A queueing perspective on randomized work sharing vs work stealing

B. Van Houdt

December 4, 2018
Outline

1. Setting

2. Traditional strategies

3. Rate-based Strategies

4. Global attraction

5. Non-exponential job sizes
Outline

1 Setting

2 Traditional strategies

3 Rate-based Strategies

4 Global attraction

5 Non-exponential job sizes
Setting

- Set of $N$ servers, each subject to local Poisson arrivals (rate $\lambda$)
Set of $N$ servers, each subject to local Poisson arrivals (rate $\lambda$)

- Inefficient: as servers may be idle while others have pending jobs
Setting

- Set of $N$ servers, each subject to local Poisson arrivals (rate $\lambda$)
- Inefficient: as servers may be idle while others have pending jobs
- Redistribute the work/jobs
Setting

- Set of $N$ servers, each subject to local Poisson arrivals (rate $\lambda$)
- Inefficient: as servers may be idle while others have pending jobs
- Redistribute the work/jobs

Strategies:

1. Work stealing (pull): lightly-loaded servers attempt to steal work
2. Work sharing (push): heavily-loaded servers attempt to share work
Setting

- Set of \( N \) servers, each subject to local Poisson arrivals (rate \( \lambda \))
- Inefficient: as servers may be idle while others have pending jobs
- Redistribute the work/jobs

Strategies:

1. Work stealing (pull): idle servers attempt to steal work
2. Work sharing (push): servers with pending jobs attempt to share work
Setting

- Set of $N$ servers, each subject to local Poisson arrivals (rate $\lambda$)
- Inefficient: as servers may be idle while others have pending jobs
- Redistribute the work/jobs

Strategies:

1. Work stealing (pull): idle servers attempt to steal work
2. Work sharing (push): servers with pending jobs attempt to share work

Stealing is clearly best under very high loads, sharing under very low loads
1. Setting

2. Traditional strategies

3. Rate-based Strategies

4. Global attraction

5. Non-exponential job sizes
Randomized Strategies [Eager, Lazowska & Zahorjan 1984]:

1. Work stealing: Whenever a server becomes idle, it probes up to $L_p$ servers at random to steal a job.
Randomized Strategies [Eager, Lazowska & Zahorjan 1984]:

1. Work stealing: Whenever a server becomes idle, it probes up to $L_p$ servers at random to steal a job.

2. Work sharing: Whenever a job arrives in a busy server, it probes up to $L_p$ servers at random to transfer the incoming job.
An arriving job probes up to $L_p$ servers at random for idle server $s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$
An arriving job probes up to $L_p$ servers at random for idle server $s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$

Set of ODEs:

\[
\begin{align*}
\frac{ds_1(t)}{dt} &= \lambda(1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^{L_p}) \quad - (s_1(t) - s_2(t)) \\
\frac{ds_i(t)}{dt} &= \lambda(s_{i-1}(t) - s_i(t))s_1(t)^{L_p} \quad - (s_i(t) - s_{i+1}(t)) 
\end{align*}
\]

for $i \geq 2$. 

Unique fixed point:

\[
\pi_{i+1} = \lambda^{-1}(1 + (L_p + 1)i)
\]

for $i \geq 0$. 

Randomized work stealing/sharing 

December 4, 2018 8 / 38
An arriving job probes up to $L_p$ servers at random for idle server $s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$

Set of ODEs:

\[
\frac{ds_1(t)}{dt} = \lambda (1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^{L_p}) - (s_1(t) - s_2(t))
\]

\[
\frac{ds_i(t)}{dt} = \lambda (s_{i-1}(t) - s_i(t)) s_1(t)^{L_p} - (s_i(t) - s_{i+1}(t))
\]

for $i \geq 2$.

