On Routing Policies for Synchronized Queues

Mauro Escobar

Advisor: Mariana Olvera-Cravioto

Industrial Engineering and Operations Research
Columbia University

November 2nd, 2015

Plan

1. Motivation
2. Model
3. Preliminary Results
4. Numerical Results
5. Conclusions
6. References

Motivation

- Cloud computing

Motivation

- Cloud computing
- Large network of parallel servers

Motivation

- Cloud computing
- Large network of parallel servers
- Cost of information, single centralized queue is not scalable

Plan

1. Motivation
2. Model
3. Preliminary Results
4. Numerical Results
5. Conclusions
6. References

Model

Sequence of queueing networks indexed by number of servers, n:

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel
- Service time: general distribution S

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel
- Service time: general distribution S
- Arrivals: Poisson process with rate λn

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel
- Service time: general distribution S
- Arrivals: Poisson process with rate λn
- Each job is split into a random number of pieces k, according to some distribution $f_n(k)$, $k \in \{1, 2, \ldots, m_n\}$, $m_n \leq n$

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel
- Service time: general distribution S
- Arrivals: Poisson process with rate λn
- Each job is split into a random number of pieces k, according to some distribution $f_n(k)$, $k \in \{1, 2, \ldots, m_n\}$, $m_n \leq n$
- Each piece is routed to k different servers

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel
- Service time: general distribution S
- Arrivals: Poisson process with rate λn
- Each job is split into a random number of pieces k, according to some distribution $f_n(k)$, $k \in \{1, 2, \ldots, m_n\}$, $m_n \leq n$
- Each piece is routed to k different servers
- **Routing must occur at arrival, no centralize queue**

Model

Sequence of queueing networks indexed by number of servers, n:

- Identical servers, operate in parallel
- Service time: general distribution S
- Arrivals: Poisson process with rate λn
- Each job is split into a random number of pieces k, according to some distribution $f_n(k)$, $k \in \{1, 2, \ldots, m_n\}$, $m_n \leq n$
- Each piece is routed to k different servers
- **Routing must occur at arrival, no centralize queue**
- All pieces must start service at the same time, FCFS

Model

We consider 3 different Routing Schemes:

Model

We consider 3 different Routing Schemes:

- **Optimal**: Pieces are routed to servers with least workload

Model

We consider 3 different Routing Schemes:

- **Optimal**: Pieces are routed to servers with least workload
- **Blind**: Pieces are routed randomly

Model

We consider 3 different Routing Schemes:

- **Optimal**: Pieces are routed to servers with least workload
- **Blind**: Pieces are routed randomly
- **Intermediate**: First route pieces to idle servers, then randomly route the remaining pieces

Model

We consider 3 different Routing Schemes:

- **Optimal**: Pieces are routed to servers with least workload
- **Blind**: Pieces are routed randomly
- **Intermediate**: First route pieces to idle servers, then randomly route the remaining pieces

Cost of Information vs. Service Quality?

Model

We consider 3 different Routing Schemes:

- **Optimal**: Pieces are routed to servers with least workload
- **Blind**: Pieces are routed randomly
- **Intermediate**: First route pieces to idle servers, then randomly route the remaining pieces

Cost of Information vs. Service Quality?

We analize the **expected waiting time** in the 3 cases.

Model: Optimal Routing Scheme

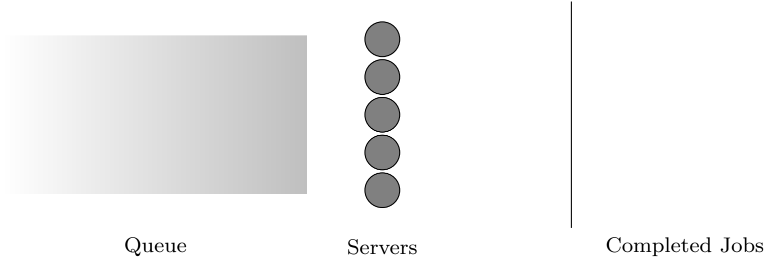
Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

- Equivalently, jobs wait in a unique queue until the number of servers needed are free

Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

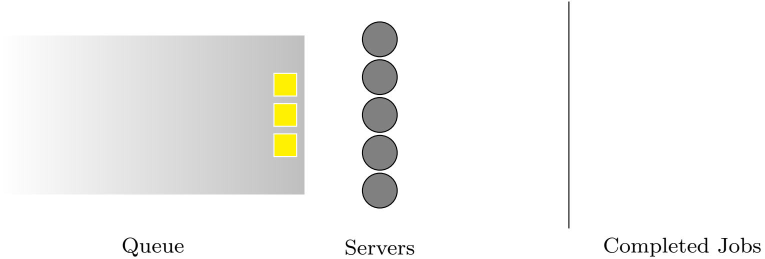
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

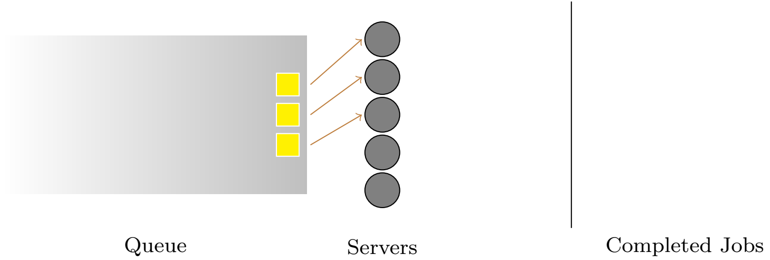
Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Analyzed by Green [G80] with service time of pieces $S \sim \mathrm{Exp}(\mu)$.

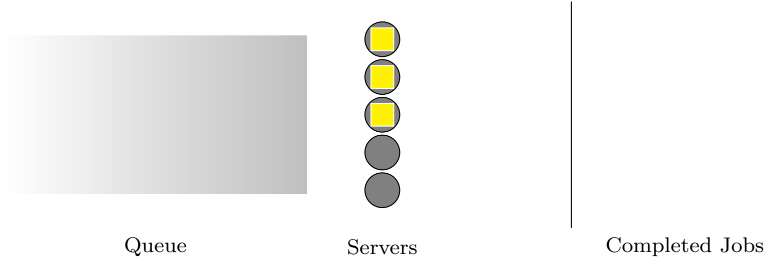
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \mathrm{Exp}(\mu)$.

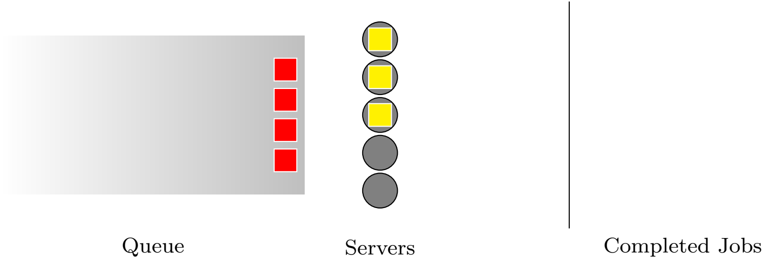
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

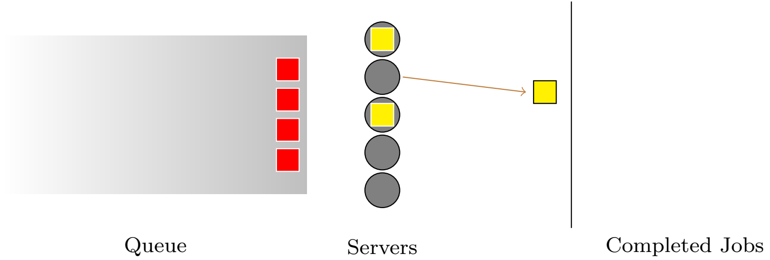
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

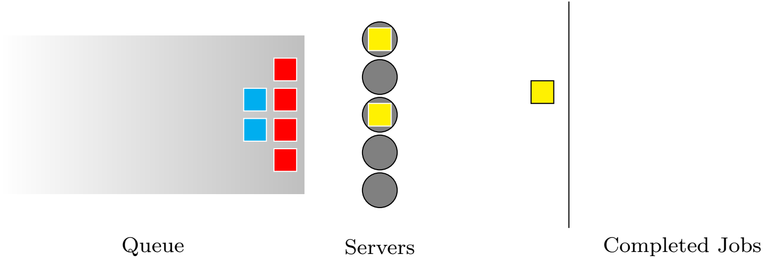
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

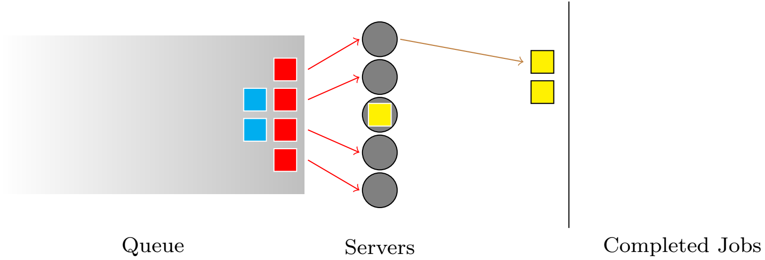
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \mathrm{Exp}(\mu)$.

