Optimal Rate-Matrix Pruning For Heterogeneous Systems

Zhisheng Zhao
Georgia Tech
Atlanta, GA, USA

Debankur Mukherjee
Georgia Tech
Atlanta, GA, USA

ABSTRACT

We consider large-scale load balancing systems where processing time distribution of tasks depend on both task and server types. We analyze the system in the asymptotic regime where both the number of task and server types tend proportionally to infinity. In such heterogeneous setting, popular policies like Join Fastest Idle Queue (JFIQ), Join Fastest Shortest Queue (JFSQ) are known to perform poorly and they even shrink the stability region. Moreover, to the best of our knowledge, in this setup, finding a scalable policy with provable performance guarantee has been an open question prior to this work. In this paper, we propose and analyze two asymptotically delay-optimal dynamic load balancing policies: (a) one that efficiently reserves the processing capacity of each server for "good" tasks and route tasks under the Join Idle Queue policy; and (b) a speed-priority policy that increases the probability of servers processing tasks at a high speed. Leveraging a framework inspired by the graphon literature and using the mean-field method and stochastic coupling arguments, we prove that both policies above achieve asymptotic zero queueing, whereby the probability that a typical task is assigned to an idle server tends to 1 as the system scales.

Keywords

heterogeneous load balancing, data locality, Join-the-Idle Queue, fluid method, stochastic coupling

1. INTRODUCTION

Advanced cloud computing platforms, such as AWS, Azure, and Google Cloud, handle millions of requests per second. Efficiently assigning tasks across servers using a load balancing algorithm is critical in such environments. While previous theoretical works have mostly focused on homogeneous load balancing models, where parallel servers process only one type of task at the same rate, real-world cloud computing platforms receive requests containing multiple classes of tasks with varying characteristics, such as accessing websites, training machine learning models, or backing up data. Additionally, with the expansion of these platforms, servers can be of different types (multi-skilled), as evident from AWS's website, which lists at least 9 server types with varying memory and bandwidth. Moreover, due to the storage capacity limitation at servers (a.k.a. *data locality*), a server can only have required resource files to execute only a (small) subset of tasks. Thus, it is natural to model such large-scale data center networks as heterogeneous parallel-server systems, where the time to process a task in a server depends on *both the type of the task and that of the server*. Even though our primary motivation is data center networks, it is worthwhile to mention that similar heterogeneity exists in many other service systems as well. For example, in hospitals patients arriving at the emergency room may have different types of emergencies and multiple medical staff available, or in manufacturing systems different types of machines and workers are present for different operations, such as assembly, packaging, and painting.

For such general heterogeneous setting, popular routing policies, like Join Shortest Queue (JSQ), Join Idle Queue (JIQ), Join Fastest Shortest Queue (JFSQ) and the Join Fastest Idle Queue (JFIQ) are known to perform poorly. One reason is that they prioritize servers with the shortest or idle queue and might assign tasks to servers that cannot process at a relatively high speed with high probability, leading to inefficient server utilization.

In the seminal work [3], Stolyar proposed the MinDrift policy, which can be understood as the Gcμ-rule ([2, Section 4]) in the (output-queued) load balancing setup. It has been shown that MinDrift asymptotically minimizes the server workload in the conventional heavy traffic regime. However, implementing the MinDrift policy requires the dispatcher to know the total expected workload and service rate of every compatible server for the new task, which could result in a prohibitive communication burden when dealing with a large number N servers.

