Dynamic Control of Parallel-Server Systems

Jim Dai
Georgia Institute of Technology

Tolga Tezcan
University of Illinois at Urbana-Champaign

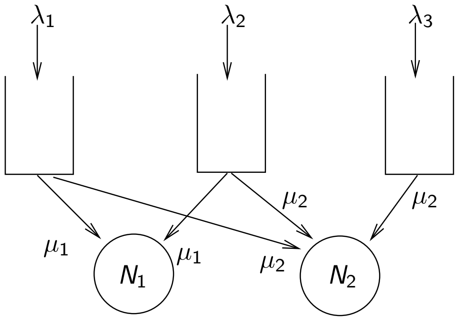
May 13, 2009

Outline

- Parallel-server systems
- Part I: Background
- Part II: Dynamic control

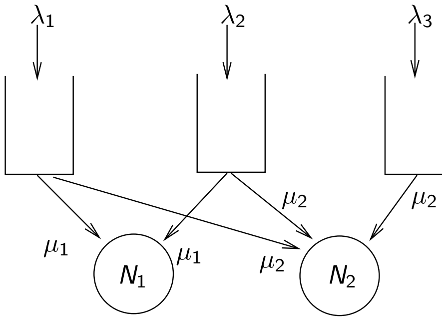
- Tezcan-Dai (2009), Dynamic Control of N-Systems with Many Servers: Asymptotic Optimality of a Static Priority Policy in Heavy Traffic, *Operations Research*.
- Dai-Tezcan (2008), Optimal Control of Parallel Server Systems with Many Servers in Heavy Traffic, *Queueing Systems*.

Parallel-Server Systems with Many Servers



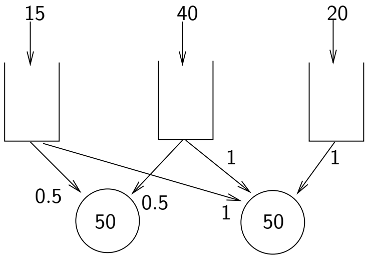
- I customer classes: arrival rate for class $i \in \mathcal{I}$ is λ_i.
- J server pools: pool $j \in \mathcal{J}$ has N_j servers.

Parallel-Server Systems with Many Servers



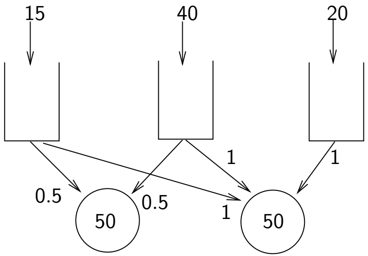
- I customer classes: arrival rate for class $i \in \mathcal{I}$ is λ_i.
- J server pools: pool $j \in \mathcal{J}$ has N_j servers.
- Large number of servers; motivated by customer call/contact centers.

Decisions



- Design: should agents be cross-trained?
- Staffing: long term and short term
- Routing
  - When an arrival finds idle servers, which server to join?
  - When a server finishes service, which customer to serve next?

Decisions



- Design: should agents be cross-trained?
- Staffing: long term and short term
- Routing
  - When an arrival finds idle servers, which server to join?
  - When a server finishes service, which customer to serve next?
- These decisions are made at different time scales.
  In this talk, we focus on routing decisions.

Part I: Background

- ED, QD, and QED regimes
- Square-root safety staffing rule
- Customer abandonment
- Distributions of random times