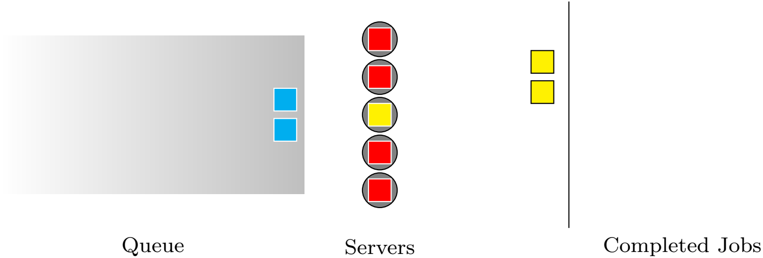
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

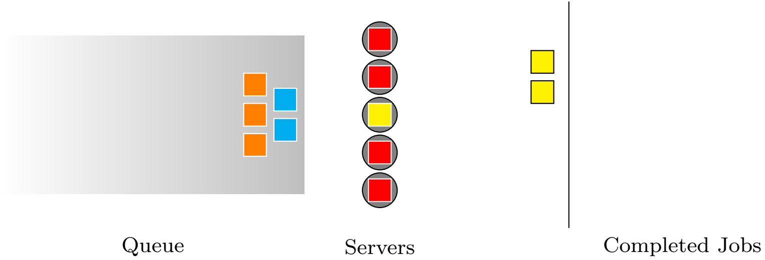
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

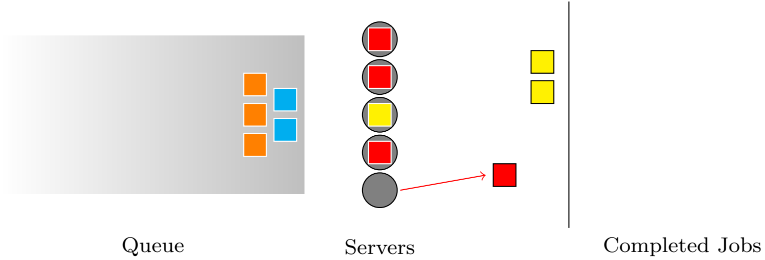
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

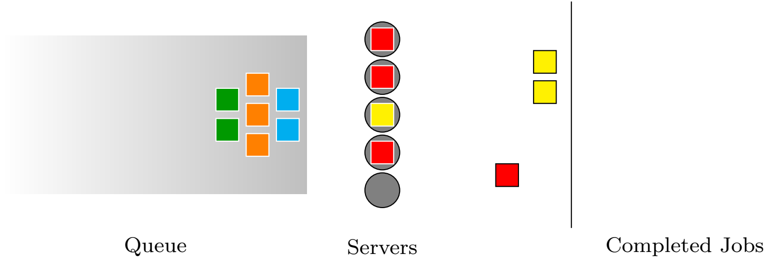
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

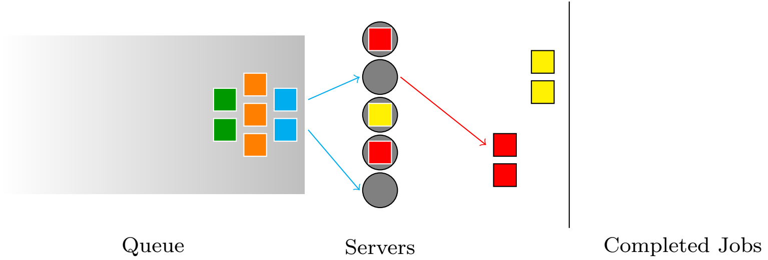
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

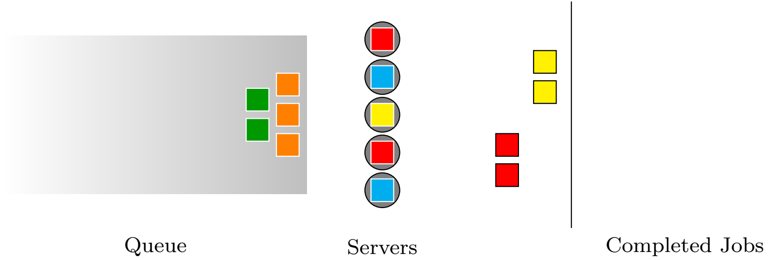
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \mathrm{Exp}(\mu)$.

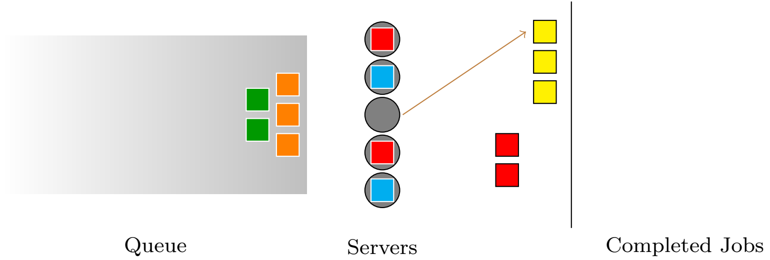
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \mathrm{Exp}(\mu)$.

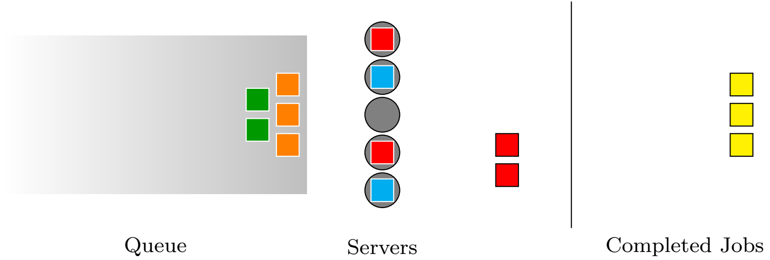
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \mathrm{Exp}(\mu)$.

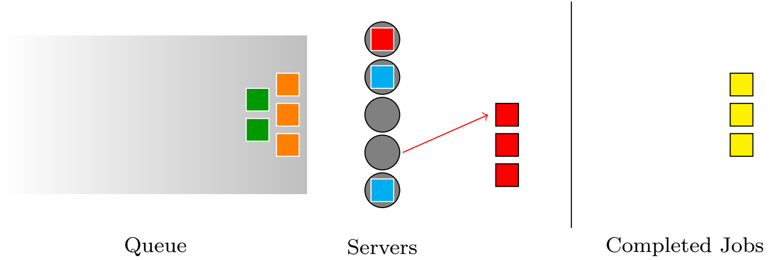
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

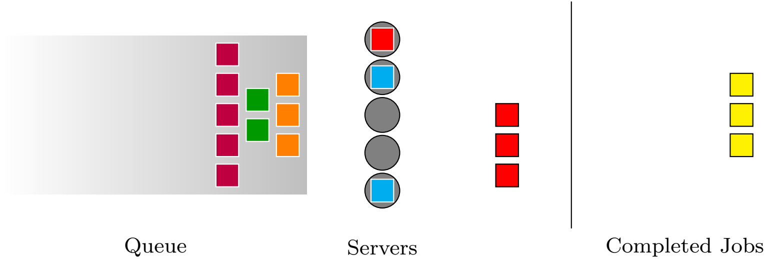
- Equivalently, jobs wait in a unique queue until the number of servers needed are free



Model: Optimal Routing Scheme

Analyzed by Green [G80] with service time of pieces $S \sim \text{Exp}(\mu)$.

- Equivalently, jobs wait in a unique queue until the number of servers needed are free



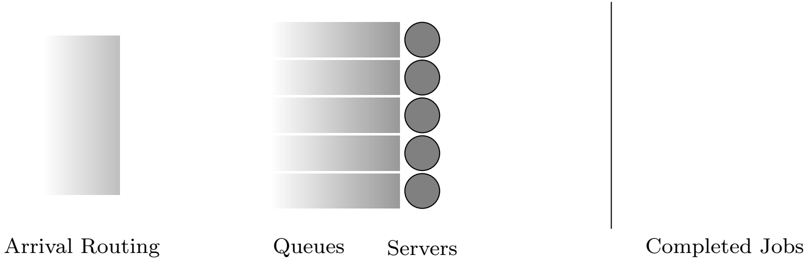
Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

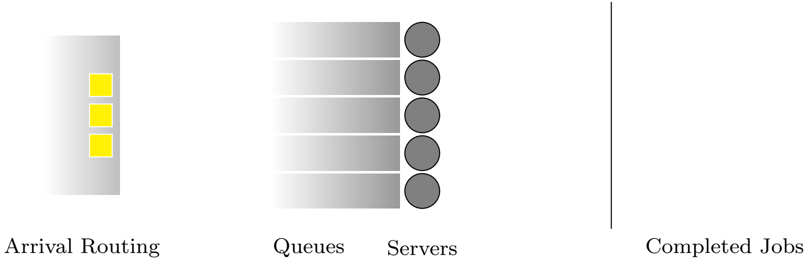
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

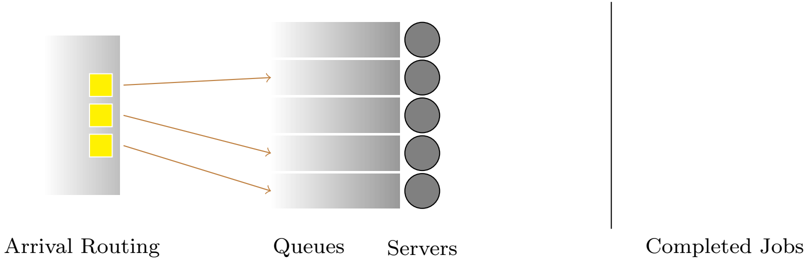
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

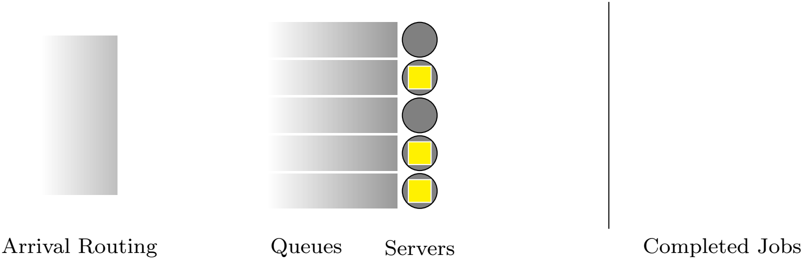
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviato and Ruiz-Lacedelli [OR15].

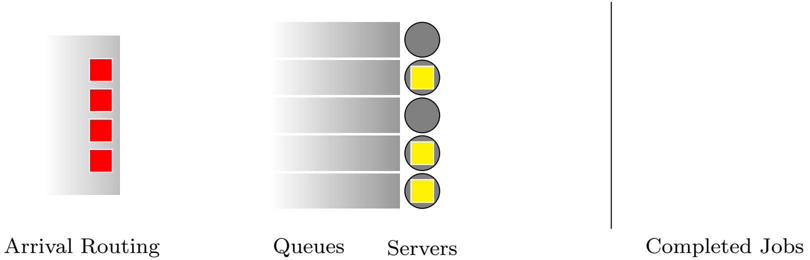
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

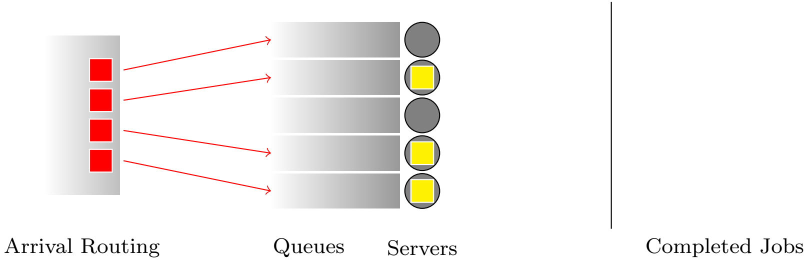
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