Unique fixed point: $\pi_{i+1} = \lambda^{1+(L_p+1)i}$ for $i \geq 0$. 
Work stealing: mean field model for expo job sizes

- Server that becomes idle probes up to $L_p$ servers at random
- $s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$
Server that becomes idle probes up to $L_p$ servers at random

$s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$

Set of ODEs:

$$\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) - \frac{(s_1(t) - s_2(t))(1 - s_2(t))^{L_p}}{s_2(t)}$$

$$\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t)) - (s_i(t) - s_{i+1}(t))$$

$$- \frac{(s_i(t) - s_{i+1}(t))}{s_2(t)}(s_1(t) - s_2(t))(1 - (1 - s_2(t))^{L_p})$$

for $i \geq 2$, where $\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))$ if $s_2(t) = 0$ and $i \geq 2$. 
Work stealing: mean field model for expo job sizes

- Server that becomes idle probes up to \( L_p \) servers at random
- \( s_i(t) \): fraction of queues containing at least \( i \) jobs at time \( t \)
- Set of ODEs:

\[
\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) - (s_1(t) - s_2(t))(1 - s_2(t))^{L_p}
\]

\[
\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t)) - (s_i(t) - s_{i+1}(t)) - \frac{(s_i(t) - s_{i+1}(t))}{s_2(t)}(s_1(t) - s_2(t))(1 - (1 - s_2(t))^{L_p})
\]

for \( i \geq 2 \), where \( \frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t)) \) if \( s_2(t) = 0 \) and \( i \geq 2 \).

- Unique fixed point: \( \pi_2 \) root of

\[
g(x) = \lambda(1 - \lambda) - (\lambda - x)(1 - x)^{L_p} = 0,
\]
Work stealing versus sharing (aka pull versus push)

Let’s compare, right?
Work stealing versus sharing (aka pull versus push)

Let’s compare, right? NO!

⇒ Communication overhead depends on the load and is not the same
Outline

1. Setting
2. Traditional strategies
3. Rate-based Strategies
4. Global attraction
5. Non-exponential job sizes
Rate-based strategies

Randomized Strategies [Minnebo, VH 2014]:

1. Work stealing: Whenever a server is idle, it randomly probes at rate $r_{\text{steal}}$ to steal a job.
Randomized Strategies [Minnebo, VH 2014]:

1. **Work stealing**: Whenever a server is idle, it randomly probes at rate \( r_{steal} \) to steal a job.

2. **Work sharing**: Whenever a server has pending jobs, it randomly probes at rate \( r_{share} \) to transfer a pending job.

Aren’t traditional strategies better?

**NO!**

Randomized work stealing/sharing
Rate-based strategies

Randomized Strategies [Minnebo, VH 2014]:

1. Work stealing: Whenever a server is idle, it randomly probes at rate $r_{steal}$ to steal a job

2. Work sharing: Whenever a server has pending jobs, it randomly probes at rate $r_{share}$ to transfer a pending job

Aren’t traditional strategies better?
Rate-based strategies

Randomized Strategies [Minnebo, VH 2014]:

1. Work stealing: Whenever a server is idle, it randomly probes at rate $r_{steal}$ to steal a job.

2. Work sharing: Whenever a server has pending jobs, it randomly probes at rate $r_{share}$ to transfer a pending job.

Aren’t traditional strategies better? NO!
Rate-based: mean field model for expo job sizes

- Single mean field model for stealing/sharing
- $s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$
Rate-based: mean field model for expo job sizes

- Single mean field model for stealing/sharing
- $s_i(t)$: fraction of queues containing at least $i$ jobs at time $t$
- Set of ODEs:

$$\frac{d}{dt}s_1(t) = \lambda(1 - s_1(t)) - (s_1(t) - s_2(t)) + r(1 - s_1(t))s_2(t),$$

$$\frac{d}{dt}s_i(t) = \lambda(s_{i-1}(t) - s_i(t)) - (s_i(t) - s_{i+1}(t))$$

$$- r(1 - s_1(t))(s_i(t) - s_{i+1}(t)),$$

for $i \geq 2$. 
Rate-based: mean field model for expo job sizes

- Single mean field model for stealing/sharing
- \( s_i(t) \): fraction of queues containing at least \( i \) jobs at time \( t \)
- Set of ODEs:

\[
\frac{d}{dt} s_1(t) = \lambda (1 - s_1(t)) - (s_1(t) - s_2(t)) + r(1 - s_1(t))s_2(t),
\]

\[
\frac{d}{dt} s_i(t) = \lambda (s_{i-1}(t) - s_i(t)) - (s_i(t) - s_{i+1}(t))
\]

\[\quad - r(1 - s_1(t))(s_i(t) - s_{i+1}(t)),\]

for \( i \geq 2 \).