Model description. Consider a heterogeneous parallel-server system denoted by $G^N = (\mathcal{W}^N, \mathcal{V}^N, \boldsymbol{\lambda}^N, \mathcal{U}^N)$. In this system, $\mathcal{W}^N = \{1, ..., W(N)\}$ represents the set of dispatchers, where each dispatcher $i \in \mathcal{W}^N$ can only handle one type of task. Hence, the terms 'task-type' and 'dispatcher' will be used interchangeably. $\mathcal{V}^N = \{1, ..., N\}$ denotes the set of servers, where each server $j \in \mathcal{V}^N$ has a dedicated queue with infinite buffer capacity, and tasks are scheduled using the FCFS policy. The arrival process of tasks at the dispatcher $i \in \mathcal{W}^N$ is a Poisson process with rate $\lambda_i^N \in \boldsymbol{\lambda}^N = (\lambda_1^N, ..., \lambda_{W(N)}^N)$, independently of other processes. $\mathcal{U}^N = (\mu_{i,j}^N, i \in \mathcal{W}^N, j \in \mathcal{V}^N) \in \mathbb{R}_+^{W(N)\times N}$ represents a matrix of service rates, where the service time of a type $i \in \mathcal{W}^N$ task at server $j \in \mathcal{V}^N$ is exponentially distributed with mean $1/\mu_{i,j}^N$, if $\mu_{i,j}^N > 0$. Otherwise (i.e., when

MAMA workshop

$\mu^N_{i,j} = 0$), by convention, the server j cannot process type i tasks. A server $j \in \mathcal{V}^N$ is considered '*compatible*' for type $i \in \mathcal{W}^N$ tasks if $\mu^N_{i,j} > 0$. It is assumed that tasks arriving at a dispatcher must be instantaneously and irrevocably assigned to one of the compatible servers. A schematic diagram of the system is shown in Figure 1

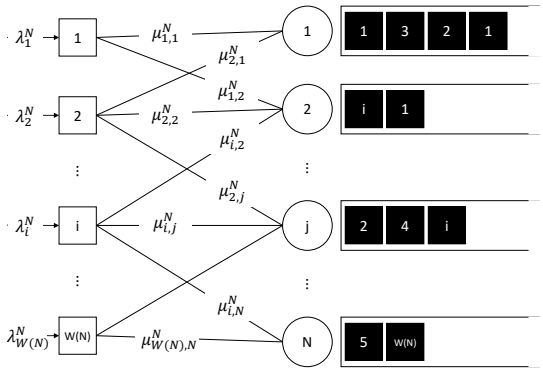

Figure 1: Heterogeneous Load Balancing System G^N

Our contribution. In this paper, we propose two scalable algorithms that achieves the highly desirable 'zero-queueing' property in general heterogeneous systems: (a) The first one using an appropriate rate-matrix pruning step before the system goes live, we reserve the capacity of each server for a subset of 'good' tasks, which are tasks that can be processed efficiently. After that, dispatchers assign tasks to servers according to the vanilla JIQ policy (that does not use processing rate information). (b) In the second one, each dispatcher clusters its compatible servers into several groups based on their service capability. For each new task, the dispatcher randomly chooses a target group for a new task, which gives more weight to the group with higher service rates, and then assigns the task to the shortest or idle queue within the target group.

2. MAIN RESULTS

We consider a sequence $\{G^N = (\mathcal{W}^N, \mathcal{V}^N, \boldsymbol{\lambda}^N, \mathcal{U}^N)\}_{N\in\mathbb{N}}$ of systems and analyze the asymptotic behavior under the JIQ-type policy. To have consistency in the above sequence, we assume that the sequence $\{G^N\}_{N\in\mathbb{N}}$ has a nested structure, that is, for all $N \in \mathbb{N}$, $\mathcal{W}^N \subseteq \mathcal{W}^{N+1}$, $\mathcal{V}^N \subseteq \mathcal{V}^{N+1}$, and $\mu^N_{i,j} = \mu^{N+1}_{i,j}$, $\forall (i,j) \in \mathcal{W}^N \times \mathcal{V}^N$. Inspired by the concept of Graphon [1, Chapter 7], we define two membership mapping functions $\phi_i : \mathbb{N} \to [0,1)$, $i = 1, 2$, for dispatchers and servers, respectively. Next, we model the heterogeneity of the processing rates using these mapping functions and a function $f : [0,1)^2 \to \mathbb{R}_+$ such that for $(i,j) \in \mathcal{W}^N \times \mathcal{V}^N$, $\mu^N_{i,j} = f(\phi_1(i), \phi_2(j))$. With the function f, we can model the service rates between dispatchers and servers for all systems in the sequence in a consistent way, instead of writing a matrix $\mathcal{U}^N$ whose dimension explodes as $N \to \infty$. Hence, we formally introduce what we call an 'f-sequence' and use it in the rest of the analysis.