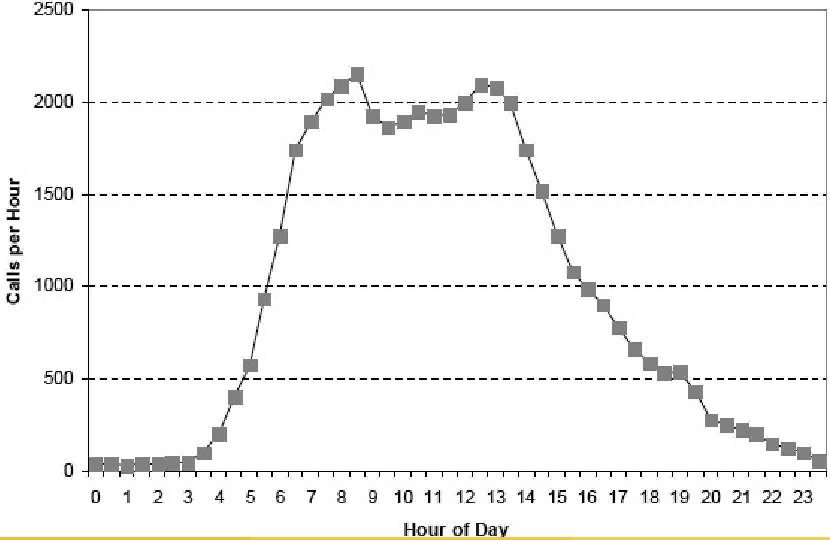
S. Zeltyn and A. Mandelbaum (2005), Call centers with impatient customers: many-server asymptotics of the $M/M/n+G$ queue, *Queueing Systems*, **51**.

Sample from a US health insurance company

Table 1
Example of half-hour ACD report.

Time	Calls	Answered	Abandoned%	ASA	AHT	Occ%	# of agents
Total	20,577	19,860	3.5	30	307	95.1	
8:00	332	308	7.2	27	302	87.1	59.3
8:30	653	615	5.8	58	293	96.1	104.1
9:00	866	796	8.1	63	308	97.1	140.4
9:30	1,152	1,138	1.2	28	303	90.8	211.1
10:00	1,330	1,286	3.3	22	307	98.4	223.1
10:30	1,364	1,338	1.9	33	296	99.0	222.5
11:00	1,380	1,280	7.2	34	306	98.2	222.0
11:30	1,272	1,247	2.0	44	298	94.6	218.0
12:00	1,179	1,177	0.2	1	306	91.6	218.3
12:30	1,174	1,160	1.2	10	302	95.5	203.8
13:00	1,018	999	1.9	9	314	95.4	182.9
13:30	**1,061**	**961**	**9.4**	**67**	**306**	**100.0**	**163.4**
14:00	1,173	1,082	7.8	78	313	99.5	188.9
14:30	**1,212**	**1,179**	**2.7**	**23**	**304**	**96.6**	**206.1**
15:00	1,137	1,122	1.3	15	320	96.9	205.8
15:30	1,169	1,137	2.7	17	311	97.1	202.2
16:00	1,107	1,059	4.3	46	315	99.2	187.1
16:30	914	892	2.4	22	307	95.2	160.0
17:00	**615**	**615**	**0.0**	**2**	**328**	**83.0**	**135.0**
17:30	420	420	0.0	0	328	73.8	103.5
18:00	49	49	0.0	14	180	84.2	5.8

Time-Varying Arrival Rate (Green, Kolesar and Soares)



Different Operational Regimes

- **13:30:** 100% occupancy, relately high abandonment rate (9.4%), more than 1 minute ASA; Efficiency-Driven (ED) regime.
- **17:00**, 83% server utilization, no abandonment, ASA less than 2 seconds; Quality-Driven QD regime.
- **14:30** 96.6% utilization, abandonment 2.7%, ASA 23 seconds; Quality- and Efficiency-Driven (QED) regime.

Staffing Rule?

Assume that $\mu = 1$. In the $M/M/n$ setting,

λ	n	util.	$\mathbb{P}\{$ delay $\}$
100	107	93.4%	38%
1000	1021	97.9%	40%
5000	5047	99.0%	39.4%

- $R = \lambda/\mu$,
- Square-root safety-staffing rule:

$$n = \lceil R + \beta\sqrt{R} \rceil ?$$

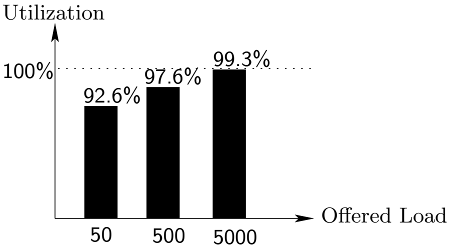
- Any relationship between α and β?
- $\alpha = 40\%$, $\beta = 0.65$. $1000 + .65\sqrt{1000} = 1020.6$.