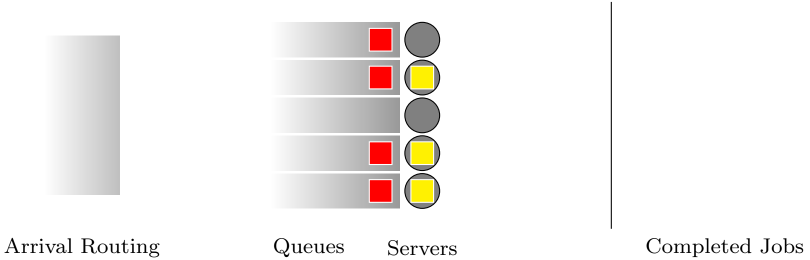
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

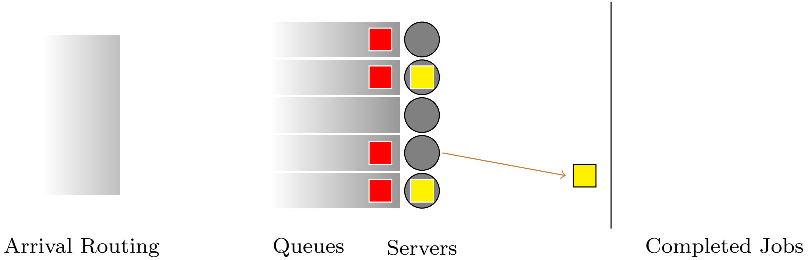
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

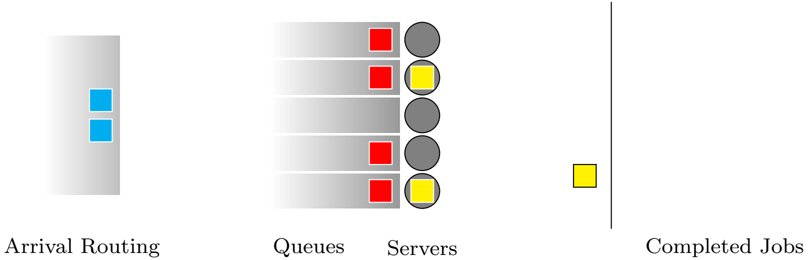
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

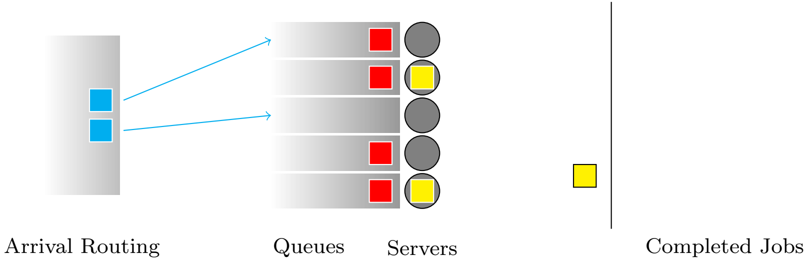
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

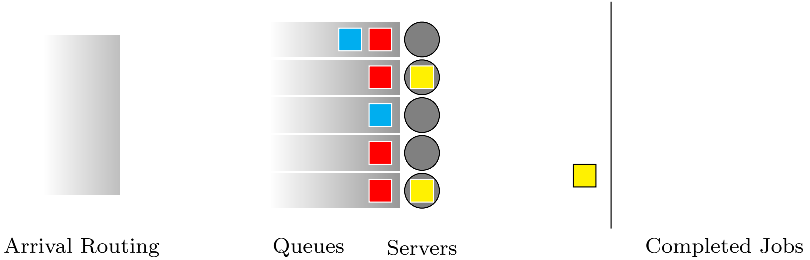
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

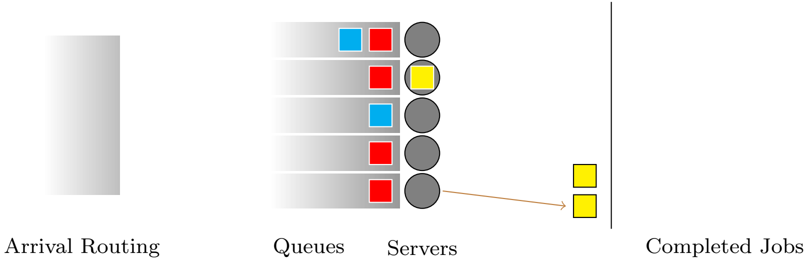
Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

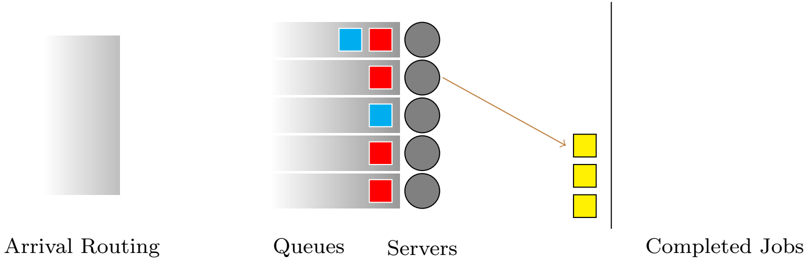
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

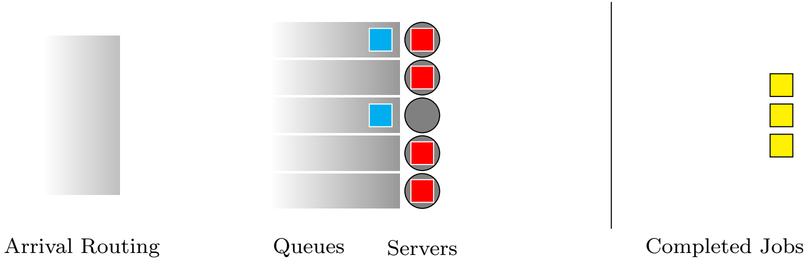
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

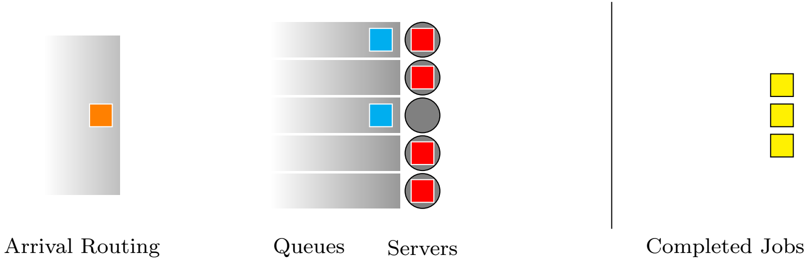
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

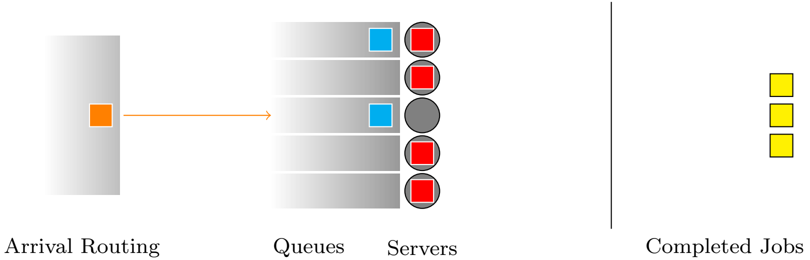
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

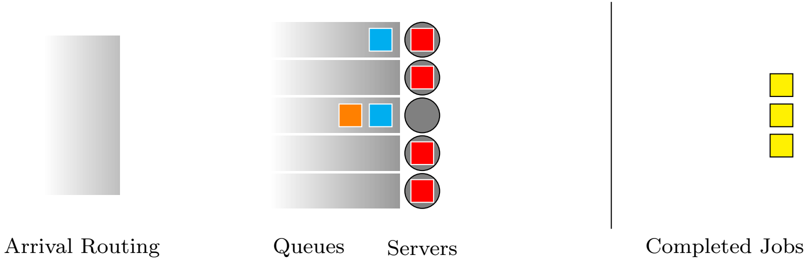
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

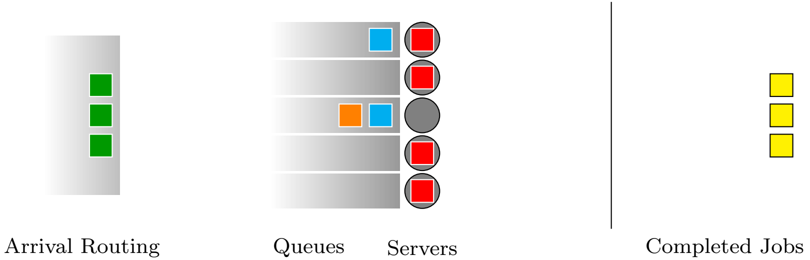
Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

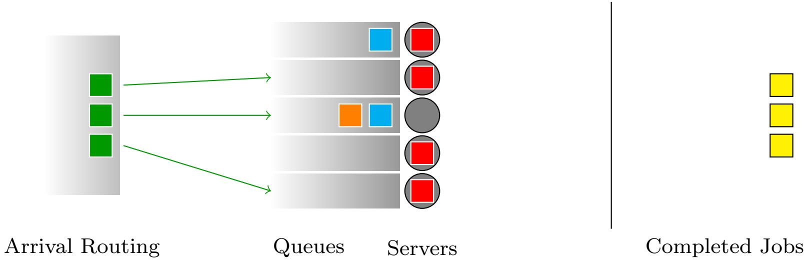
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

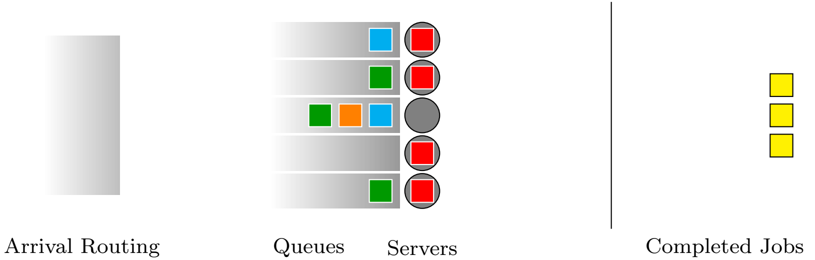
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

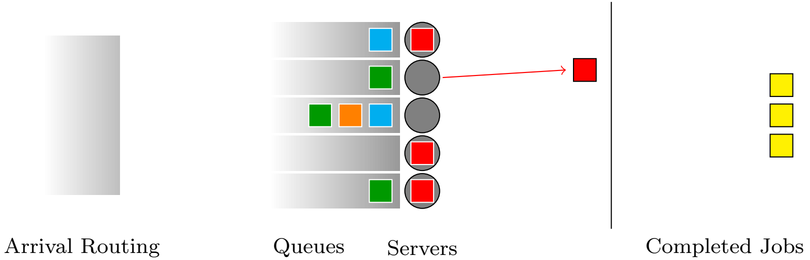
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