- Unique fixed point: for \( i \geq 1 \)

\[
\pi_i(r) = \lambda \left( \frac{\lambda}{1 + (1 - \lambda)r} \right)^{i-1}.
\]
Overall probe rate $R$

Let $R$ be the number of probes transmitted per unit of time.
Overall probe rate $R$

Let $R$ be the number of probes transmitted per unit of time

- Rate-based work stealing:

$$R_{\text{steal}} = (1 - \lambda)r_{\text{steal}}$$
Overall probe rate $R$

Let $R$ be the number of probes transmitted per unit of time

- Rate-based work stealing:
  
  $$R_{\text{steal}} = (1 - \lambda)r_{\text{steal}}$$

- Rate-based work sharing:
  
  $$R_{\text{share}} = r_{\text{share}}\pi_2(r_{\text{share}}) = \frac{\lambda^2 r_{\text{share}}}{1 + (1 - \lambda)r_{\text{share}}}.$$
Overall probe rate $R$

Let $R$ be the number of probes transmitted per unit of time

- Rate-based work stealing:

$$R_{steal} = (1 - \lambda)r_{steal}$$

- Rate-based work sharing:

$$R_{share} = r_{share}\pi_2(r_{share}) = \frac{\lambda^2 r_{share}}{1 + (1 - \lambda)r_{share}}.$$  

- Traditional work sharing:

$$R_{trad,\text{share}} = \lambda^2 \left(1 + \sum_{i=1}^{L_p-1} \lambda^i\right) = \lambda^2 \frac{1 - \lambda^{L_p}}{1 - \lambda}.$$
Comparing strategies

Given $\lambda$ and some $R$

Rate-based strategies:
set $r_{share}$ and $r_{steal}$ such that $R_{steal} = R_{share} = R$

Work sharing strategies:
set $r_{share} = 1 - \lambda L_p (1 - \lambda L_p)$
such that $R_{share} = R_{trad,share}$

Remarkably, $\pi_i + 1 \cdot (r_{share}) = \lambda \cdot \frac{1}{1 + (1 + L_p) i}$,
so if overall probe rate is matched, we get the same limiting queue length distribution. Same holds for work rate-based versus traditional stealing.
Given $\lambda$ and some $R$

- Rate-based strategies:
  
  set $r_{\text{share}}$ and $r_{\text{steal}}$ such that $R_{\text{steal}} = R_{\text{share}} = R$
Comparing strategies

Given $\lambda$ and some $R$

- Rate-based strategies:
  
  set $r_{share}$ and $r_{steal}$ such that $R_{steal} = R_{share} = R$

- Work sharing strategies:

  set $r_{share} = \frac{1-\lambda^{L_p}}{(1-\lambda)\lambda^{L_p}}$ such that $R_{share} = R_{trad,share}$
Comparing strategies

Given $\lambda$ and some $R$

- Rate-based strategies:
  
  set $r_{\text{share}}$ and $r_{\text{steal}}$ such that $R_{\text{steal}} = R_{\text{share}} = R$

- Work sharing strategies:
  
  set $r_{\text{share}} = \frac{1-\lambda^{L_p}}{(1-\lambda)\lambda^{L_p}}$ such that $R_{\text{share}} = R_{\text{trad,share}}$

Remarkably,

$$\pi^{i+1}(r_{\text{share}}) = \lambda^{1+(1+L_p)i},$$

so if overall probe rate is matched, we get the same limiting queue length distribution. Same holds for work rate-based versus traditional stealing.
Theorem (Minnebo, VH. 2014): The mean response time $D$ of a job under sharing equals

$$D_{share} = \frac{\lambda}{(1 - \lambda)(\lambda + R)},$$

for $R < \frac{\lambda^2}{1 - \lambda}$ and $D_{share} = 1$ for $R \geq \frac{\lambda^2}{1 - \lambda}$. Under stealing we get

$$D_{steal} = \frac{1 + R}{1 - \lambda + R}.$$  

Hence, given $R$ sharing is best if and only if

$$\lambda < \frac{\sqrt{(1 + R)^2 + 4(1 + R)} - (1 + R)}{2}.$$  

Further, for any $R$, sharing outperforms stealing for all $\lambda < \phi - 1$, where $\phi = \frac{1 + \sqrt{5}}{2}$ is the golden ratio.
Exponential job sizes (mean 1): boundary at $R = \max(\frac{\lambda^2}{1-\lambda} - 1, 0)$
Finite system accuracy (overall probe rate $R=1$)