QED Theorem (Halfin-Whitt, 1981)

- Consider a sequence of $M/M/n$ models, $n = 1, 2, 3, \ldots$
- Then the following **3 points of view** are equivalent:
  - Customer:
  $$\lim_{n\to\infty} \mathbb{P}\{\text{Wait} > 0\} = \alpha, \quad 0 < \alpha < 1;$$
  - Server: $\rho_n = \lambda_n/(n\mu)$
  $$\lim_{n\to\infty} \sqrt{n}(1 - \rho_n) = \beta, \quad 0 < \beta < \infty;$$
  - Manager:
  $$n \approx R + \beta\sqrt{R}, \quad \text{when } R = \lambda \times \mathbb{E}(S) \quad \text{large;}$$
- Here,
$$\alpha = [1 + \beta\Phi(\beta)/\phi(\beta)]^{-1}$$
and ϕ and Φ are the standard normal density and the distribution.

Square-Root Safety Staffing and QED

- Servers' utilization: $R/n \approx 1 - \frac{\beta}{\sqrt{n}}$
- For $\alpha = 0.5$, $\beta \approx 0.508$.
- Let $\mu = 1$, and $\lambda = 50, 500, 5000$.



Square-Root Safety-Staffing Rule

- Garnett, O., Mandelbaum, A. and Reiman, M. (2002). Designing a call center with impatient customers. *Manufacturing and Service Operations Management*, **48** 566–583.
- S. Borst, A. Mandelbaum, and M. Reiman, *Dimensioning large call centers*, Operations Research, 52 (2004), pp. 17–34.
- S. Halfin and W. Whitt, *Heavy-traffic limits for queues with many exponential servers*, Operations Research, 29 (1981), pp. 567–588.

Abandonment affects system performance: I

An example: 50 agents, 48 calls per minute, 1 minute average service time, 2 minute average patience;

	$M/M/n$	$M/M/n+M$
Fraction abandoning	0	3.1%
Average waiting time	20.8 sec.	3.6 sec.
Waiting time's 90th percentile	58.1 sec.	12.5 sec.
Average queue size	17	3
Agents' utilization	96%	93%

Abandonment affects system performance: II

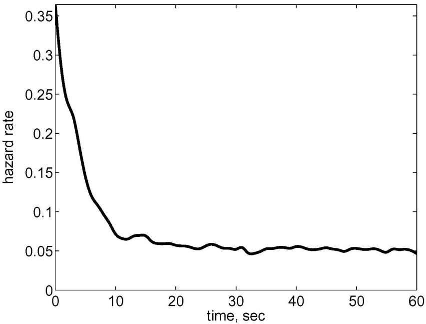
An example: 50 agents, 55 calls per minute, 1 minute average service time, 2 minute average patience;

	$M/M/n$	$M/M/n+M$
Fraction abandoning	0	10.2%
Average speed to answer	87.7 sec.	12.5 sec.
Average queue size	72.2	11.2
Agents' utilization	98.8%	98.8%

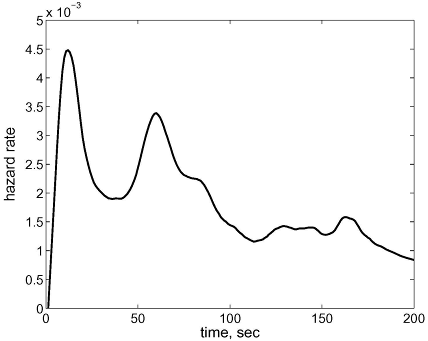
$\lambda^* = 55(1-0.102) = 49.39.$

Wrong model, wrong output!

Patience Time Distributions: Hazard rate



American bank



Israeli bank

- Does the distribution matter?