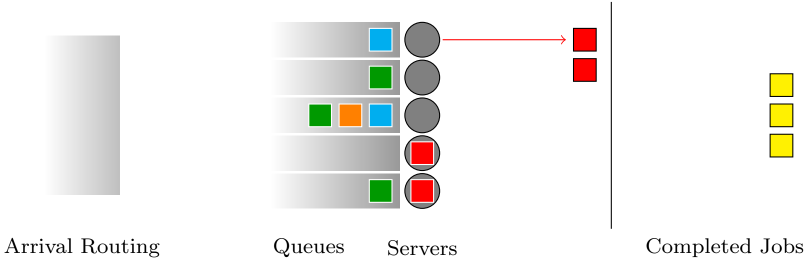
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

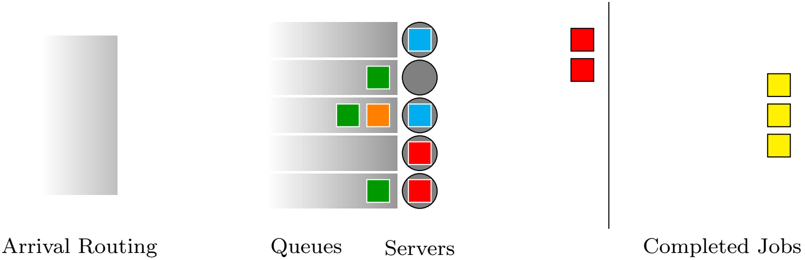
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

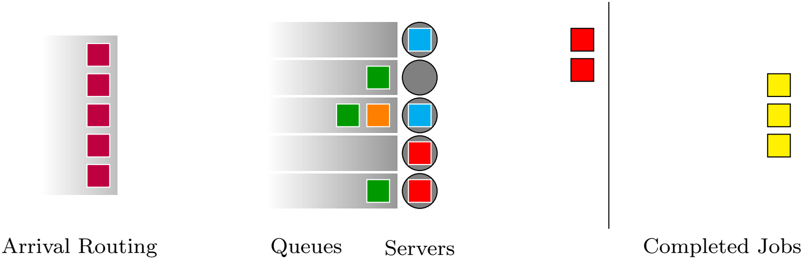
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

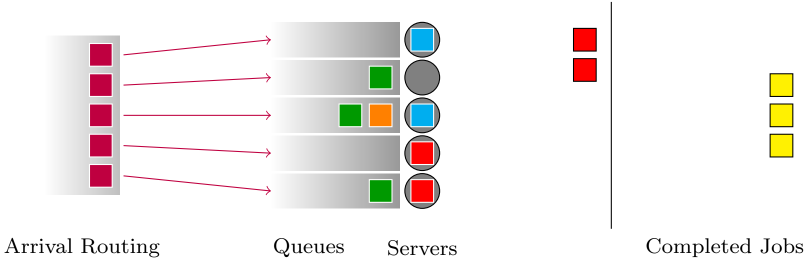
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

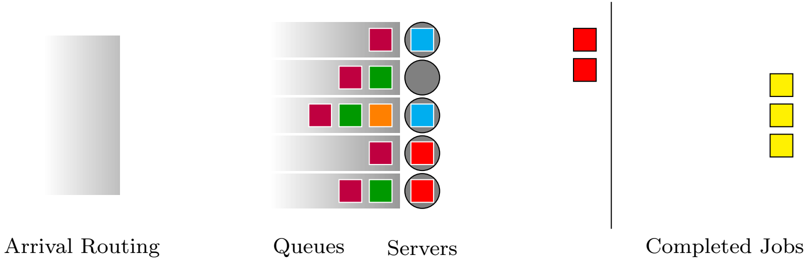
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviote and Ruiz-Lacedelli [OR15].

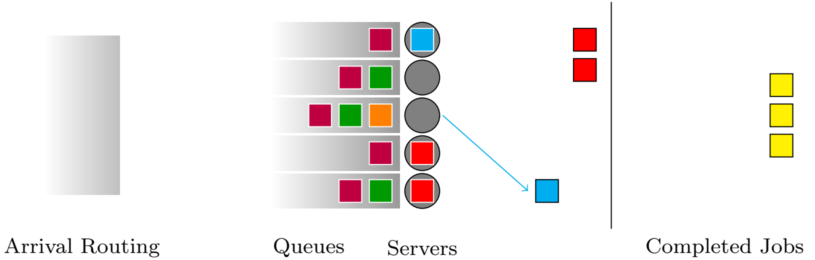
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

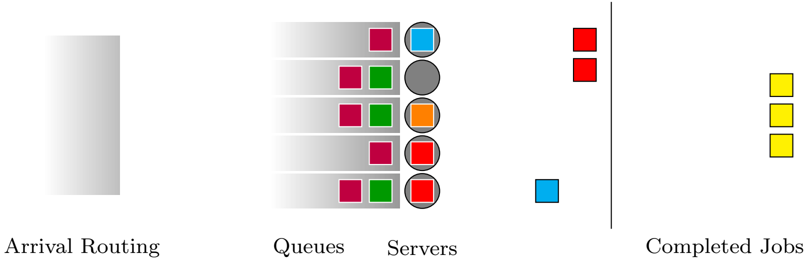
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

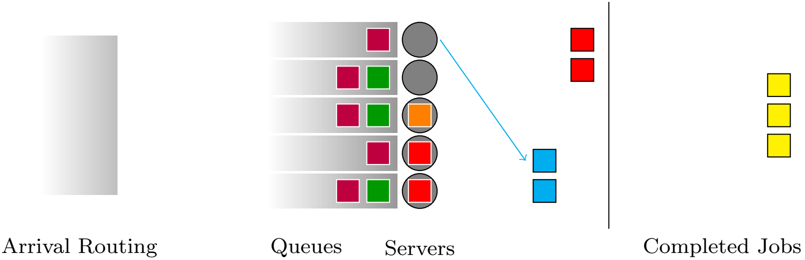
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

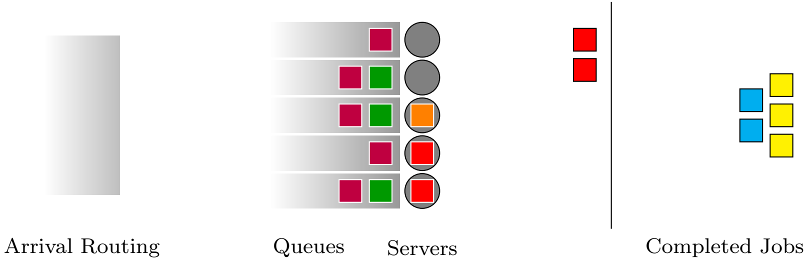
Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Analyzed by Olvera-Craviote and Ruiz-Lacedelli [OR15].

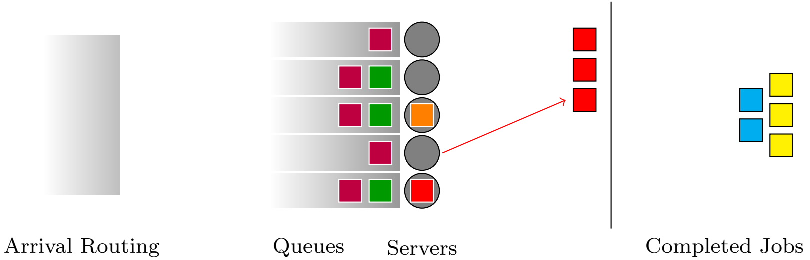
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

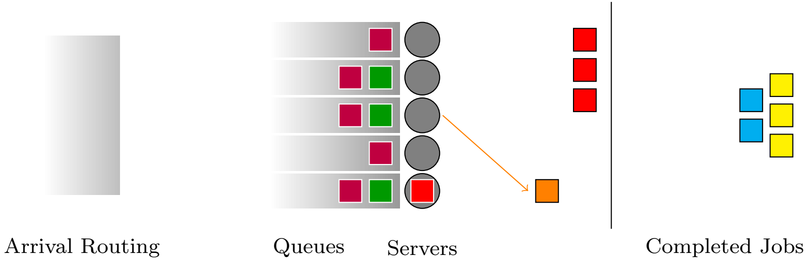
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

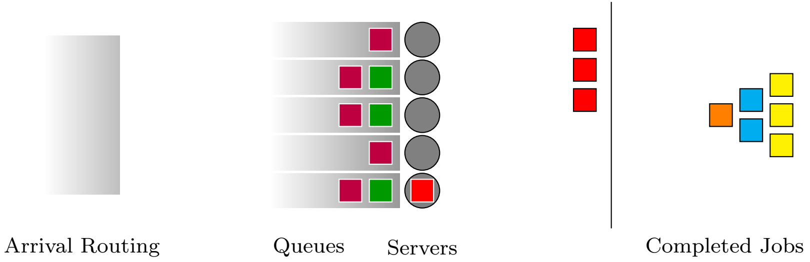
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

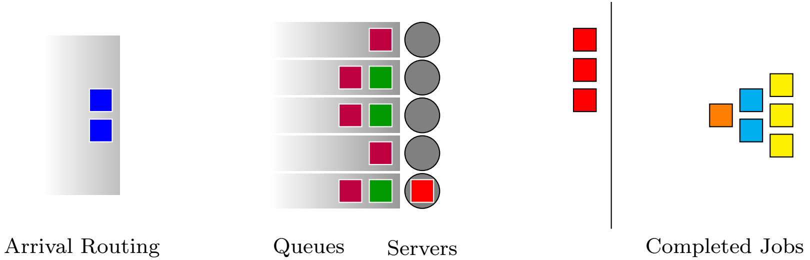
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

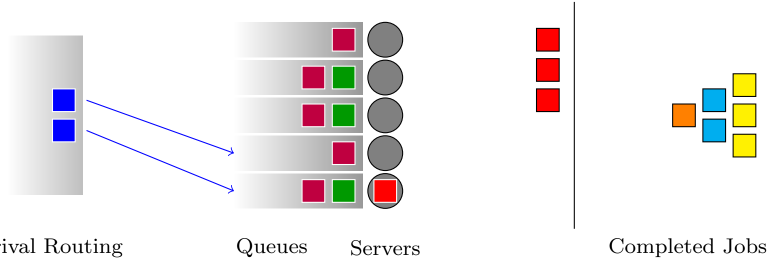
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Craviotto and Ruiz-Lacedelli [OR15].

