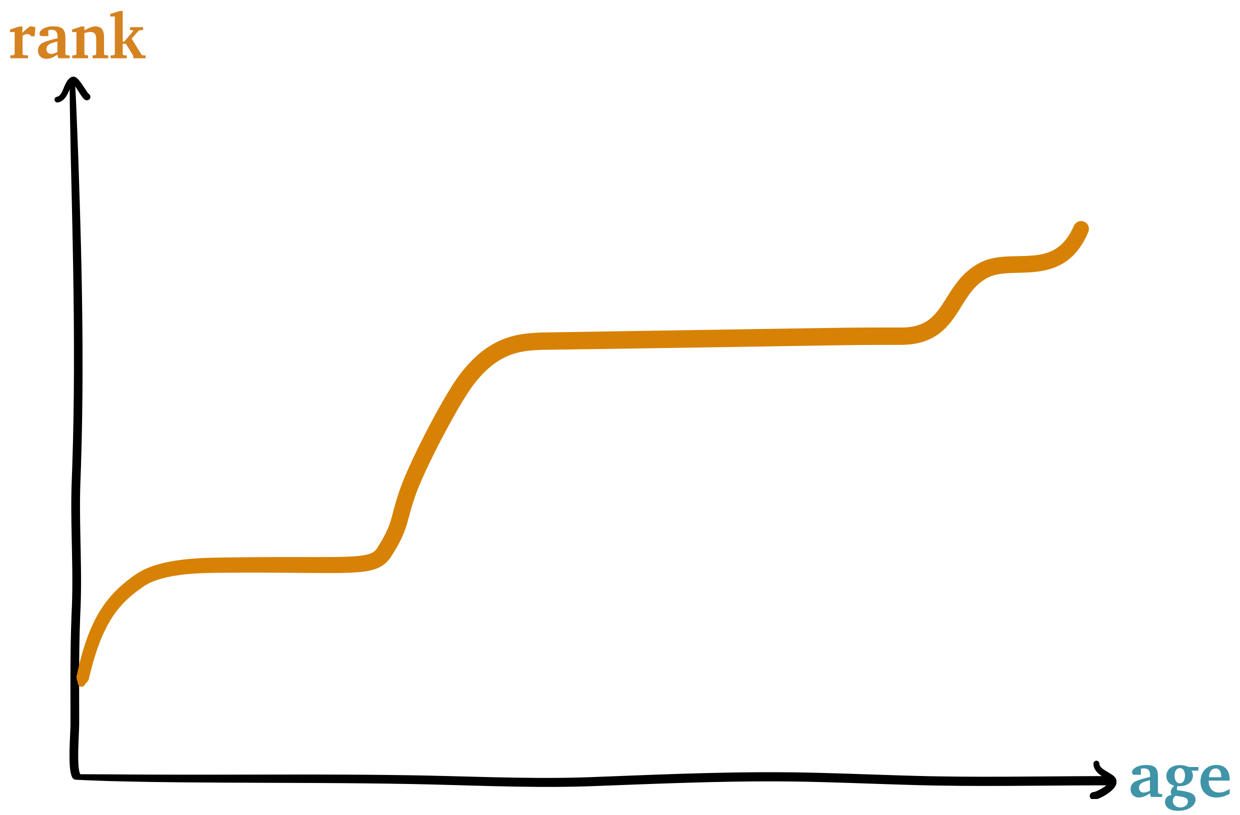


M-SERPT

$$r(a) = \max_{0 \le b \le a} \mathbf{E}[X - b \mid X > b]$$

M-SERPT

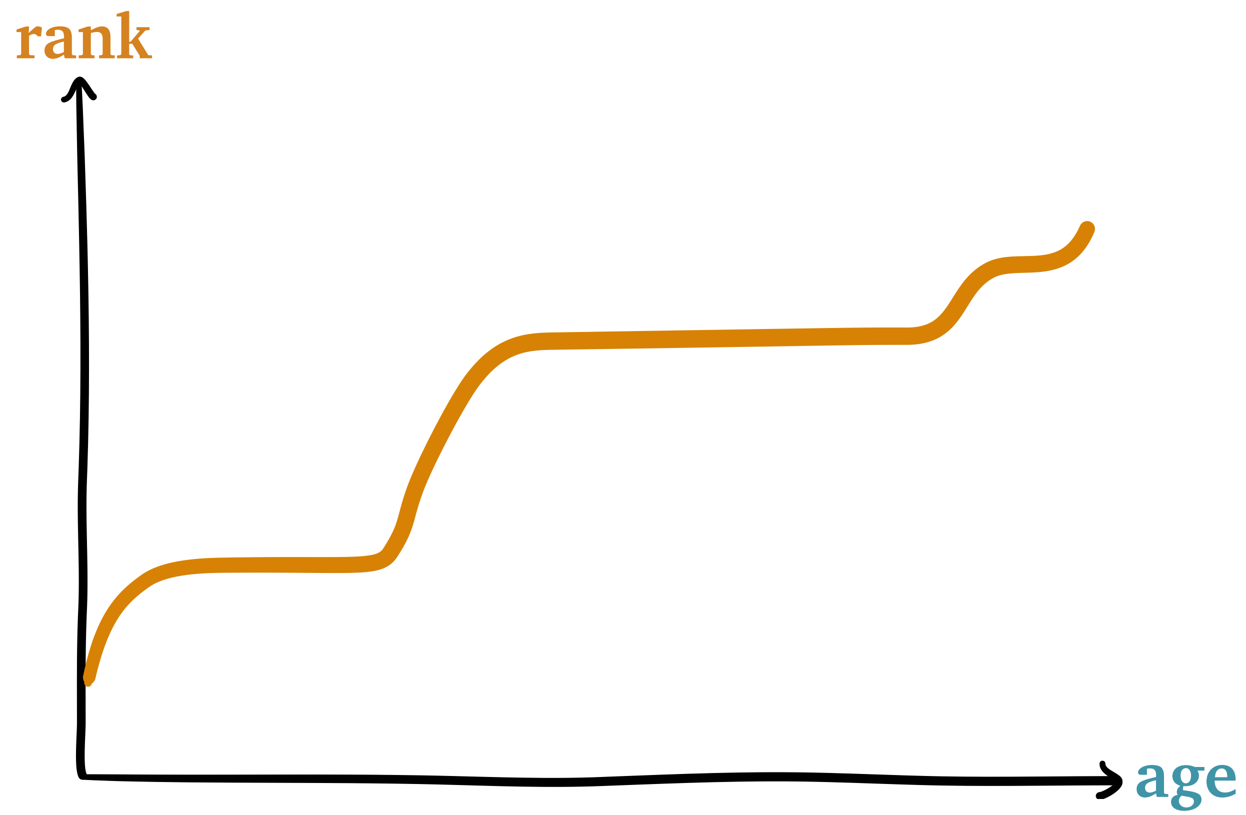
$$r(a) = \max_{0 \le b \le a} \mathbf{E}[X - b \mid X > b]$$