Definition 2.1 (f-sequence). *Given a function $f : [0,1]^2 \to \mathbb{R}_+$. A sequence $\{G^N\}_N = (\mathcal{W}^N, \mathcal{V}^N, \mathcal{U}^N, \boldsymbol{\lambda}^N)$ is an f-sequence, if for each $N \in \mathbb{N}$, $u^N_{i,j} = f(\phi_1(i), \phi_2(j))$, $\forall (i,j) \in \mathcal{W}^N \times \mathcal{V}^N$.*

We will first analyze the case when f is a step-function and then use it to approximate the general system.

2.1 Special Case: Stepwise f

In this section, we consider an f-sequence $\{G^N\}_{N\in\mathbb{N}}$ with the stepwise function f defined as follows: for each $h \in [H]$ and $m \in [M]$,

$$f(x,y) = \mu_{h,m} \geq 0, \quad \forall (x,y) \in [w_{h-1}, w_h) \times [v_{m-1}, v_m), \tag{2.1}$$

where $0 = w_0 < w_1 < ... < w_H = 1$ and $0 = v_0 < v_1 < ... < v_M = 1$. By Definition 2.1, for each $h \in [H]$ and $m \in [M]$, and all $(i,j) \in \mathcal{W} \times \mathcal{V}$ such that $(\phi_1(i), \phi_2(j)) \in [w_{h-1}, w_h) \times [v_{m-1}, v_m)$, $\mu^N_{i,j} = f(\phi_1(i), \phi_2(j)) = \mu_{h,m}$, which implies that if $\phi_1(i_1)$ and $\phi_1(i_2)$ are both in $[w_{h-1}, w_h)$, the dispatchers i_1 and i_2 are indistinguishable since the rows $\boldsymbol{\mu}^N_{i_1} = (\mu^N_{i_1,1}, ..., \mu^N_{i_1,N})$ and $\boldsymbol{\mu}^N_{i_2} = (\mu^N_{i_2,1}, ..., \mu^N_{i_2,N})$ are the same. Hence, dispatchers can be classified into finite classes. Let $\mathcal{W}^N_h = \{i \in \mathcal{W}^N : \phi_1(i) \in [w_{h-1}, w_h)\}$ for each $h \in [H]$ with $\mathcal{W}^N = \cup_{h\in[H]} \mathcal{W}^N_h$. Similarly, let $\mathcal{V}^N_m = \{j \in \mathcal{V}^N : \phi_2(j) \in [v_{m-1}, v_m)\}$ for each $m \in [M]$ with $\mathcal{V}^N = \cup_{m\in[M]} \mathcal{V}^N_m$. For the asymptotic analysis and to avoid heavy traffic, we define the subcritical regime as follows.

Definition 2.2 (**p**-Subcritical). *The sequence of systems $\{G^N\}_{N\in\mathbb{N}}$ with stepwise f as in (2.1), is said to be in the subcritical regime if the followings are satisfied:*

(i) $\lim_{N\to\infty} \sum_{i\in\mathcal{W}^N_h} \lambda^N_i / N = \lambda_h > 0$, $\forall h \in [H]$;

(ii) $\lim_{N\to\infty} \sum_{j\in\mathcal{V}^N} \mathbb{1}_{(j\in\mathcal{V}^N_m)}/N = v_m - v_{m-1}$, $\forall m \in [M]$;