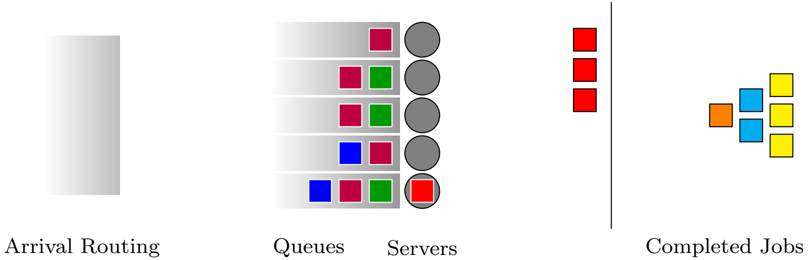
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

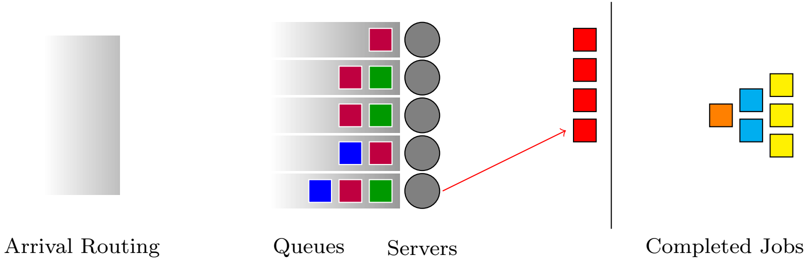
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

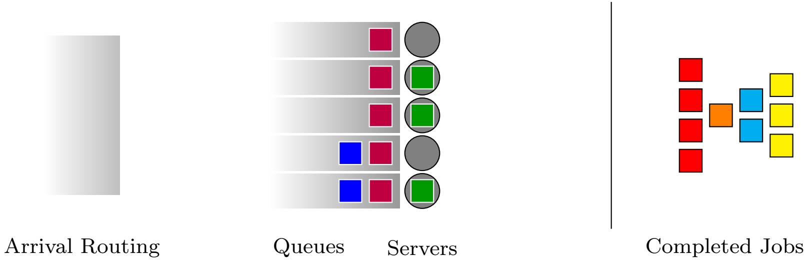
- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



Model: Blind Routing Scheme

Analyzed by Olvera-Cravioto and Ruiz-Lacedelli [OR15].

- k pieces of a job are routed randomly to k different servers at arrival, forming a queue at their assigned servers



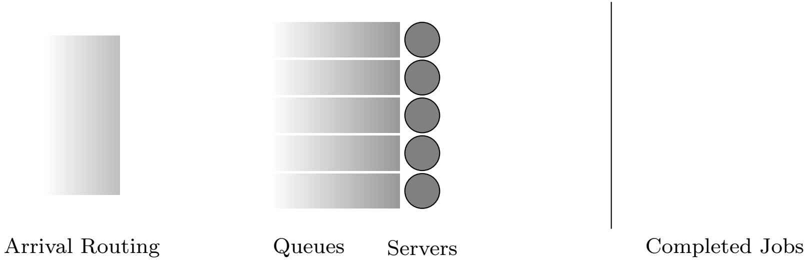
Model: Intermediate Routing Scheme

We propose a mixed scheme.

- First route pieces to idle servers, then randomly route the remaining pieces

We propose a mixed scheme.

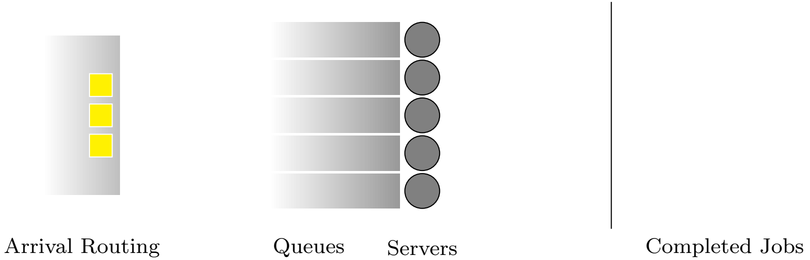
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

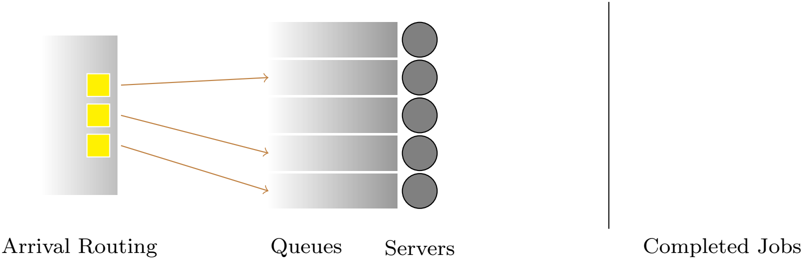
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

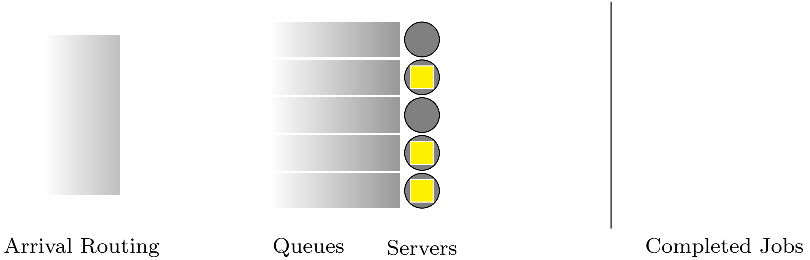
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

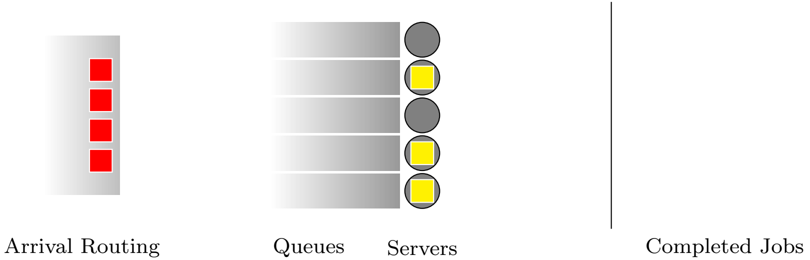
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

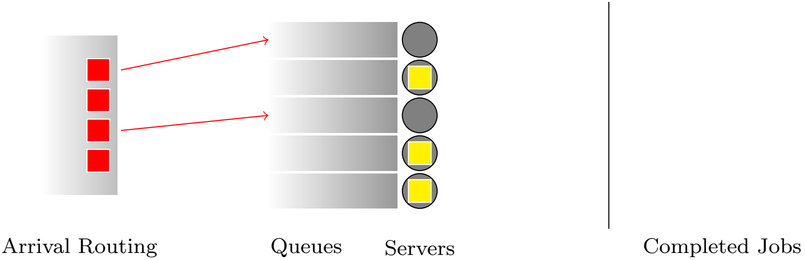
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

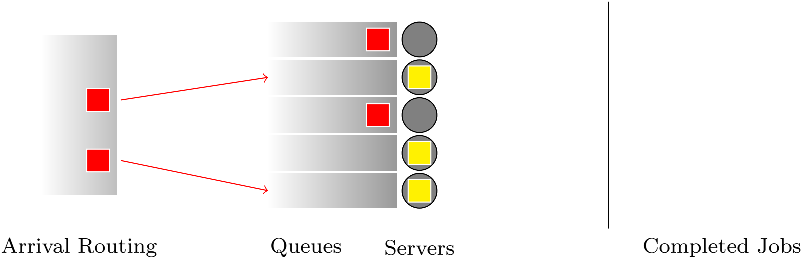
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

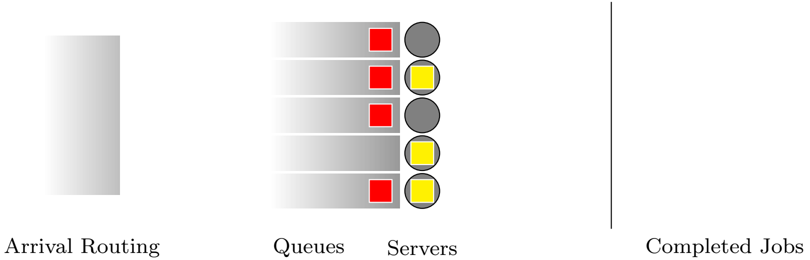
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

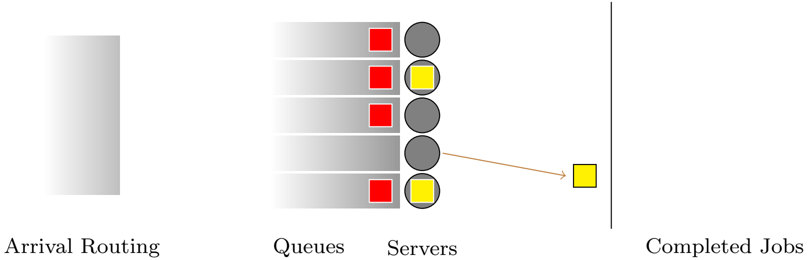
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

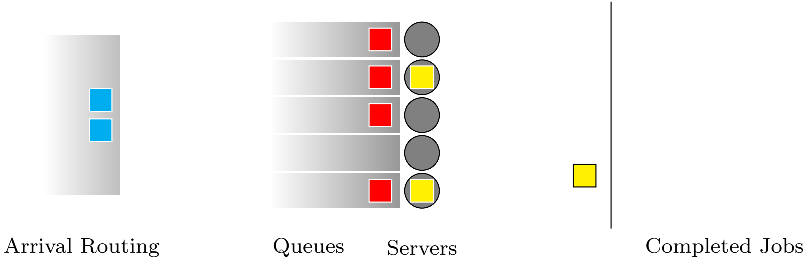
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

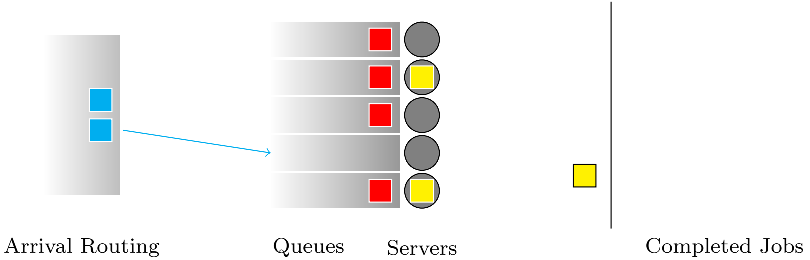
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

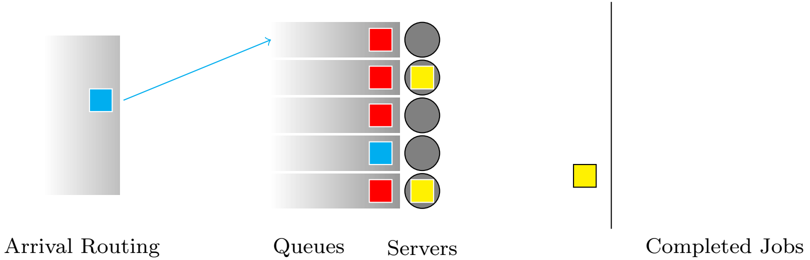
We propose a mixed scheme.