$\begin{array}{c}
\lambda = 0.7 \\
0.035 \\
0.017 \\
0.009 \\
0.004 \\
0.002 \\
0.001 \\
0.0006
\end{array}
$

$\begin{array}{c}
\lambda = 0.8 \\
0.065 \\
0.032 \\
0.016 \\
0.008 \\
0.004 \\
0.002 \\
0.001
\end{array}
$

$\begin{array}{c}
\lambda = 0.9 \\
0.035 \\
0.017 \\
0.009 \\
0.004 \\
0.002 \\
0.001 \\
0.0006
\end{array}
$

$\begin{array}{c}
\lambda = 0.95 \\
0.035 \\
0.017 \\
0.009 \\
0.004 \\
0.002 \\
0.001 \\
0.0006
\end{array}
$

⇒ Can be further improved by refined mean field approximation
Finite system accuracy (overall probe rate $R=1$)

⇒ Good prediction of border between 2 regions for $N = 100$ servers
Outline

1. Setting
2. Traditional strategies
3. Rate-based Strategies
4. Global attraction
5. Non-exponential job sizes
Simple proof by monotonicity

- Traditional work sharing: set of ODEs:

\[
\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^L_p) - (s_1(t) - s_2(t))
\]

\[
\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))s_1(t)^L_p - (s_i(t) - s_{i+1}(t))
\]

for \( i \geq 2 \).
Simple proof by monotonicity

- Traditional work sharing: set of ODEs:

\[
\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^{L_p}) - (s_1(t) - s_2(t))
\]

\[
\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))s_1(t)^{L_p} - (s_i(t) - s_{i+1}(t))
\]

for \( i \geq 2 \).

- To simplify matters, let’s truncate the queues at length \( B \).
Simple proof by monotonicity

- Traditional work sharing: set of ODEs:

\[
\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^L_p) - (s_1(t) - s_2(t))
\]

\[
\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))s_1(t)^L_p - (s_i(t) - s_{i+1}(t))
\]

for \(i \geq 2\).

- To simplify matters, let’s truncate the queues at length \(B\) ⇒ Same set of ODEs applies, but with \(s_{B+1}(t) = 0\)
Simple proof by monotonicity

- Global attraction: show $\lim_{t \to \infty} s(t) = \pi$, the unique fixed point, for any initial $s(0) \in \{(s_1, \ldots, s_B)|1 \geq s_1 \geq \ldots \geq s_B \geq 0\}$
Simple proof by monotonicity

- Global attraction: show \( \lim_{t \to \infty} s(t) = \pi \), the unique fixed point, for any initial \( s(0) \in \{ (s_1, \ldots, s_B) | 1 \geq s_1 \geq \ldots \geq s_B \geq 0 \} \)

- Componentwise partial order: \( s \leq \tilde{s} \) with \( s = (s_1, \ldots, s_B) \) and \( \tilde{s} = (\tilde{s}_1, \ldots, \tilde{s}_B) \) if \( s_i \leq \tilde{s}_i \) for all \( i \)
Simple proof by monotonicity

- **Global attraction**: show $\lim_{t \to \infty} s(t) = \pi$, the unique fixed point, for any initial $s(0) \in \{(s_1, \ldots, s_B)| 1 \geq s_1 \geq \ldots \geq s_B \geq 0\}$

- **Componentwise partial order**: $s \leq \tilde{s}$ with $s = (s_1, \ldots, s_B)$ and $\tilde{s} = (\tilde{s}_1, \ldots, \tilde{s}_B)$ if $s_i \leq \tilde{s}_i$ for all $i$

- Let $s(t)$ and $\tilde{s}(t)$ be the unique solution of the set of ODEs with $s(0) = s$ and $\tilde{s}(0) = \tilde{s}$, respectively.
Simple proof by monotonicity