(iii) *There exists a stochastic matrix $\mathbf{p} = (p_{h,m}, h \in [H], m \in [M]) \in [0,1]^{H\times M}$ such that for all $m \in [M]$,*

$$\sum_{h\in[H]} \frac{\lambda_h p_{h,m}}{\mu_{h,m}(v_m - v_{m-1})} < 1. \tag{2.2}$$

Recall from earlier discussion that the main challenge in the current setup is the service rate depends on *both* dispatcher and server types. Interestingly, in the **p**-subcritical regime, for large enough N, we can (a) construct a subsystem $\tilde{G}^N$ that is a union of dispatcher-independent systems, where the service rate only depends on server-type, and (b), propose a speed-priority policy called **p**-based JIQ policy, under which the evolution of G^N can be coupled with a system $\hat{G}^N$ a union of server-independent systems, where the service rate only depends on dispatcher-type. These are the contents of Sections 2.1.1 and 2.1.2, respectively.

2.1.1 Union of dispatcher-independent systems

Denote $\varepsilon^*_m := (v_m - v_{m-1}) - \sum_{h\in[H]} \frac{\lambda_h p_{h,m}}{\mu_{h,m}}$, $m \in [M]$. Let $\boldsymbol{\varepsilon} := (\varepsilon_{h,m})_{h\in[H],m\in[M]}$. Define a polytope Poly(**p**) as follows: $\text{Poly}(\mathbf{p}) := \big\{\boldsymbol{\varepsilon} = (\varepsilon_{h,m}, h \in [H], m \in [M]) \in [0,1]^{H\times M} : \boldsymbol{\varepsilon} \text{ satisfies (2.3)}\big\}$

$$\begin{aligned} &\varepsilon_{h,1} : \cdots : \varepsilon_{h,M} = p_{h,1} : \cdots : p_{h,M}, \quad \forall h \in [H], \\ &\sum_{h\in[H]} \varepsilon_{h,m} \leq \varepsilon^*_m, \quad \forall m \in [M]. \end{aligned} \tag{2.3}$$

By the definition of the **p**-subcritical regime, it is easy to check that Poly(**p**) is non-empty. Consider any fixed feasible solution $\boldsymbol{\varepsilon} \in \text{Poly}(\mathbf{p})$. Using such an $\boldsymbol{\varepsilon}$, for each system G^N, we can construct a sub-system $\tilde{G}^N(\mathbf{p}, \boldsymbol{\varepsilon}) = (\mathcal{W}^N, \mathcal{V}^N, \boldsymbol{\lambda}^N, \tilde{\mathcal{U}}^N)$ as follows:

- Step 1: For each $\mathcal{V}_m^N$ in $\tilde{G}^N(\mathbf{p},\boldsymbol{\varepsilon})$, we divide it into $H+1$ separate sets $\{\tilde{\mathcal{V}}_{h,m}^N\}_{h\in[H]} \cup \tilde{\mathcal{V}}_{0,m}^N$ such that
  - $|\tilde{\mathcal{V}}_{h,m}^N| = \lfloor N(\frac{\lambda_h p_{h,m}}{\mu_{h,m}} + \varepsilon_{h,m})\rfloor$, $h \in [H]$
  - $|\tilde{\mathcal{V}}_{0,m}^N| = |\tilde{\mathcal{V}}_m^N| - \sum_{h\in[H]} |\tilde{\mathcal{V}}_{h,m}^N|$.
- Step 2: For each, $h \in [H]$, dispatchers in $\mathcal{W}_h^N$ is allowed only assign tasks to servers in $\cup_{m\in[M]}\tilde{\mathcal{V}}_{h,m}^N$. That is, we set $\tilde{\mu}_{i,j}^N = \mu_{i,j}^N = \mu_{h,m}$, for $i \in \mathcal{W}_h^N$ and server $j \in \cup_{m\in[M]}\tilde{\mathcal{V}}_{h.m}^N$, and set $\tilde{u}_{i,j}^N = 0$, otherwise.

