Allocation of Flows in Closed Bipartite Queueing Networks[☆]

James D. Brooks¹, Koushik Kar^{2,*}, D. Mendonça³

Abstract

This paper describes a novel method for allocating agents to routes in a closed bipartite queueing network to maximize system throughput using three open network approximations. Results are presented which compare this method with known prior work and optimal solutions to provide an empirical optimality gap. Average empirical optimality gaps of 1.29%, 1.13% and 1.29% are observed for the three approximations considered. Further, because many systems are under the control of rational agents, conditions are derived in order to determine properties of the market context that induce optimal behavior. It is shown that uniform rewards do not yield an efficient rational equilibrium in general. However, for systems with homogeneous servers and travel times or those with travel times that are much larger than queue waiting times, uniform rewarding is optimal.

Keywords: (I) Networks, (O) OR in disaster relief, Queueing systems, Routing control.

1. Introduction

Optimization of queueing systems has a long history [1]. Kleinrock provides the following general categories of optimization of queueing systems: capacity assignment, flow assignment, and topology [2]. Bell and Stidham further distinguish between dynamic (control) strategies and design (or static) strategies [3]. Dynamic strategies are functions of the current system state, while design strategies are state-independent and put in place prior to the operation of the system. For this

[☆]This is an extended version of a paper that appeared in the *2013 IEEE International Conference on Technologies for Homeland Security*, Waltham, MA in November 2013, with the title “Dynamic allocation of entities in closed queueing networks: An application to debris removal”.

*Corresponding author

¹J. D. Brooks is with G.E. Global Research Center, Niskayuna, NY, USA. This work was done when he was affiliated with the Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, 12180 USA. E-mail: brooksja@ge.com.

²K. Kar is with the Department of Electrical, Systems, and Computer Engineering, Rensselaer Polytechnic Institute, Troy, NY, 12180 USA. E-mail: kark@rpi.edu.

³D. Mendonça is with the Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, 12180 USA. E-mail: mendod@rpi.edu.

1
2
3
4 paper, static flow assignment (i.e., routing without knowledge of current system state) is the primary
5 focus, particularly as it relates to designing a market context for systems under the control of
6 rational dispatchers. Throughout this paper (as well as in the majority of the relevant literature),
7 optimization refers generally to maximizing the throughput of a system.

8
9
10
11 The motivating application for this work is the removal of debris from a region affected by
12 a disaster event. This debris removal mission involves a set of hauling vehicles that cyclically
13 deliver debris from a potentially large number of pickup sites (typically at curbsides) to a much
14 fewer number of temporary debris storage and reduction (TDSR) sites, forming a bipartite network.
15 Improvement in the throughput of this system (i.e., the average number of loads delivered per hour)
16 translates directly to the ability of a community to return to normal operation after the disaster
17 event. In order to accomplish this task, contractors provide teams of people and equipment to load
18 and haul debris from many pickup sites to a few central processing sites. Each of these sites operates
19 as a first-come-first-served queue and the number of vehicles is approximately fixed. There is no
20 method presently in place for regulating or prescribing the allocation of the vehicles circulating in
21 the different routes (cycles) of the system. Rather, they are assigned by a contractor in such a way
22 as to maximize their profit.

23
24
25
26 Due to the geographic extent of the system, a routing (allocation) policy which does not require
27 full knowledge of the location of all agents is desired (i.e. a state-independent policy). While some
28 limited prior work has considered optimal state-independent routing probabilities in closed bipartite
29 queueing networks, no work is known to provide a method of partitioning the agents into *groups*,
30 each with its own deterministic route, as is known to be optimal [4]. This work first presents
31 a novel method for generating optimal routing probabilities in these networks using reasonable
32 modeling approximations. It then presents a method for determining an entity partition, each with
33 its own deterministic routing, from these probabilities. Methods for dealing with some fixed (given)
34 probabilistic routing in the system are also discussed.

35
36
37
38 The next section (Section 2) describes related prior work in network routing and motivates the
39 approach used. The formulation for the general allocation problem in closed queueing networks
40 is then discussed in detail (Section 3), using a small example to illustrate the derivation before
41 presenting the general result. The partition algorithm is then described before numerical results
42 are presented. These results compare the new method with both existing methods and provably
43 optimal allocations to obtain an estimate of the empirical optimality gap. Finally, numerical results
44 are presented and discussed (Section 4), followed by some conclusions and directions for future work

(Section 5).