- Global attraction: show $\lim_{t \to \infty} s(t) = \pi$, the unique fixed point, for any initial $s(0) \in \{(s_1, \ldots, s_B)|1 \geq s_1 \geq \ldots \geq s_B \geq 0\}

- Componentwise partial order: $s \leq \tilde{s}$ with $s = (s_1, \ldots, s_B)$ and $\tilde{s} = (\tilde{s}_1, \ldots, \tilde{s}_B)$ if $s_i \leq \tilde{s}_i$ for all $i$

- Let $s(t)$ and $\tilde{s}(t)$ be the unique solution of the set of ODEs with $s(0) = s$ and $\tilde{s}(0) = \tilde{s}$, respectively.

- Let $s_E(t)$ and $s_F(t)$ be the unique solution of the set of ODEs with $s_E(0) = (0, \ldots, 0)$ and $s_F(0) = (0, \ldots, 0, 1)$, respectively.
Simple proof by monotonicity: Step 1

- **STEP 1:** show that partial order is preserved over time, that is,

\[ s(t) \leq \tilde{s}(t) \text{ for all } t \geq 0 \text{ if } s(0) \leq \tilde{s}(0) \]
Simple proof by monotonicity: Step 1

- **STEP 1**: show that partial order is preserved over time, that is,

\[ s(t) \leq \tilde{s}(t) \text{ for all } t \geq 0 \text{ if } s(0) \leq \tilde{s}(0) \]

- How: verify that the drift of \( s_i(t) \) is increasing in \( s_j(t) \) for \( j \neq i \)
Simple proof by monotonicity: Step 1

- **STEP 1**: show that partial order is preserved over time, that is,

  \[ s(t) \leq \tilde{s}(t) \text{ for all } t \geq 0 \text{ if } s(0) \leq \tilde{s}(0) \]

- How: verify that the drift of \( s_i(t) \) is increasing in \( s_j(t) \) for \( j \neq i \)

- Let’s do this:

\[
\frac{ds_1(t)}{dt} = \lambda(1 - s_1(t)) + \lambda s_1(t)(1 - s_1(t)^{L_p}) - (s_1(t) - s_2(t))
\]

\[
\frac{ds_i(t)}{dt} = \lambda(s_{i-1}(t) - s_i(t))s_1(t)^{L_p} - (s_i(t) - s_{i+1}(t))
\]

for \( i \geq 2 \).
Simple proof by monotonicity: Step 2

- **STEP 2**: Show that $s_E(s) \leq s_E(t)$ and $s_F(s) \geq s_F(t)$ for $0 < s < t$
Simple proof by monotonicity: Step 2

- **STEP 2**: Show that $s_E(s) \leq s_E(t)$ and $s_F(s) \geq s_F(t)$ for $0 < s < t$

- **How**: immediate by Step 1 as

  $$(0, \ldots, 0) \leq s_E(t - s) \text{ implies that } s_E(s) \leq s_E(t)$$

  and

  $$(0, \ldots, 1) \geq s_F(t - s) \text{ implies that } s_F(s) \geq s_F(t)$$
Simple proof by monotonicity: Step 2

- **STEP 2**: Show that $s_E(s) \leq s_E(t)$ and $s_F(s) \geq s_F(t)$ for $0 < s < t$

- How: immediate by Step 1 as

  $$(0, \ldots, 0) \leq s_E(t - s) \text{ implies that } s_E(s) \leq s_E(t)$$

  and

  $$(0, \ldots, 1) \geq s_F(t - s) \text{ implies that } s_F(s) \geq s_F(t)$$

$\Rightarrow$ As we are working in subset of $[0, 1]^B$, one can check that Step 2 implies

$$\lim_{t \to \infty} s_E(t) = \lim_{t \to \infty} s_F(t) = \pi,$$

where $\pi$ is the unique fixed point
Simple proof by monotonicity: Step 3

- **STEP 3**: Argue that

\[
\lim_{t \to \infty} s_E(t) = \lim_{t \to \infty} s_F(t) = \pi,
\]

implies global attraction due to Step 1.
Simple proof by monotonicity: Step 3

- **STEP 3:** Argue that

\[
\lim_{t \to \infty} s_E(t) = \lim_{t \to \infty} s_F(t) = \pi,
\]

implies global attraction due to Step 1

- **How:** for any \( s = (s_1, \ldots, s_B) \) we have \( s_E(0) \leq s \leq s_F(0) \)
**Simple proof by monotonicity: Step 3**