M-SERPT

$$r(a) = \max_{0 \le b \le a} \mathbf{E}[X - b \mid X > b]$$

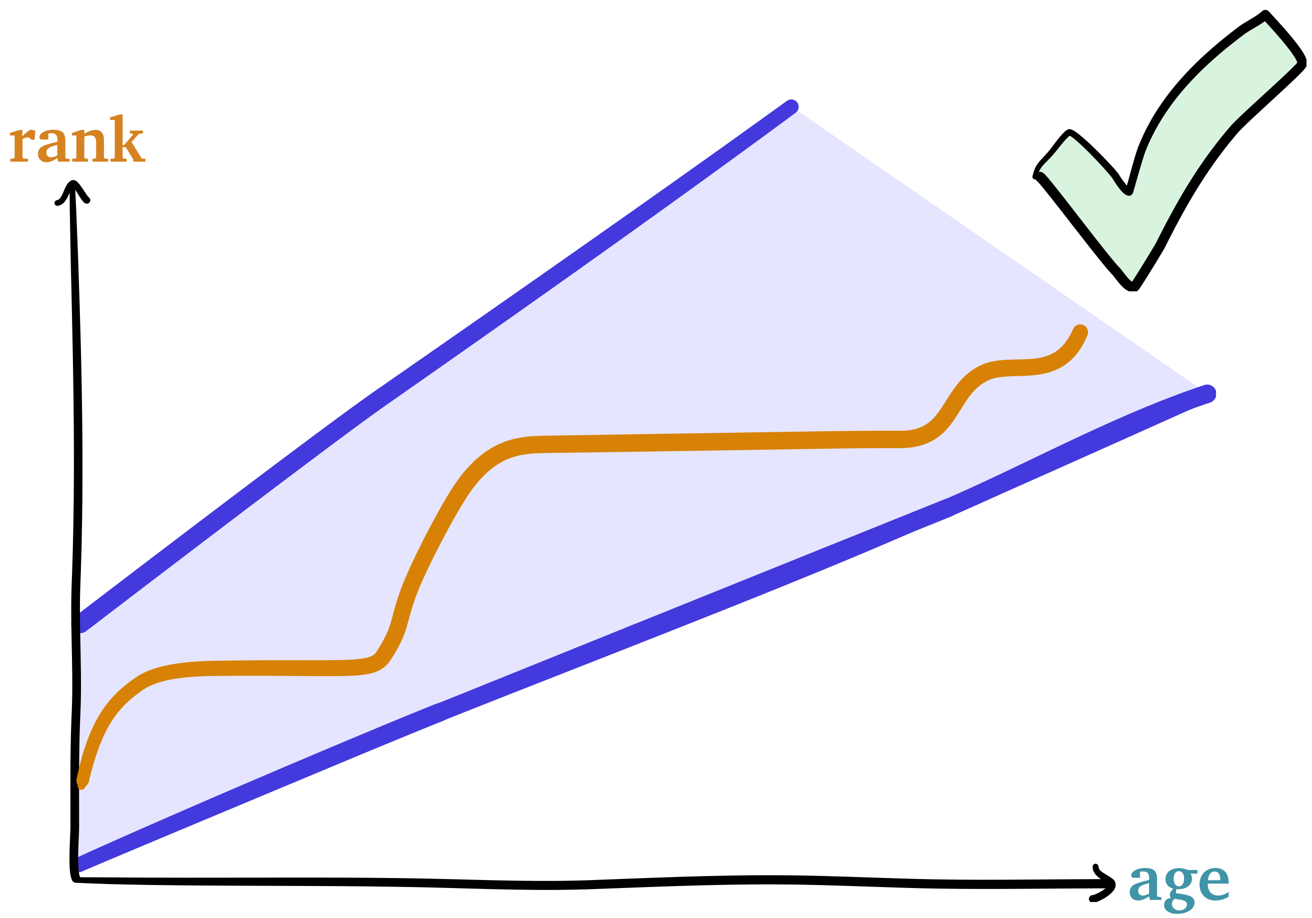
$\Theta(a)$



M-SERPT

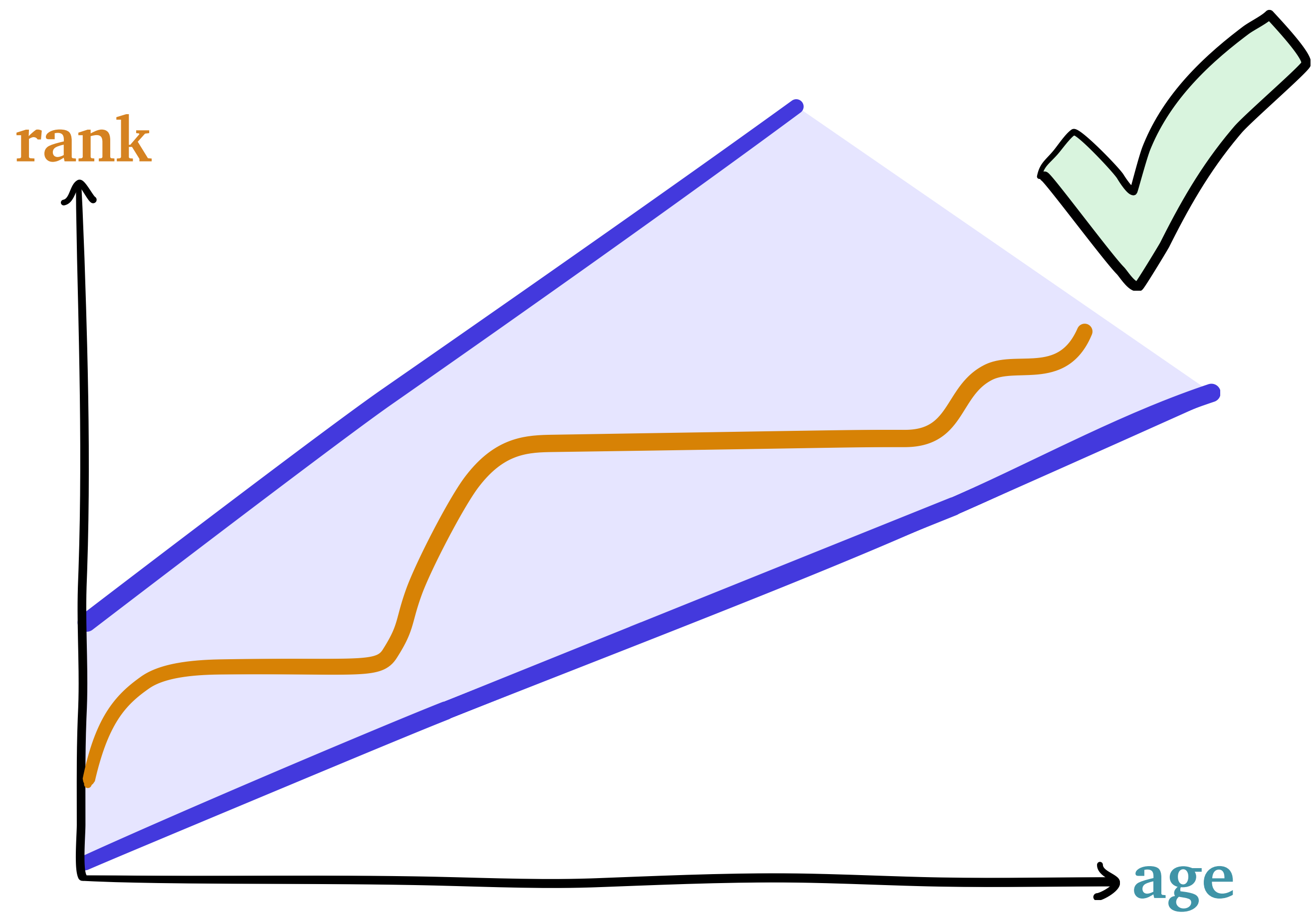
$\Theta(a)$

$$r(a) = \max_{0 \le b \le a} \mathbf{E}[X - b \mid X > b]$$



M-SERPT

$$r(a) = \max_{0 \le b \le a} \mathbf{E}[X - b \mid X > b]$$



$\gamma = \delta = 1 \;\Rightarrow\;$ M-SERPT is **tail-optimal**

Gittins

$$\inf_{b \geq a} \frac{1}{h_X(b)} \leq r_{\text{Gittins}}(a) \leq r_{\text{M-SERPT}}(a)$$