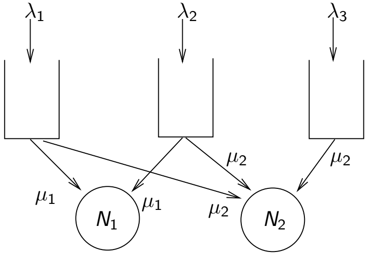
Some Insights

- In QED regime, the distribution of patience time “does not matters” with a given mean (M-Z 2005, Dai-He 2009), but one must build customer patience into the model.
- In QED regime, service time distribution matters (Reed, ...)
- In ED regime, the performance is mainly driven by the patience time distribution. (Whitt 2006)

Whitt's Study: $M/GI/100/200 + GI$

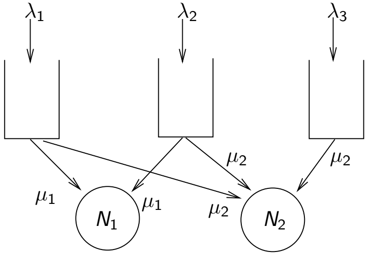
$M/GI/100/200+GI$ model with $\lambda=120$ and $E[T]=1.0$						
	E_2 time-to-abandon cdf Service cdf			LN(1, 4) time-to-abandon cdf Service cdf		
Perf. meas.	E_2	LN(1, 4)	Approx.	E_2	LN(1, 4)	Approx.
$P(A_s)$	0.16653 ±0.00035	0.16683 ±0.00060	0.16667 —	0.1678 ±0.00023	0.1696 ±0.00054	0.16667 —
$E[Q_s]$	40.25 ±0.057	39.56 ±0.097	41.11 —	14.51 ±0.018	14.52 ±0.043	14.63 —
$\mathrm{Var}(Q_s)$	139.6 ±0.69	221.6 ±1.09	0.00 —	61.1 ±0.18	81.5 ±0.30	0.00 —
$\mathrm{SCV}(Q_s)$	0.086	0.142	0.00	0.290	0.387	0.000
$E[N_s]$	140.3 ±0.057	139.5 ±1.22	141.11 —	114.4 ±0.019	114.2 ±0.47	114.6 —
$P(W_s=0)$	0.00046 ±0.00006	0.0068 ±0.00035	0.00000 —	0.032 ±0.00037	0.065 ±0.00077	0.000 —
$E[W_s \mid S_s]$	0.353 ±0.00051	0.343 ±0.00094	0.365 —	0.126 ±0.00017	0.125 ±0.00040	0.131 —
$\mathrm{Var}(W_s \mid S_s)$	0.0097 ±0.000058	0.0176 ±0.000087	0.0000 —	0.0046 ±0.000014	0.0066 ±0.000027	0.0000 —
$\mathrm{SCV}(W_s \mid S_s)$	0.078	0.149	0.000	0.290	0.422	0.000
$E[W_s \mid A_s]$	0.247 ±0.00025	0.261 ±0.00041	0.231 —	0.095 ±0.00008	0.103 ±0.00014	0.077 —

Part II: Back to the Parallel Server System



- Design: should agents be cross-trained?
- Staffing: long term and short term
- Routing
  - When an arrival finds idle servers, which server to join?
  - When a server finishes service, which customer to serve next?

Part II: Back to the Parallel Server System



- Design: should agents be cross-trained?
- Staffing: long term and short term
- Routing
  - When an arrival finds idle servers, which server to join?
  - When a server finishes service, which customer to serve next?
- These decisions are made at different time scales.
  In this talk, we focus on routing decisions.

Dynamic control

- A simple policy π^*
- An LP and the asymptotic framework
- State space collapse (SSC) and hydrodynamic models

Costs

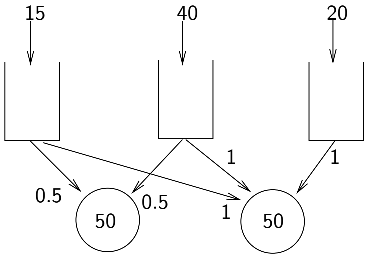
- holding cost h_i per class i waiting customer per unit time;
- penalty cost c_i per abandoned customer from class i
- class i has exponential patience time distribution with rate γ_i.
- the "total cost" per class i customer per unit time is

$$h_i + c_i\gamma_i.$$

Assume that buffer 1 is the cheapest:

$$h_1 + c_1\gamma_1 \le h_i + c_i\gamma_i \quad i \in \mathcal{I}.$$

Routing Policies: Design Objectives



- Simple
- “Robust” to deal with fluctuating λ: e.g., $\lambda = (20, 50, 10)$.
- Asymptotically optimal

A Dynamic Priority Policy π^*

- Let

$$X(t) = \sum_{i \in \mathcal{I}} Q_i(t) + \sum_{j \in \mathcal{J}} Z_j(t)$$

be the total number of customers in the system at time t.