- **STEP 3**: Argue that

\[
\lim_{t \to \infty} s_E(t) = \lim_{t \to \infty} s_F(t) = \pi,
\]

implies global attraction due to Step 1

- How: for any \( s = (s_1, \ldots, s_B) \) we have \( s_E(0) \leq s \leq s_F(0) \)

- Hence, by Step 1 we have for all \( t \)

\[
s_E(t) \leq s(t) \leq s_F(t),
\]

Taking limits yields global attraction!
Outline

1. Setting

2. Traditional strategies

3. Rate-based Strategies

4. Global attraction

5. Non-exponential job sizes
Phase-type (PH) distributions

- Characterized by $n \times n$ subgenerator $S$ and stochastic vector $\alpha = (\alpha_1, \ldots, \alpha_n)$
Phase-type (PH) distributions

- Characterized by \( n \times n \) subgenerator \( S \) and stochastic vector \( \alpha = (\alpha_1, \ldots, \alpha_n) \)
- \( \text{cdf} \; H(y) = 1 - \alpha e^{Sy} \mathbf{1} \), where \( \mathbf{1} \) is a vector of ones
- \( \text{pdf} \; h(y) = \alpha e^{Sy} \mu \), where \( \mu = -S \mathbf{1} \)
Phase-type (PH) distributions

- Characterized by $n \times n$ subgenerator $S$ and stochastic vector $\alpha = (\alpha_1, \ldots, \alpha_n)$
- cdf $H(y) = 1 - \alpha e^{Sy} \mathbf{1}$, where $\mathbf{1}$ is a vector of ones
- pdf $h(y) = \alpha e^{Sy} \mu$, where $\mu = -S\mathbf{1}$
- $\alpha_i$ is the probability that a job starts service in phase $i$
Phase-type (PH) distributions

- Characterized by $n \times n$ subgenerator $S$ and stochastic vector $\alpha = (\alpha_1, \ldots, \alpha_n)$
- cdf $H(y) = 1 - \alpha e^{Sy} \mathbb{1}$, where $\mathbb{1}$ is a vector of ones
- pdf $h(y) = \alpha e^{Sy} \mu$, where $\mu = -S \mathbb{1}$
- $\alpha_i$ is the probability that a job starts service in phase $i$
- entry $(i,j)$ of $S$, for $i \neq j$, is the rate at which the job in service changes its service phase from $i$ to $j$
Phase-type (PH) distributions

- Characterized by $n \times n$ subgenerator $S$ and stochastic vector $\alpha = (\alpha_1, \ldots, \alpha_n)$
- cdf $H(y) = 1 - \alpha e^{Sy}1$, where $1$ is a vector of ones
- pdf $h(y) = \alpha e^{Sy}\mu$, where $\mu = -S1$
- $\alpha_i$ is the probability that a job starts service in phase $i$
- entry $(i, j)$ of $S$, for $i \neq j$, is the rate at which the job in service changes its service phase from $i$ to $j$
- $\mu_i$ is the rate at which a job in phase $i$ completes service
Phase-type (PH) distributions

- Characterized by \( n \times n \) subgenerator \( S \) and stochastic vector \( \alpha = (\alpha_1, \ldots, \alpha_n) \)
- cdf \( H(y) = 1 - \alpha e^{Sy}1 \), where \( 1 \) is a vector of ones
- pdf \( h(y) = \alpha e^{Sy} \mu \), where \( \mu = -S1 \)
- \( \alpha_i \) is the probability that a job starts service in phase \( i \)
- entry \( (i, j) \) of \( S \), for \( i \neq j \), is the rate at which the job in service changes its service phase from \( i \) to \( j \)
- \( \mu_i \) is the rate at which a job in phase \( i \) completes service