Gittins

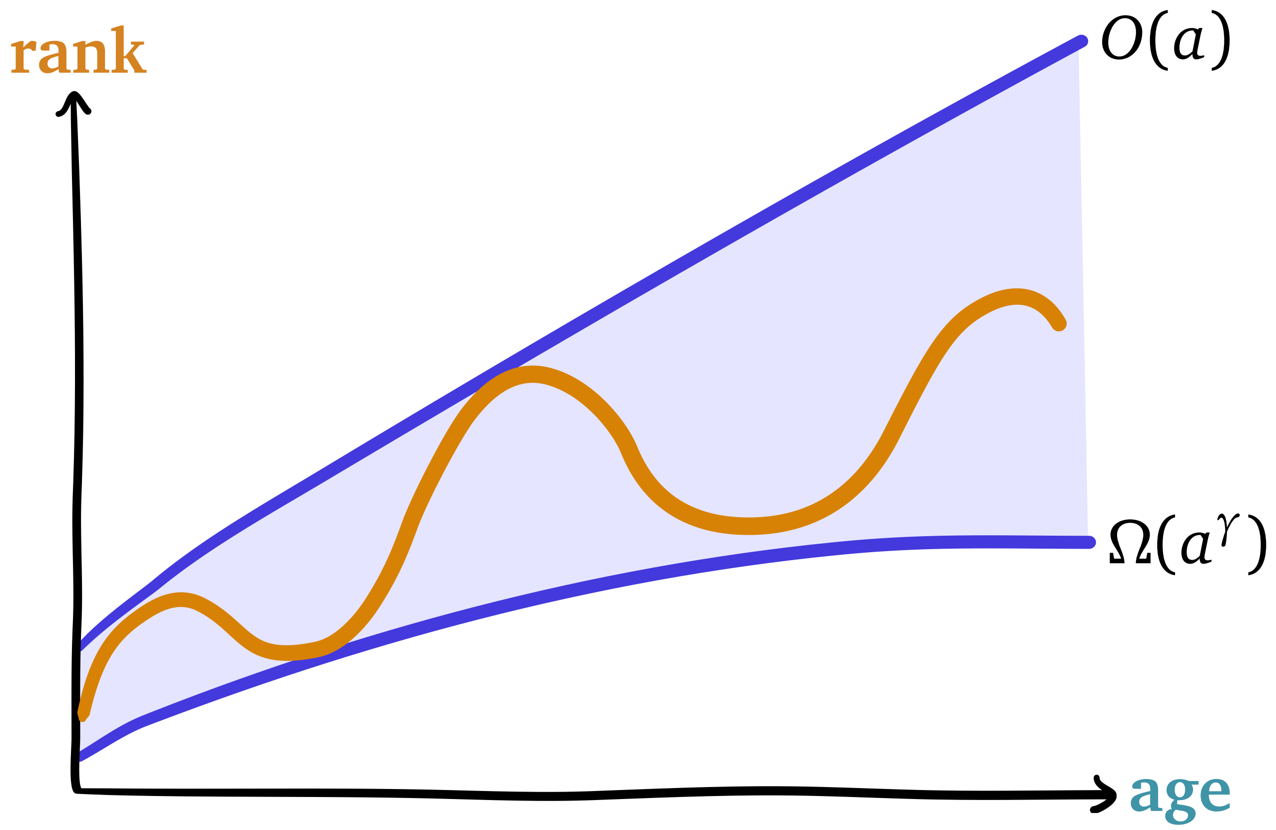
hazard rate of X

$$\inf_{b \geq a} \frac{1}{h_X(b)} \leq r_{\text{Gittins}}(a) \leq r_{\text{M-SERPT}}(a)$$

Gittins

hazard rate of X

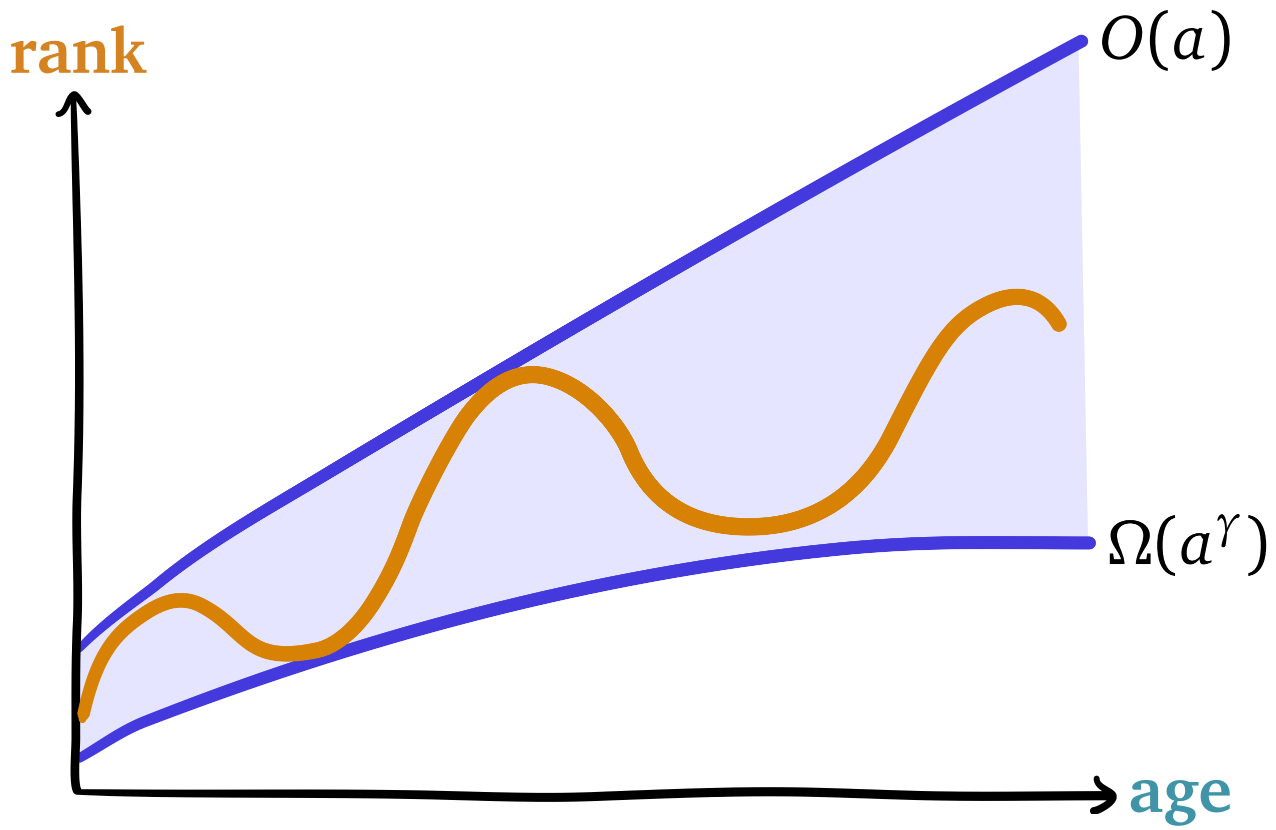
$$\inf_{b \geq a} \frac{1}{h_X(b)} \leq r_{\text{Gittins}}(a) \leq r_{\text{M-SERPT}}(a)$$



Gittins

hazard rate of X

$$\inf_{b \geq a} \frac{1}{h_X(b)} \leq r_{\text{Gittins}}(a) \leq r_{\text{M-SERPT}}(a)$$