- First route pieces to idle servers, then randomly route the remaining pieces



We propose a mixed scheme.

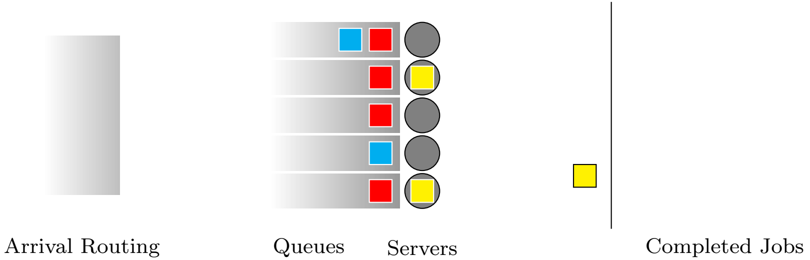
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

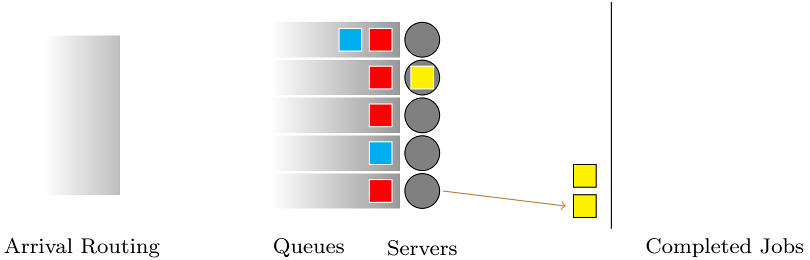
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

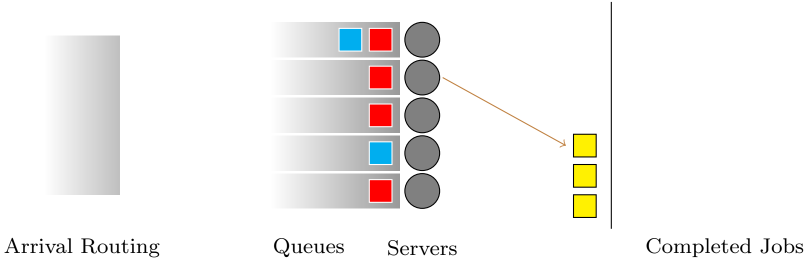
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

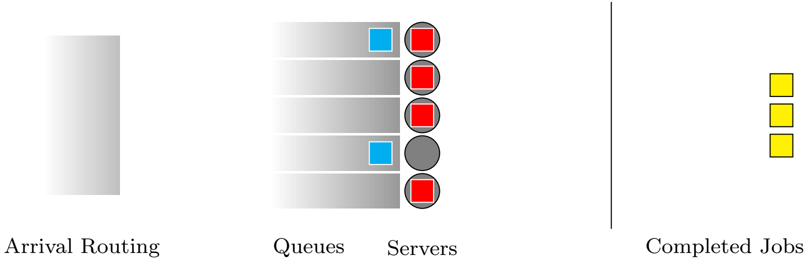
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

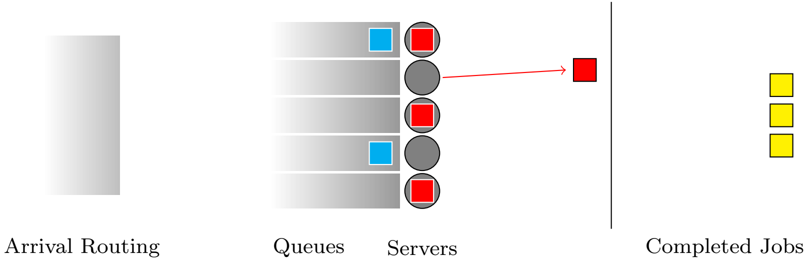
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

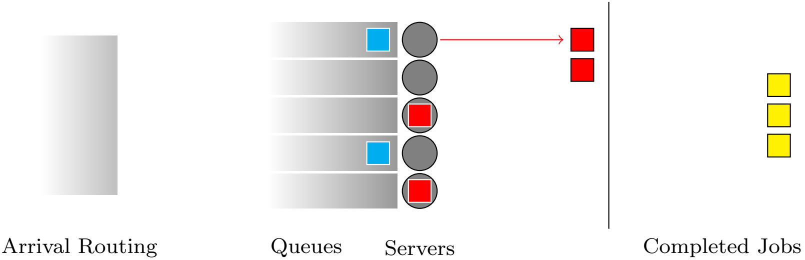
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

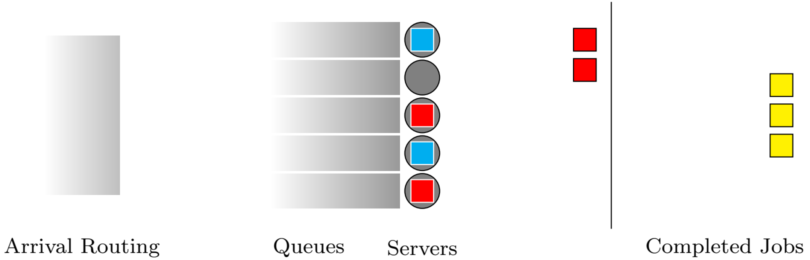
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

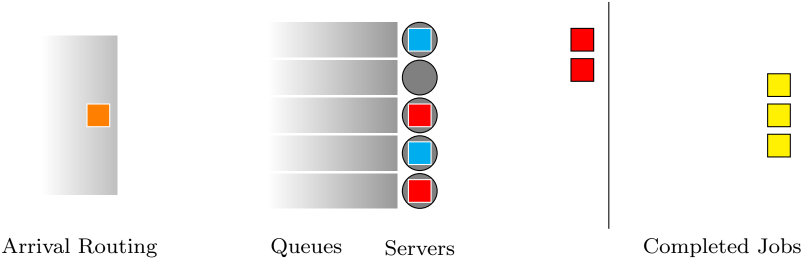
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

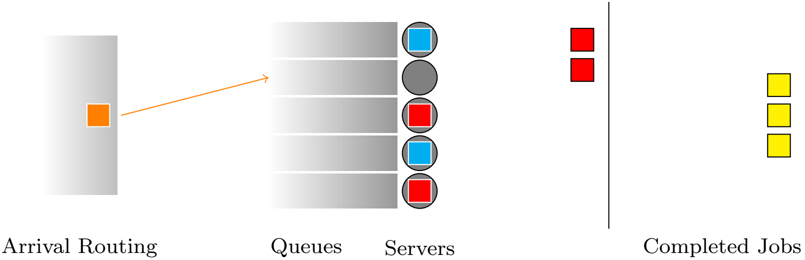
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

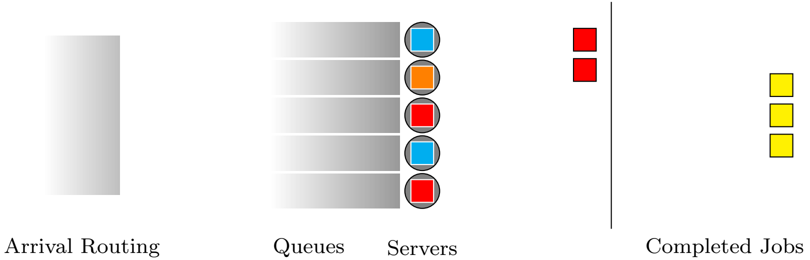
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

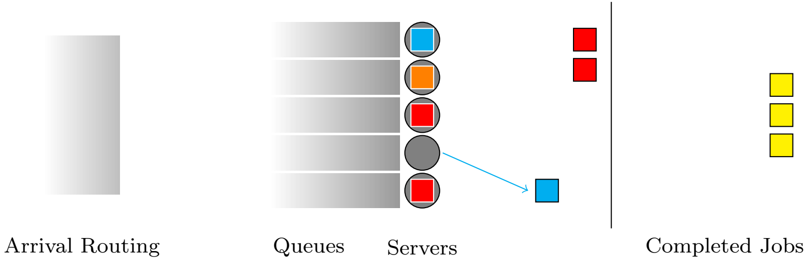
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

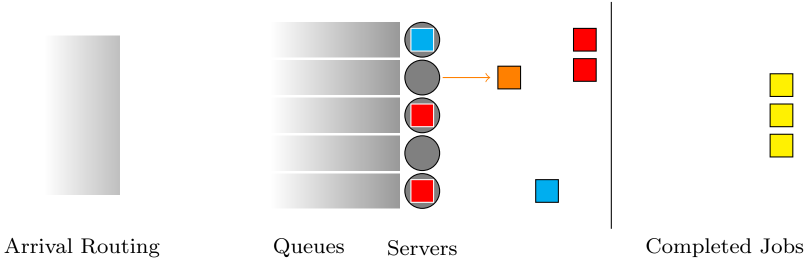
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

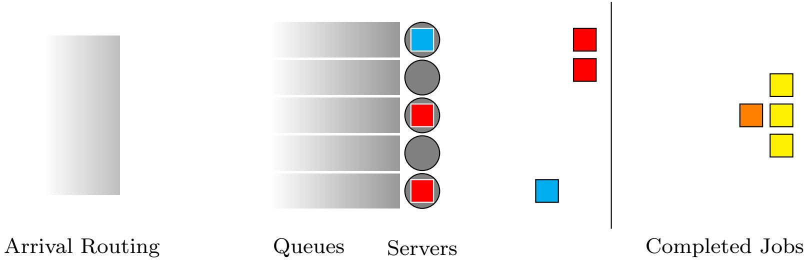
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