\( \Rightarrow \) PH distributions are dense in the class of probability distributions on \([0, \infty)\) and many fitting tools exist
$f_{\ell,i}(t)$: fraction of servers in phase $i$ containing exactly $\ell$ jobs at time $t$ and let $\mathbf{f}_\ell(t) = (f_{\ell,1}(t), \ldots, f_{\ell,n}(t))$
Rate-based: mean field model for PH job sizes

- $f_{\ell,i}(t)$: fraction of servers in phase $i$ containing exactly $\ell$ jobs at time $t$ and let $\vec{f}(t) = (f_{\ell,1}(t), \ldots, f_{\ell,n}(t))$
- Set of ODEs:

$$
\begin{align*}
\frac{d}{dt} \vec{f}(t) &= \lambda \vec{f}_{\ell-1}(t) 1[\ell > 1] - \lambda \vec{f}(t) + \lambda f_0(t) \alpha 1[\ell = 1] \\
&+ \vec{f}_{\ell+1}(t) \mu \alpha + rf_0(t)(\vec{f}_{\ell+1}(t) - 1[\ell > 1] \vec{f}(t)) \\
&+ \vec{f}(t) S + 1[\ell = 1] rf_0(t) \left(1 - f_0(t) - \vec{f}_1(t) \mathbb{1}\right) \alpha,
\end{align*}
$$

for $\ell \geq 1$ and

$$
\frac{d}{dt} f_0(t) = -\lambda f_0(t) + \vec{f}_1(t) \mu - rf_0(t) \left(1 - f_0(t) - \vec{f}_1(t) \mathbb{1}\right).
$$
Fixed point for rate-based strategies with PH job sizes

The queueing system has the following characteristics:

- There is a single server, infinite waiting room and service times follow a phase-type distribution \((\alpha, S)\) with mean 1. Customers are served in FCFS order.

Arrivals occur according to a Poisson process with rate \(\lambda\) when the server is busy and at rate \(\lambda_0\) when the server is idle.

Negative arrivals occur at rate \((1 - \lambda) r\) when the queue length exceeds one and reduce the queue length by one (by removing a customer from the back of the queue).

The arrival rate \(\lambda_0\) is such that the probability of having an idle queue is \(1 - \lambda\) and thus depends on \(\lambda, r\) and \((\alpha, S)\) only.
The queueing system has the following characteristics:

- There is a single server, infinite waiting room and service times follow a phase-type distribution \((\alpha, S)\) with mean 1. Customers are served in FCFS order.

- Arrivals occur according to a Poisson process with rate \(\lambda\) when the server is busy and at rate \(\lambda_0\) when the server is idle.
The queueing system has the following characteristics:

- There is a single server, infinite waiting room and service times follow a phase-type distribution \((\alpha, S)\) with mean 1. Customers are served in FCFS order.
- Arrivals occur according to a Poisson process with rate \(\lambda\) when the server is busy and at rate \(\lambda_0\) when the server is idle.
- Negative arrivals occur at rate \((1 - \lambda)r\) when the queue length exceeds one and reduce the queue length by one (by removing a customer from the back of the queue).
The queueing system has the following characteristics:

- There is a single server, infinite waiting room and service times follow a phase-type distribution \((\alpha, S)\) with mean 1. Customers are served in FCFS order.
- Arrivals occur according to a Poisson process with rate \(\lambda\) when the server is busy and at rate \(\lambda_0\) when the server is idle.
- Negative arrivals occur at rate \((1 - \lambda) r\) when the queue length exceeds one and reduce the queue length by one (by removing a customer from the back of the queue).
- The arrival rate \(\lambda_0\) is such that the probability of having an idle queue is \(1 - \lambda\) and thus depends on \(\lambda, r\) and \((\alpha, S)\) only.
Quasi-birth-death (QBD) Markov chain:

\[ Q(r) = \begin{bmatrix}
  -\lambda_0(r) & \lambda_0(r)\alpha & & \\
  \mu & S - \lambda I & A_1 & \\
  A_{-1}(r) & A_0(r) & A_1 & \\
  & & & \ddots & \ddots & \ddots
\end{bmatrix}, \]

with

\[ A_{-1}(r) = \mu\alpha + (1 - \lambda)rI, \]
\[ A_0(r) = S - \lambda I + (1 - \lambda)rI, \]
\[ A_1 = \lambda I. \]
Stationary distribution:

\[ \pi_\ell(r) = \lambda \frac{\alpha(\lambda(I - G(r)) - S)^{-1} R(r)^{\ell-1}}{\alpha(\lambda(I - G(r)) - S)^{-1}(I - R(r))^{-1}1}, \]  