Theorem: Gittins is **tail-optimal** if X's hazard rate obeys

$$h_X(a) = O(a^{-\gamma})$$

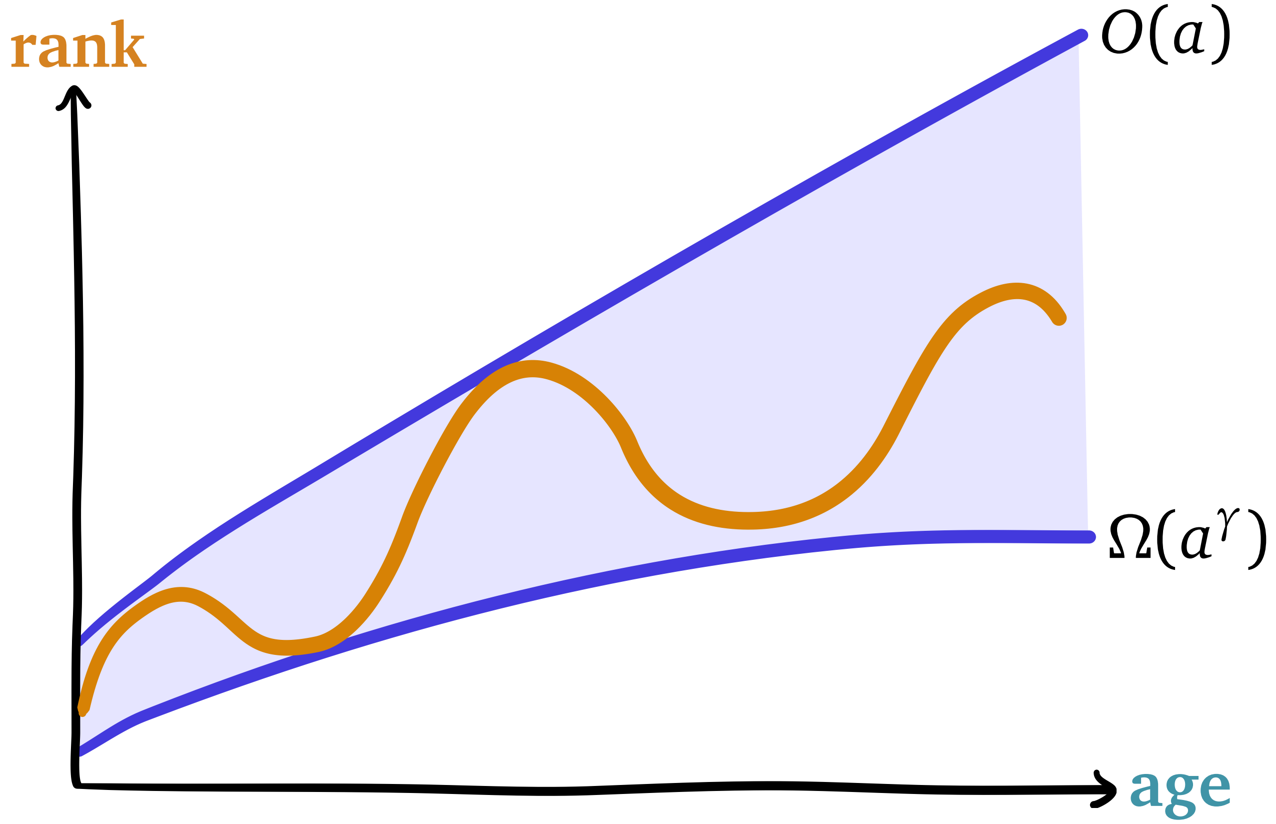
for some

$$\gamma > \sqrt{1 + \left(\frac{\alpha - 1}{2\beta}\right)^2} - \frac{\alpha - 1}{2\beta}.$$

Gittins

hazard rate of X

$$\inf_{b \geq a} \frac{1}{h_X(b)} \leq r_{\text{Gittins}}(a) \leq r_{\text{M-SERPT}}(a)$$



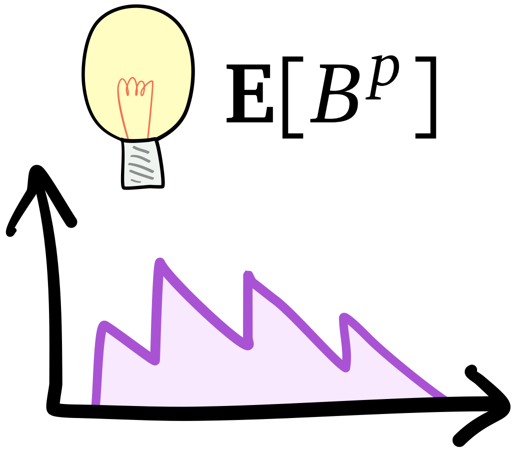
Theorem: Gittins is **tail-optimal** if X's hazard rate obeys

$$h_X(a) = O(a^{-\gamma})$$

for s

$\gamma = 1$ suffices

$$\gamma > \sqrt{1 + \left(\frac{\alpha - 1}{2\beta}\right)^2} - \frac{\alpha - 1}{2\beta}.$$



Part 2:

sketch proof techniques

Proof Outline

Proof Outline

tail-optimal

Proof Outline

$\mathbf{E}[T(x)^p]$ small

Núñez-Queija's
method (2002)

⇓

tail-optimal

Proof Outline

response time
of job of size x

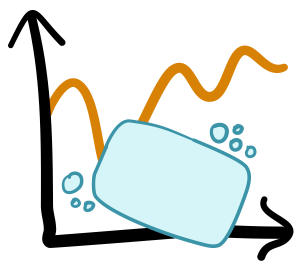
$\mathbf{E}[T(x)^p]$ small

Núñez-Queija's
method (2002)

tail-optimal

Proof Outline

$\mathbf{E}[B^p]$ small



SOAP
(2018)

$\mathbf{E}[T(x)^p]$ small

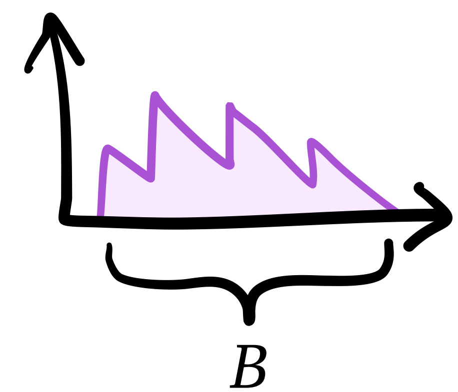
Núñez-Queija's
method (2002)

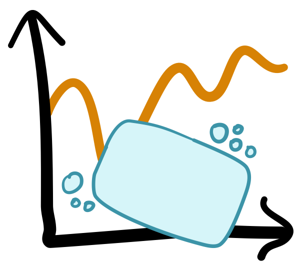
tail-optimal

Proof Outline

$\mathbf{E}[B^p]$ small

M/G/1 **busy period**





SOAP
(2018)