- Let

$$\hat{X}(t) = X(t) - |N| = \sum_{i \in \mathcal{I}} Q_i(t) - \sum_{j \in \mathcal{J}} I_j(t).$$

Note that

$$x = a - b, \quad a, b > 0$$
$$a \geq x^+,$$
$$b \geq x^-,$$
$$a = x^+ \text{ only when } b = 0,$$
$$b = x^- \text{ only when } a = 0.$$

-
    - $\sum_{i \in \mathcal{I}} Q_i(t) \geq (\hat{X}(t))^+$,
    - $\sum_{j \in \mathcal{J}} I_j(t) \geq (\hat{X}(t))^-$,

The Routing Policy π^* for the Example

- When a server in the slow pool is ready to pick, choose

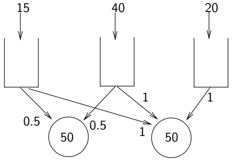
$$\operatorname{argmax}\{Q_1(t), Q_2(t)\}.$$

- When a server in the fast pool is ready to pick, choose

$$\operatorname{argmax}\{Q_1(t) - (\hat{X}(t))^+, Q_2(t), Q_3(t)\}.$$

- When an arriving customer is to choose a pool, choose

$$\operatorname{argmax}\{I_1(t) - (\hat{X}(t))^-, I_2(t)\}.$$



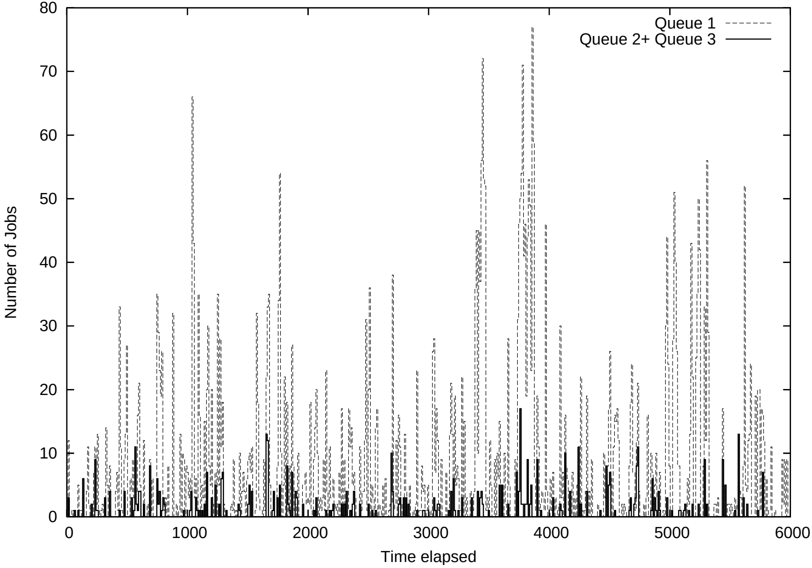
The Policy π^*

The following characteristics define the proposed policy π^*:

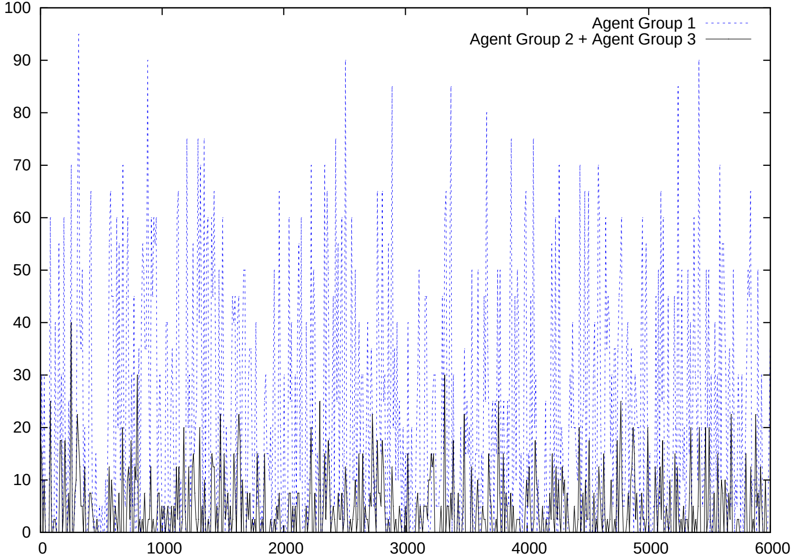
- each server is non-idling;
- a server chooses the leading customer in a buffer with the longest queue, where the queue length in buffer 1 is adjusted to be $Q_1(t) - \left(\hat{X}(t)\right)^+$,
- an arriving customer joins the server pool that has a maximum number of idle servers, except that the number of idle servers at the slowest pool, assumed to be pool 1, is adjusted to be $I_1(t) - \left(\hat{X}(t)\right)^-$.

Gurvich and Whitt (2009), Service-level differentiation in many-server service systems via queue-ratio routing, *OR*;
Queue-and-idleness-ratio controls in many-server service systems, *MOR*

SSC for Queue Length



SSC for Idle Server



Many-Server Heavy Traffic: capacity scales with volume

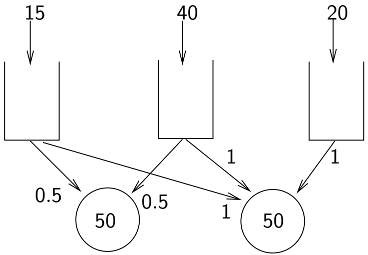
- We assume that the sequence of arrival rates to class i satisfies

$$\lim_{r\to\infty} \frac{\lambda_i^r}{|N^r|} = \lambda_i, \text{ for all } i \in \mathcal{I} \text{ and for some } 0 < \lambda_i < \infty. \qquad (1)$$

- Also, the sequence of number of servers in each pool is assumed to satisfy

$$\lim_{r\to\infty} \frac{N_j^r}{|N^r|} = \beta_j, \text{ for all } j \in \mathcal{J} \text{ and for some } \beta_j > 0 \text{ and} \qquad (2)$$

Asymptotic Framework: Example



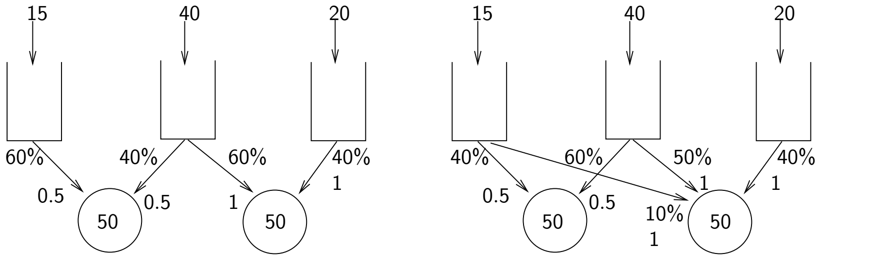
- $\beta = (1/2, 1/2)$,
- $\lambda = (.15, .4, .2)$

Static Planning LP

The static planning problem (SPP) is defined by

$$
\begin{aligned}
& \min \rho \\
& \text{s.t.} \\
& \sum_{j \in \mathcal{J}(i)} \beta_j \mu_{ij} x_{ij} = \lambda_i, \text{ for all } i \in \mathcal{I}, \\
& \sum_{i \in \mathcal{I}(j)} x_{ij} \leq \rho, \text{ for all } j \in \mathcal{J}, \\
& x_{ij} \geq 0, \text{ for all } j \in \mathcal{J} \text{ and } i \in \mathcal{I}.
\end{aligned} \tag{3}
$$