Note that the constructed system $\tilde{G}^N(\mathbf{p},\boldsymbol{\varepsilon})$ can be viewed as the union of H separate dispatcher-independent (i.e., service rates depend only on server-types) systems: For $h \in [H]$, $\tilde{G}_h^N(\mathbf{p},\boldsymbol{\varepsilon})$ contains dispatchers $\mathcal{W}_h^N$ and servers $\cup_{m\in[M]}\tilde{\mathcal{V}}_{h,m}^N$. Also, for each $h \in [H]$, $\{\tilde{G}_h^N(\mathbf{p},\boldsymbol{\varepsilon})\}_N$ is in the subcritical regime as well. Thus, by [4, Theorem 2], we have the following theorem.

THEOREM 2.3. *Consider the stepwise f-sequence $\{G^N\}_N$ in $\mathbf{p}$-subcritical regime. For large enough N, we construct a subsystem $\tilde{G}^N$ of G^N as described above. Then, under the JIQ policy, in steady state, an arriving task will be assigned to an idle server with probability tending to 1 as $N \to \infty$.*

2.1.2 Union of server-independent systems

Recall the $\mathbf{p}$-subcritical regime with the stochastic matrix $\mathbf{p}$. We now introduce the $\mathbf{p}$-based JIQ policy as follows.

DEFINITION 2.4 ($\mathbf{p}$-BASED JIQ). *Consider a dispatcher $i \in \mathcal{W}_h^N$. When a task arrives at dispatcher i, it first selects a target server-type m^* with discrete distribution $\bar{p}_h = (p_{h,m})_{m\in[M]}$ and sends the new task to one of idle servers uniformly at random in $\mathcal{V}_{m^*}^N$, if any exist, and other wise to one of the servers in $\mathcal{V}_{m^*}^N$, chosen uniformly at random.*

Under the $\mathbf{p}$-based JIQ policy, note that each set $\mathcal{V}_m^N$ for $m \in [M]$, receives tasks from $\mathcal{W}_h^N$, $h \in [H]$ with rate $N\lambda_h p_{h,m}$. Also, servers in $\mathcal{V}_m^N$ are identical. By the Poisson thinning property, we can view the system G^N as the union of M server-independent systems $\{\hat{G}_m^N\}_{m\in[M]}$, i.e., where service rates depend only on dispatcher-types. For each $\hat{G}_m^N$, it consists of servers as the same as that in $\mathcal{V}_m^N$. Tasks that are served in $\hat{G}_m^N$ with rate μ_k will arrive at $\hat{G}_m^N$ as a Poisson process with rate $N\lambda_{m,k}^{\mathbf{P}}$. By the classical fluid method, we can establish the zero-queueing property of the policy:

THEOREM 2.5. *Consider the stepwise f-sequence $\{G^N\}_N$ in $\mathbf{p}$-subcritical regime. Under the $\mathbf{p}$-based JIQ policy, in steady state, tasks are assigned to idle servers with probability tending to 1 as $N \to \infty$.*

The key for implementing both approaches discussed above is to find the stochastic matrix $\mathbf{p}$ which can be done by solving the LP in (2.2). Also, both approaches can be implemented in a token-based fashion, inheriting scalability properties similar to the JIQ policy.

2.2 General Case

In this section, we will discuss how Theorems 2.3 and 2.5 can be extended to the general f case.