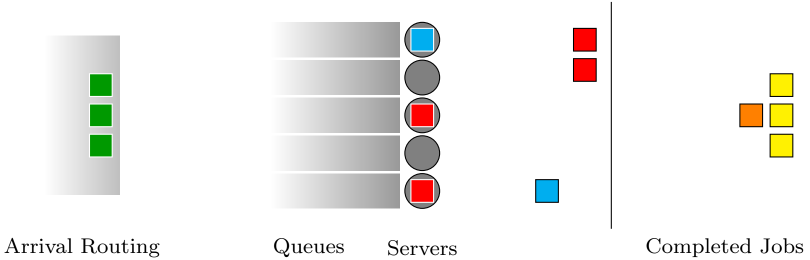
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

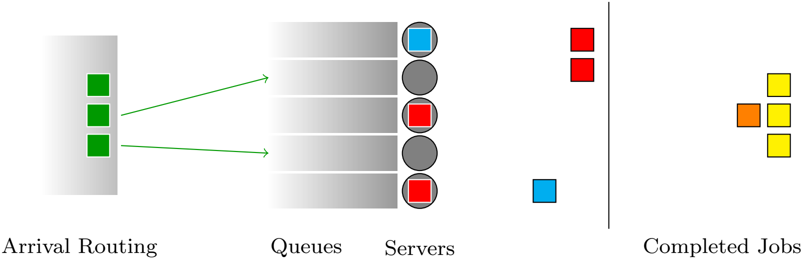
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

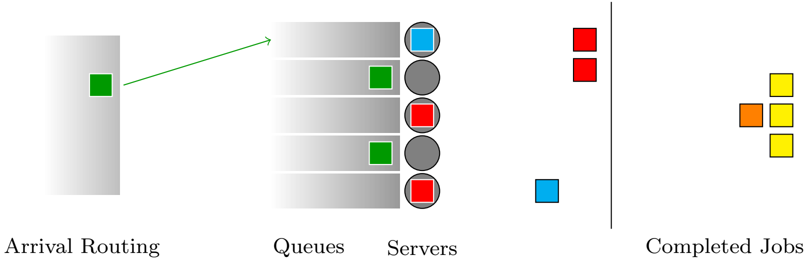
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

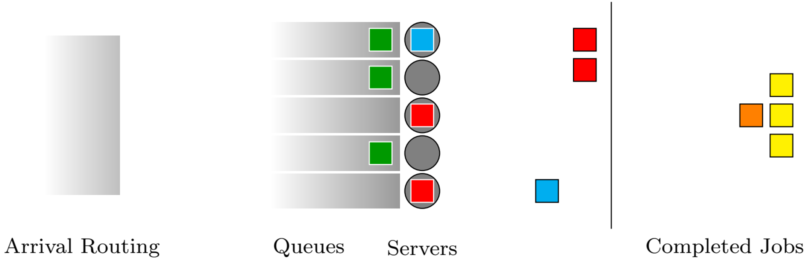
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

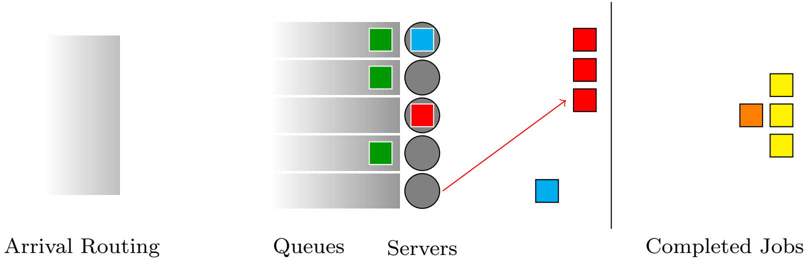
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

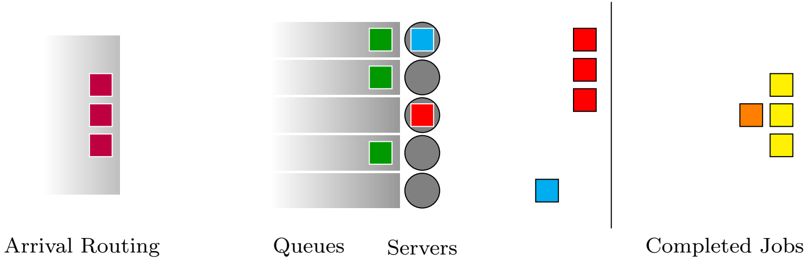
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

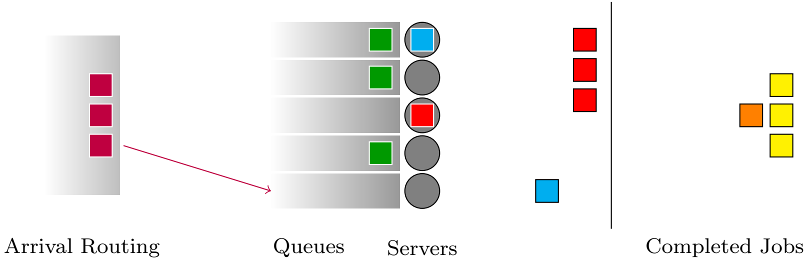
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

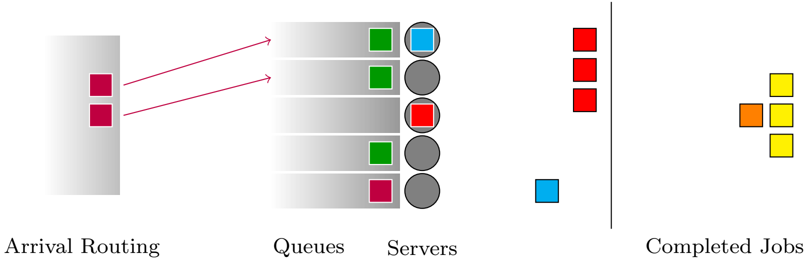
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

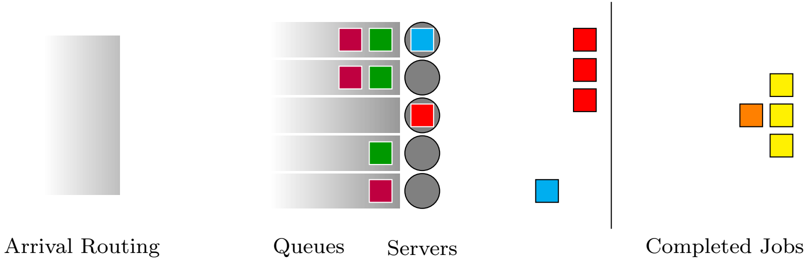
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

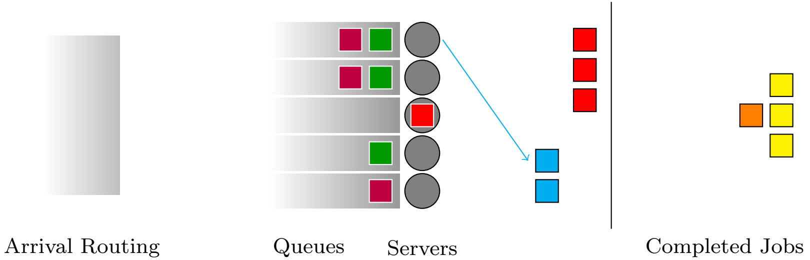
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

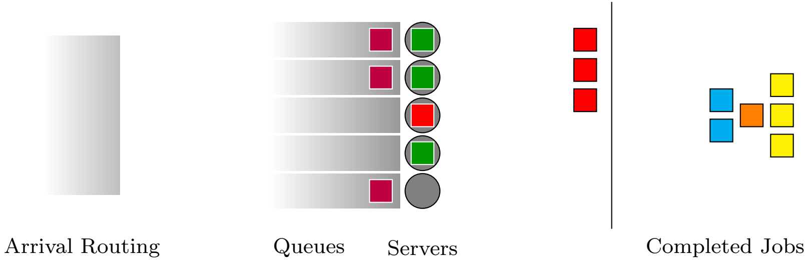
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

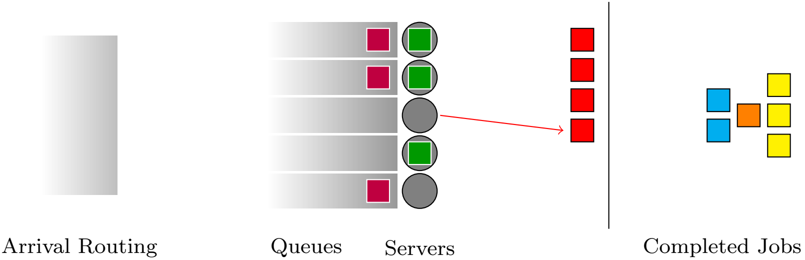
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

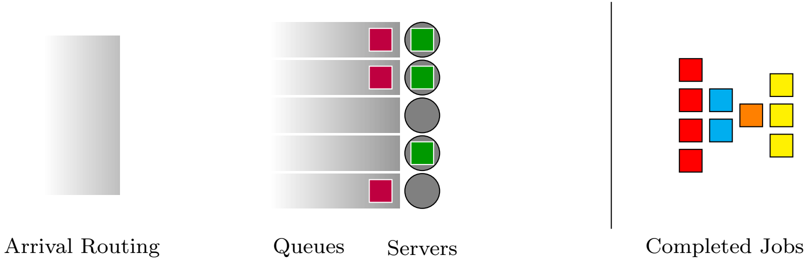
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

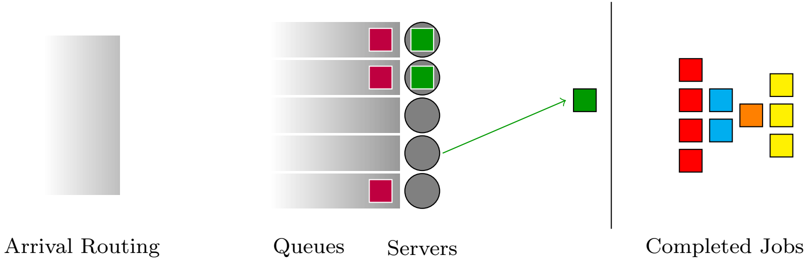
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

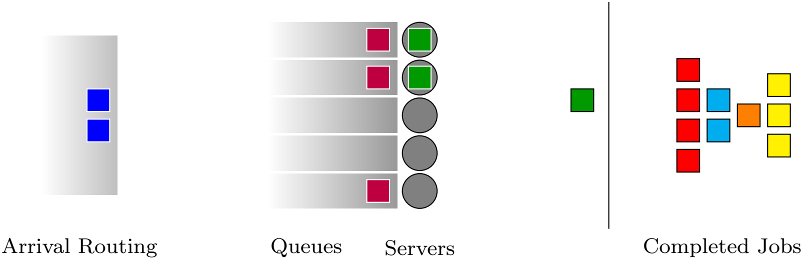
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