Multiple LP Solutions



Many-Server Heavy Traffic Condition

- Let (ρ^*, x^*) be an optimal solution to the SPP. We assume that

$$\rho^* = 1 \quad \text{and} \quad \sum_{i \in \mathcal{J}(j)} x^*_{ij} = 1 \ \text{ for all } j \in \mathcal{J}.$$

- for each class $i \in \mathcal{I}$

$$\lambda^r_i = \sum_{j \in \mathcal{J}(i)} \mu_j x^*_{ij} N^r_j + \theta_i \sqrt{|N^r|} \tag{4}$$

some θ_i.

Five Assumptions

- The many-server heavy traffic condition holds.
- Buffer 1 is the cheapest buffer, namely,

$$h_1 + c_1\gamma_1 \le h_i + c_i\gamma_i \text{ and } \gamma_1 \ge \gamma_i, \text{ for all } i \in \mathcal{I}. \quad (5)$$

- Service time and patience time distributions are exponential.
- Service rates are pool-dependent only, not class-dependent; we index the server pools in a way so that

$$\mu_1 \le \mu_j \text{ for all } j \in \mathcal{J}. \quad (6)$$

- An LP graph is connected.

Objective Function

- Let $Q_i^r(t)$ denote the number of class k customers in queue at time t;
- Let $R_i^r(t)$ denote the number of class k customers who have abandoned the system by time t.
- We define the diffusion scaling for these processes by

$$\hat{Q}_i^r(t) = \frac{Q_i^r(t)}{\sqrt{|N^r|}} \text{ and } \hat{R}_i^r(t) = \frac{R_i^r(t)}{\sqrt{|N^r|}} \text{ for } t \geq 0 \text{ and } i \in \mathcal{I}.$$

- For a fixed $T > 0$, the total cost in $[0, T]$ is

$$\zeta^r(T) = \sum_{i \in \mathcal{I}} \left(\int_0^T h_i \hat{Q}_i^r(s)\, ds + c_i \hat{R}_i^r(T) \right).$$

Asymptotic Optimality

Theorem

Assume the five assumptions, and an appropriate initial condition. Then, the total cost $\zeta^r(T)$ is asymptotically minimized as $r \to \infty$ in the following sense: for any $x > 0$,

$$\liminf_{r\to\infty} \mathbb{P}\{\zeta^{r,\pi}(T) > x\} \geq \liminf_{r\to\infty} \mathbb{P}\{\zeta^{r,\pi^*}(T) > x\}. \tag{7}$$

Proof Steps

- A lower bound
  - The lower bound proof is similar to the proof of Theorem 3.2 in Tezcan-Dai (2006).
- The bound is achieved under π^*
  - The policy π^* is asymptotically efficient; fluid model has a certain invariant state.
  - A certain state space collapse (SSC) result holds under diffusion scaling.

State Space Collapse under π^*

1. All buffers except buffer 1 are empty; namely, for $i \geq 2$,

$$\|\hat{Q}_i^r(t)\|_T \to 0 \quad \text{as } r \to \infty.$$

2. All pools are fully busy except pool 1; namely, for $j \geq 2$,

$$\|\hat{I}_j^r(t)\|_T \to 0 \quad \text{as } r \to \infty.$$

3. All waiting happens in buffer 1;

$$\|\hat{Q}_1^r(t) - \big(\hat{X}^r(t)\big)^+\| \to 0 \quad \text{as } r \to \infty.$$

4. All idling happens in pool 1;

$$\|\hat{I}_1^r(t) - \big(\hat{X}^r(t)\big)^-\| \to 0 \quad \text{as } r \to \infty.$$

Proving State Space Collapse

- Study a deterministic hydrodynamic model;
- Prove a state space collapse (SSC) result for the hydrodynamic model;
- Apply Dai-Tezcan (05):
  - SSC for a deterministic hydrodynamic model implies multiplicative SSC for the corresponding stochastic parallel server system;
  - Extend Bramson's framework from conventional heavy traffic to many-server heavy traffic.