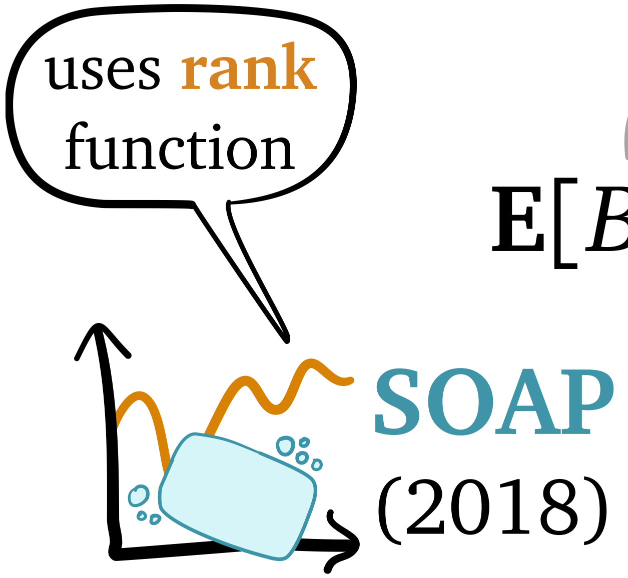
$\mathbf{E}[T(x)^p]$ small

Núñez-Queija's
method (2002)

tail-optimal

Proof Outline

$\mathbf{E}[B^p]$ small

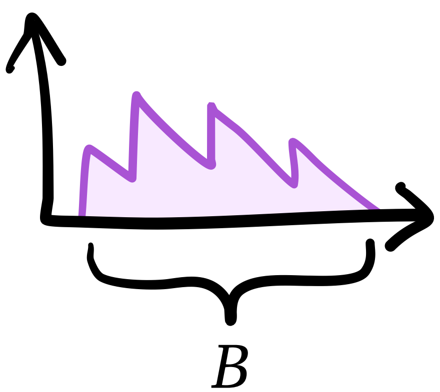


$\mathbf{E}[T(x)^p]$ small

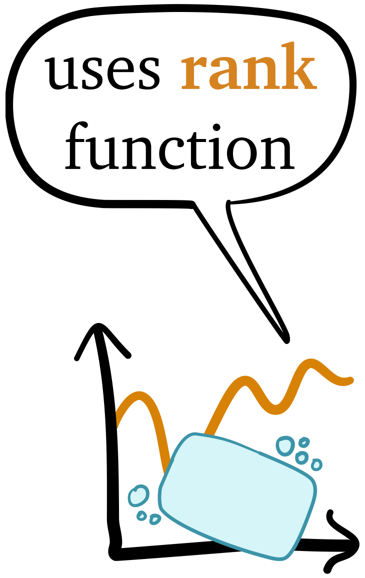
Núñez-Queija's method (2002)

tail-optimal

M/G/1 **busy period**



Proof Outline



$\mathbf{E}[B^p]$ small

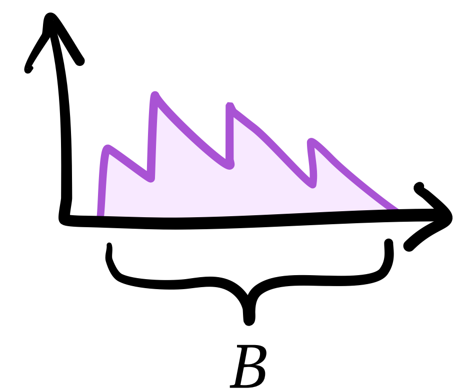
SOAP (2018)

$\mathbf{E}[T(x)^p]$ small

Núñez-Queija's method (2002)

tail-optimal

M/G/1 **busy period**

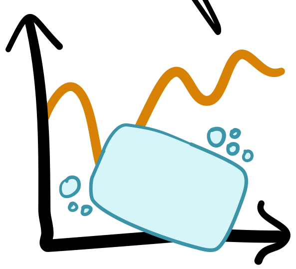


new bound on *fractional moments*

Proof Outline

uses **rank** function

$\mathbf{E}[B^p]$ small



SOAP
(2018)

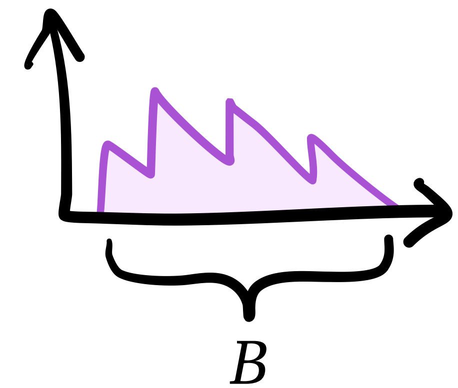
new version

$\mathbf{E}[T(x)^p]$ small

Núñez-Queija's
method (2002)

tail-optimal

M/G/1 **busy period**

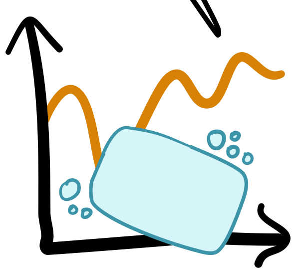


new bound on
fractional moments

Proof Outline

uses **rank** function

$\mathbf{E}[B^p]$ small



SOAP (2018)

new version

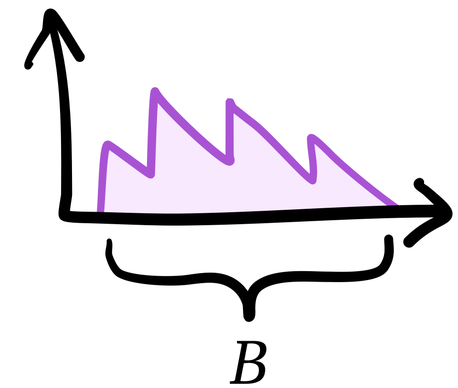
$\mathbf{E}[T(x)^p]$ small

Núñez-Queija's method (2002)