ASSUMPTION 2.6. (i) *(Arrival rate function) There exists an integrable function $\lambda : [0,1) \to \mathbb{R}_+$ with $\int_0^1 \lambda(x)dx = a > 0$ such that $\lambda(\phi_1(i)) = \lambda_i^N$, $\forall i \in \mathcal{W}^N, N \in \mathbb{N}$.*

(ii) *(Service rate function) The function f has finitely many discontinuity points on $[0,1)^2$, and there exists $\mu^o > 0$ such that for all $x \in [0,1)$, $|\{y \in [0,1) : f(x,y) \geq \mu^o\}| > 0$, where $|\cdot|$ is the Lebesgue measure.*

(iii) *(Regularity of membership map) For any subinterval $E \subseteq [0,1)$,*

$$\lim_{N\to\infty} \sum_{i\in\mathcal{W}^N} \frac{\mathbb{1}_{(\phi_1(i)\in E)}}{W(N)} = \lim_{N\to\infty} \sum_{j\in\mathcal{V}^N} \frac{\mathbb{1}_{(\phi_2(j)\in E)}}{N} = |E|.$$

(iv) $\lim_{N\to\infty} \frac{W(N)}{N} = \xi > 0$, *where ξ is a constant.*

Based on the above assumption, we define the subcritical regime for the general f-sequence as follows.

DEFINITION 2.7 ($(\mathbf{w},\mathbf{v},\mathbf{p})$-SUBCRITICAL REGIME). *The f-sequence $\{G^N\}_N$ is in the subcritical regime if the following is satisfied: There exist a pair of partitions $(\mathbf{w},\mathbf{v}) = (0 = w_0 < w_1 < \cdots < w_H = 1, 0 = v_0 < v_1 < \cdots < v_M = 1)$ of $[0,1]$ and a stochastic matrix $\mathbf{p} \in [0,1]^{H\times M}$ such that*

$$\rho_m(\mathbf{w},\mathbf{v},\mathbf{p}) \coloneqq \sum_{h\in[H]} \frac{p_{h,m}\lambda_h}{(v_m - v_{m-1})\mu_{h,m}^*} < 1, \quad m \in [M], \tag{2.4}$$

where, for each $h \in [H]$ and $m \in [M]$, $\lambda_h = \frac{1}{\xi}\int_{w_{h-1}}^{w_h} \lambda(x)dx$ and $\mu_{h,m}^* = \min_{(x,y)\in[w_{h-1},w_h)\times[v_{m-1},v_m)} f(x,y)$.

With the tuple $(\mathbf{w},\mathbf{v},\mathbf{p})$, we can construct a stepwise f'-sequence $\{G'^N\}_N$ as the following:

- Each system G'^N has the same dispatcher set and server set as that of the system G^N.
- For any $(x,y) \in [w_{h-1},w_h) \times [v_{m-1},v_m)$, $f'(x,y) = \min_{(a,b)\in[w_{h-1},w_h)\times[v_{m-1},v_m)} f(x,y)$.

It is not hard to check that the f'-sequence $\{G'^N\}_N$ is also in the $(\mathbf{w},\mathbf{v},\mathbf{p})$-subcritical regime. Since by the definition, $f'(x,y) \leq f(x,y)$ for all $(x,y) \in [0,1]^2$, it is also intuitive that the system G^N will have a better performance than the system G'^N in terms of queue length. Moreover, based on $(\mathbf{w},\mathbf{v},\mathbf{p})$, we can construct the sequence $\{\tilde{G}^N\}_N$ and design the $\mathbf{p}$-based JIQ policy as we did in Section 2.1. Hence, Theorems 2.3 and 2.5 will hold for the general f-sequence in the subcritical regime as well.

3. ACKNOWLEDGEMENTS

The work was supported by the NSF grant CIF-2113027.

4. REFERENCES

[1] L. Lovász. *Large Networks and Graph Limits.* Colloquium Publications, 2012.

[2] A. Mandelbaum and A. L. Stolyar. Scheduling flexible servers with convex delay costs: Heavy-traffic optimality of the generalized cμ-rule. *Oper. Res.*, 52(6):836–855, 2004.

[3] A. L. Stolyar. Optimal Routing in Output-Queued Flexible Server Systems. *Probability in the Engineering and Informational Sciences*, 19:141–189, 2005.

[4] A. L. Stolyar. Pull-based load distribution in large-scale heterogeneous service systems. *Queueing Syst.*, 80(4):341–361, 2015.