(1)

and \( \pi_0(r) = 1 - \lambda \), with

\[ A_1 + R(r)A_0(r) + R(r)^2 A_{-1}(r) = 0 \]

and \( \lambda G(r) = R(r)A_{-1}(r) \)

Theorem (VH. 2018): The steady state probability vector given by (1) is the unique fixed point \( \zeta \) of the set of ODEs with \( \zeta_0 + \sum_{\ell \geq 1} \tilde{\zeta}_\ell \|_1 = 1. \)
Theorem (VH. 2018): Given $(\alpha, S)$, $\lambda$ and $R > 0$, work sharing achieves a lower mean response time than stealing if and only if

$$1 - \lambda > \pi_{2+}(R/(1 - \lambda))_1.$$  \hfill (2)

$\Rightarrow$ Suffices to solve single QBD to decide

Theorem (VH. 2018): Given $(\alpha, S)$ and $\lambda$ there exists a $R^*$ such that work sharing is best if and only if $R > R^*$.

Theorem (VH. 2018): Given $(\alpha, S)$ and $R$ there exists a $\lambda^*$ such that work sharing is best if and only if $\lambda < \lambda^*$. 
Rate-based stealing vs sharing with PH job sizes

⇒ stealing benefits from more variability in job sizes
Rate-based stealing vs sharing with PH job sizes

⇒ boundary depends on higher moments, as expected
General boundaries for PH job sizes

Theorem (VH. 2018): For any \((\alpha, S)\), work sharing is best if

\[
\lambda < \frac{\max(1, \sqrt{r_{overall}(r_{overall} + 4)} - r_{overall})}{2}.
\]
General boundaries for PH job sizes

Conjectures:

⇒ Have weaker bounds and limit results for $r$ tending to zero
How to prove the general stealing bound?

Consider the following queueing system:

- There is a single server, infinite waiting room and service times have a general distribution with mean 1. Customers are served in FCFS order.
How to prove the general stealing bound?

Consider the following queueing system:

- There is a single server, infinite waiting room and service times have a general distribution with mean 1. Customers are served in FCFS order.
- Arrivals occur according to a Poisson process with rate $\lambda$.
- Negative arrivals occur at rate $\lambda - \lambda$ and remove a pending customer, if present.

$\Rightarrow$ Show that the probability to have exactly one job in the queue is maximized when the job length is deterministic!

Easy when $\lambda - \lambda = 0$ (via P-K formula and Jensen’s inequality)
How to prove the general stealing bound?

Consider the following queueing system:

- There is a single server, infinite waiting room and service times have a general distribution with mean 1. Customers are served in FCFS order.
- Arrivals occur according to a Poisson process with rate $\lambda_+$. 
- Negative arrivals occur at rate $\lambda_-$ and remove a pending customer, if present.

Show that the probability to have exactly one job in the queue is maximized when the job length is deterministic! Easy when $\lambda_- = 0$ (via P-K formula and Jensen’s inequality)
Consider the following queueing system:

- There is a single server, infinite waiting room and service times have a general distribution with mean 1. Customers are served in FCFS order.
- Arrivals occur according to a Poisson process with rate $\lambda_+$. 
- Negative arrivals occur at rate $\lambda_-$ and remove a pending customer, if present.
- When the server becomes idle, we instantaneously insert a new job.
How to prove the general stealing bound?

Consider the following queueing system:

- There is a single server, infinite waiting room and service times have a general distribution with mean 1. Customers are served in FCFS order.
- Arrivals occur according to a Poisson process with rate $\lambda_+$. Negative arrivals occur at rate $\lambda_-$ and remove a pending customer, if present.
- When the server becomes idle, we instantaneously insert a new job.

$\Rightarrow$ Show that the probability to have exactly one job in the queue is maximized when the job length is deterministic!

Easy when $\lambda_- = 0$ (via P-K formula and Jensen’s inequality)
Some references