new version

tail-optimal

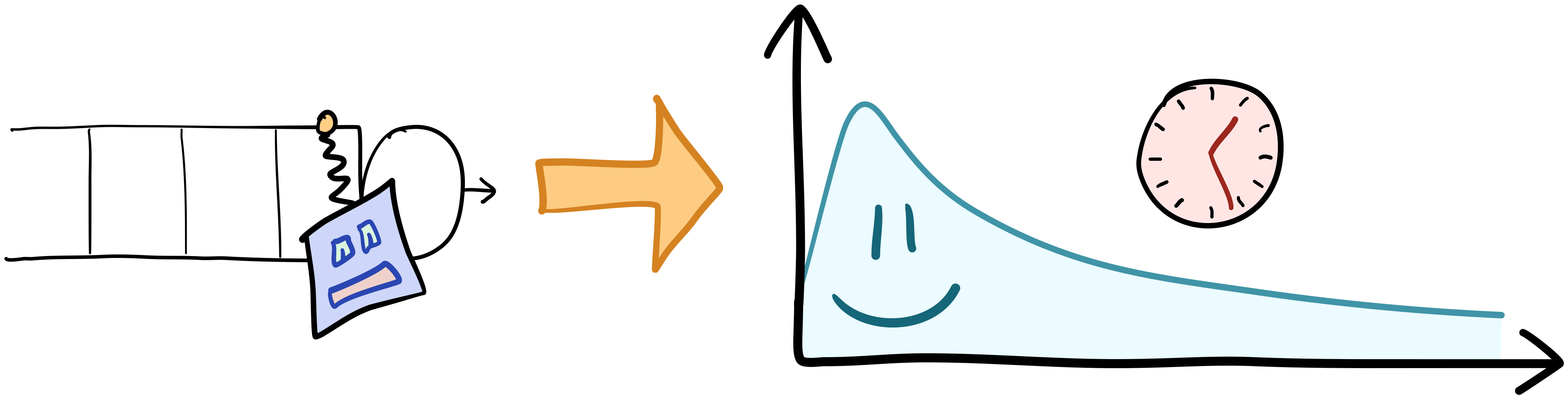
M/G/1 **busy period**



new bound on *fractional moments*

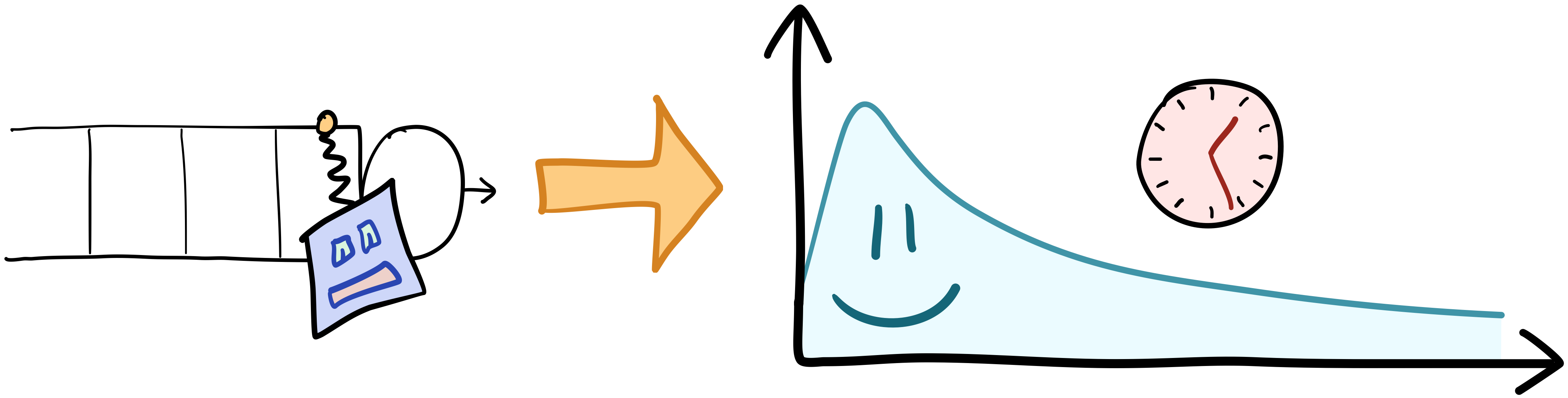
Summary

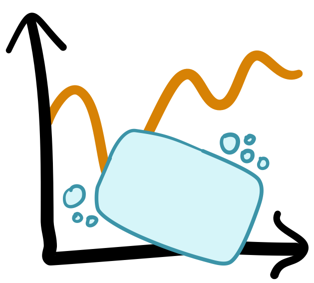
Result: sufficient condition for **tail-optimality**