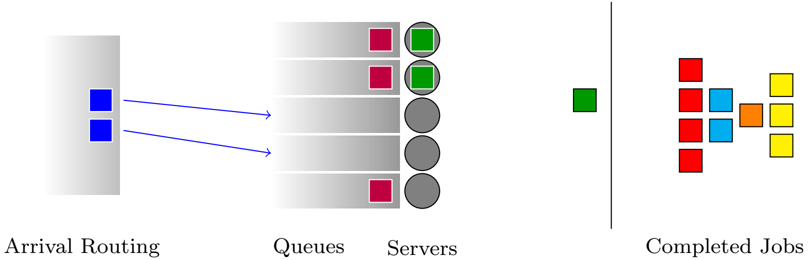
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

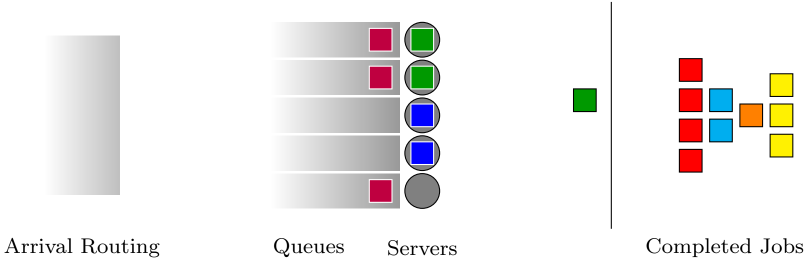
- First route pieces to idle servers, then randomly route the remaining pieces



Model: Intermediate Routing Scheme

We propose a mixed scheme.

- First route pieces to idle servers, then randomly route the remaining pieces



Plan

1 Motivation

2 Model

3 Preliminary Results

4 Numerical Results

5 Conclusions

6 References

Preliminary Results

Let f be a distribution of the number of pieces such that $f_n \overset{n}{\Rightarrow} f$, with finite mean $\mathsf{E}[N] = \sum_{k=1}^{\infty} k f(k)$.

Preliminary Results

Let f be a distribution of the number of pieces such that $f_n \overset{n}{\Rightarrow} f$, with finite mean $\mathsf{E}\left[N\right] = \sum_{k=1}^{\infty} k f(k)$.

- Green [G80]: if $\lambda \mathsf{E}\left[N\right]/\mu < 1$, then $W^{(n)} \overset{n}{\Rightarrow} 0$.

Preliminary Results

Let f be a distribution of the number of pieces such that $f_n \overset{n}{\Rightarrow} f$, with finite mean $\mathsf{E}[N] = \sum_{k=1}^{\infty} k f(k)$.

- Green [G80]: if $\lambda \mathsf{E}[N]/\mu < 1$, then $W^{(n)} \overset{n}{\Rightarrow} 0$.
- Olvera-Cravioto & Ruiz-Lacedelli [OR14]: $W^{(n)} \overset{n}{\Rightarrow} W$, if there exists $\beta > 0$ such that

$$\mathsf{E}\left[\sum_{i=1}^{N} e^{\beta(\chi_i - \tau_i)}\right] = \frac{\lambda \mathsf{E}[N]^2}{\beta + \lambda \mathsf{E}[N]} \mathsf{E}\left[e^{\beta \chi_1}\right] < 1,$$

where $N \sim f$, $\chi_i \sim S$ and $\tau_i \sim \mathrm{Exp}(\lambda \mathsf{E}[N])$ are independent.

Plan

1. Motivation
2. Model
3. Preliminary Results
4. Numerical Results
5. Conclusions
6. References

Numerical Results

We consider:

Numerical Results

We consider:

- 50 – 1,000 servers

Numerical Results

We consider:

- 50 – 1,000 servers
- 30,000 jobs

Numerical Results

We consider:

- 50 – 1,000 servers
- 30,000 jobs
- Distribution of the number of pieces:
  - $N \sim$ Poisson (light tail)
  - $N \sim$ Poisson composed with Pareto (heavy tail)

Numerical Results

We consider:

- 50 – 1,000 servers
- 30,000 jobs
- Distribution of the number of pieces:
  - $N \sim$ Poisson (light tail)
  - $N \sim$ Poisson composed with Pareto (heavy tail)
- Service time of the pieces:
  - $S \sim \text{Unif}(0, 1)$
  - $S \sim \text{Exp}(1)$

Numerical Results: Example 1

Parameters:

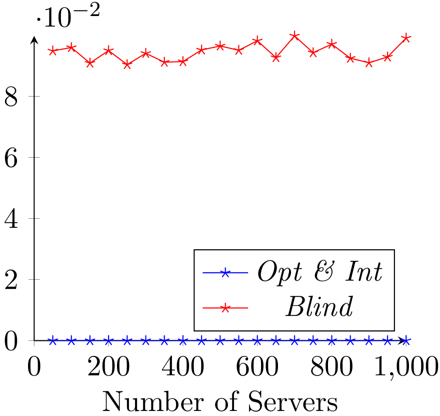
$$\lambda = 0.1 \qquad S \sim \text{Unif}(0,1) \qquad \mathsf{E}[N] = 2 \text{ (light tail)}$$

Numerical Results: Example 1

Parameters:

$$\lambda = 0.1 \qquad S \sim \text{Unif}(0,1) \qquad \mathsf{E}[N] = 2 \text{ (light tail)}$$

Average Waiting Time:



Numerical Results: Example 2

Parameters:

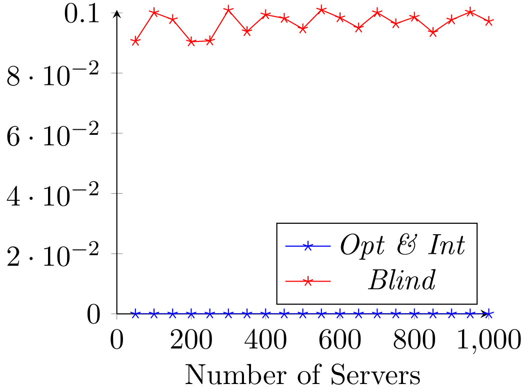
$$\lambda = 0.1 \qquad S \sim \text{Unif}(0,1) \qquad \mathsf{E}\left[N\right] = 2 \text{ (heavy tail)}$$

Numerical Results: Example 2

Parameters:

$$\lambda = 0.1 \qquad S \sim \text{Unif}(0,1) \qquad \mathsf{E}[N] = 2 \text{ (heavy tail)}$$

Average Waiting Time:



Numerical Results: Example 3

Parameters:

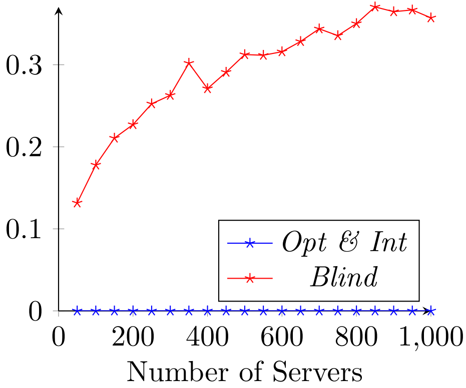
$$\lambda = 0.1 \qquad S \sim \text{Unif}(0,1) \qquad \mathsf{E}\left[N\right] = 20 \text{ (heavy tail)}$$

Numerical Results: Example 3

Parameters:

$$\lambda = 0.1 \qquad S \sim \text{Unif}(0,1) \qquad \mathsf{E}\left[N\right] = 20 \text{ (heavy tail)}$$

Average Waiting Time:



Numerical Results: Example 4

Parameters:

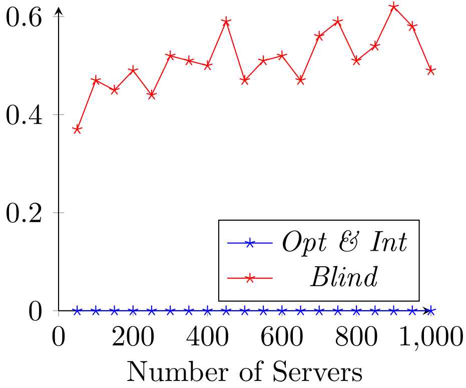
$$\lambda = 0.015 \qquad S \sim \text{Exp}(1) \qquad \mathsf{E}\left[N\right] = 4 \text{ (light tail)}$$

Numerical Results: Example 4

Parameters:

$$\lambda = 0.015 \qquad S \sim \text{Exp}(1) \qquad \mathsf{E}[N] = 4 \text{ (light tail)}$$

Average Waiting Time:



Numerical Results: Example 5

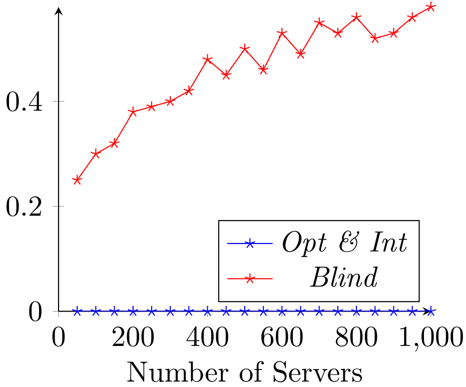
Parameters:

$$\lambda = 0.0025 \qquad S \sim \text{Exp}(1) \qquad \mathsf{E}[N] = 10 \text{ (heavy tail)}$$

Parameters:

$$\lambda = 0.0025 \qquad S \sim \text{Exp}(1) \qquad \mathsf{E}[N] = 10 \text{ (heavy tail)}$$

Average Waiting Time:



Plan

1. Motivation
2. Model
3. Preliminary Results
4. Numerical Results
5. Conclusions
6. References

Conclusions

According to our numerical results:

Conclusions

According to our numerical results:

- Light and heavy tails behave similarly

Conclusions

According to our numerical results:

- Light and heavy tails behave similarly
- Only knowing which servers are empty, decreases the waiting time significantly

Plan

1. Motivation
2. Model
3. Preliminary Results
4. Numerical Results
5. Conclusions
6. References

References

[G80] Green, Linda. "A queueing system in which customers require a random number of servers." Operations Research 28, no. 6 (1980): 1335-1346.

[OR14] Olvera-Cravioto, Mariana & Ruiz-Lacedelli, Octavio. "Parallel queues with synchronization." (2014) arXiv:1501.00186.

[S12] Stolyar, Alexander. "An infinite server system with general packing constraints." (2012) arXiv:1205.4271.