Summary

Result: sufficient condition for **tail-optimality**

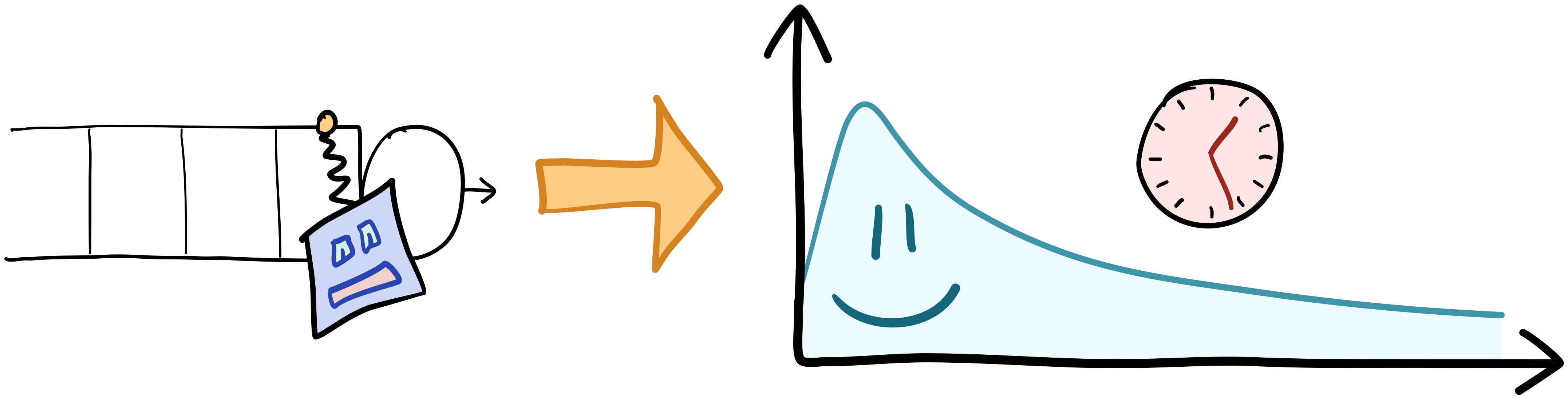


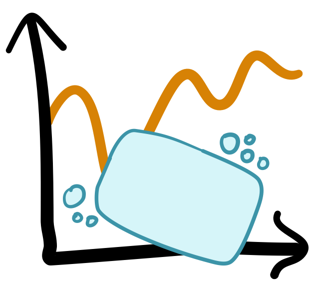


Key idea #1: condition stated using ***rank*** *function* of **SOAP** policy

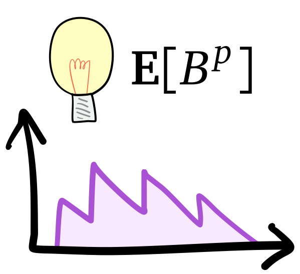
Summary

Result: sufficient condition for **tail-optimality**





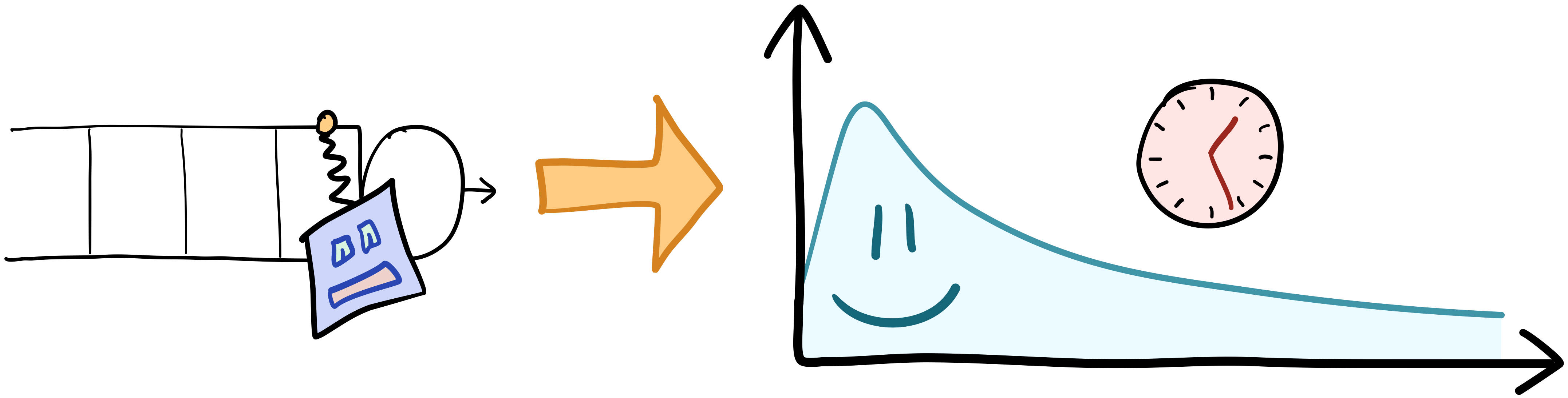
Key idea #1: condition stated using ***rank*** *function* of **SOAP** policy

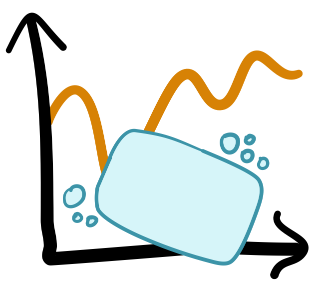


Key idea #2: new bound on *fractional moments* of M/G/1 **busy periods**

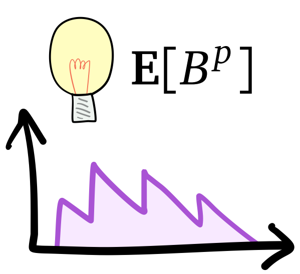
Summary

Result: sufficient condition for **tail-optimality**





Key idea #1: condition stated using ***rank*** *function* of **SOAP** policy



Key idea #2: new bound on *fractional moments* of M/G/1 **busy periods**

Get in touch: `zscully@cs.cmu.edu`

Bonus Slides

Prior Sufficient Conditions

Prior Sufficient Conditions

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

Prior Sufficient Conditions

response time
of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

Prior Sufficient Conditions

response time
of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

- Hard to verify!

Prior Sufficient Conditions

response time
of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

- Hard to verify!

NWZ (2008): *SMART* policies are **tail-optimal**

Prior Sufficient Conditions

response time
of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

- Hard to verify!

SRPT and
its friends

NWZ (2008): *SMART* policies are **tail-optimal**

Prior Sufficient Conditions

response time of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

- Hard to verify!

SRPT and its friends

NWZ (2008): *SMART* policies are **tail-optimal**

- Easy to verify…

Prior Sufficient Conditions

response time
of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

- Hard to verify!

SRPT and
its friends

NWZ (2008): *SMART* policies are **tail-optimal**

- Easy to verify…
- … but only applies with *known* job sizes

Prior Sufficient Conditions

response time
of job of size x

Núñez-Queija (2002): policy is **tail-optimal** if moments of $T(x)$ are small

- Hard to verify!

SRPT and
its friends

NWZ (2008): *SMART* policies are **tail-optimal**

- Easy to verify…
- … but only applies with *known* job sizes

Wanted:

easy-to-verify condition for systems with *unknown* job sizes