2. Literature Review

Prior research in flow routing of agents in general queueing networks spans a variety of fields, and may be classified according to the network topology of interest: parallel server system, open network, or closed network. Both centralized and decentralized routing strategies have been developed. Centralized approaches assume that the behavior of agents in the network can be fully controlled, while decentralized approaches allow that agents may be autonomous. The latter leads directly into control via market influence or other decentralized means. Both types of approaches are discussed in the remainder of this section.

2.1. Centralized Routing

Prior work in centralized routing has largely focused on the *parallel server* topology, in which arriving external agents choose from among a number of parallel servers and then, once serviced, exit the system. The objective is to minimize the expected delay (or equivalently the expected total number of agents in the system). Both dynamic (control) and static (design) policies have been explored. The join-shortest-queue (JSQ) policy is optimal for identical exponential servers [33] and asymptotically optimal for the multi-server case [24]. Other policies explored in this system topology include minimum expected delay, an extension of JSQ for heterogeneous servers [31], and round robin [32]. The throughput benefit of dynamic routing (allocation) in closed networks has been shown to decrease with increasing number of agents in the system [29].

Broadly speaking, faster servers are preferred in the optimal routing policy. If the parallel system has only two heterogeneous servers, it is optimal to send all traffic to the faster server until the number of agents waiting exceeds a threshold, at which point the slower servers should be utilized [25, 23]. Under heavy traffic, these state-dependent threshold policies converge to a static threshold policy [30]. For multiple parallel servers, faster servers are, disproportionately, more highly utilized [26, 29]. In fact, when multiple servers are available to an incoming agent, the faster-server-first policy is known to be asymptotically optimal while allowing slower servers to idle [22].

Flow optimization in *open networks* has been studied using a nonlinear programming framework [15] [13], a Markov decision process framework [14], and a Brownian heavy-traffic approximation [16]. A two-stage approach to optimizing multicommodity flows in generalized networks using

network approximations was developed in [17]. Whittle describes a method of maximizing the saturation throughput of an open network by selecting routing probabilities [15]. Optimal flows are not unique in general, though the relative server utilizations are so [13]. Combining dynamic routing and sequencing of a network comprising of multiple agent classes (each with their own dedicated queues at each server [16]), a pathwise solution outperforms optimal stochastic routing.

Throughput-maximizing allocation in *closed networks* has been explored [7] using a steepest-ascent method, in which the throughput of the system is given by the ratio of the network normalizing constant for $N - 1$ and N agents. The method relies on the fact that the throughput function is (pseudo)concave in workload. The result is an optimal probabilistic routing matrix. These results have recently been extended to the multi-server case [8]. A major drawback of these iterative methods (using Mean Value Analysis, or MVA) is their inability to produce optimality conditions which can meaningfully inform policy decisions.

To the best of our knowledge, there is no prior work that describes methods of partitioning a fixed population of agents to deterministic routes in a closed queueing network in order to maximize the system throughput. This is somewhat surprising given that the optimal routing strategy in product-form networks have been characterized [4, 6, 9].

2.2. Decentralized Routing

Much research on queueing systems assumes that the behavior of agents flowing through the system is deterministic or nearly so. These agents operate with a predefined utility function and assumed deterministic behavior based on the value of this function. The relevant bodies of literature are rational behavior in queueing systems [34] and network congestion games [35].

Wardrop equilibrium is an equilibrium notion often used in the context of network routing and traffic flow [37]. A general network is composed with delay functions on each arc which are a function of the flow (traffic) on the arc. This type of equilibrium has been explored with linear, nonlinear, and queueing delay functions [12, 19]. A network is said to be in this kind of equilibrium when all possible routes in use for each origin-destination pair have equal delay (i.e., no individual can benefit by unilaterally changing between routes in use). Furthermore, for routes between the same origin-destination pair which have no traffic (i.e., those that are unused for this origin-destination pair in equilibrium), the delay for a single vehicle must be greater than the delay of the routes in use. Finding such an equilibrium can be non-trivial. Indeed, an entire body of literature has developed around this problem (the traffic assignment problem [38]). Wardrop equilibrium is an

instance of Nash equilibrium where the number of agents is infinitely large (i.e., in *non-atomic* games) [38]. It is also less restrictive in the kinds of cost functions allowed. For this work, this equilibrium concept is extended to closed networks which have parallel cyclic paths:

Definition 1. *A closed network is said to be in Wardrop-like equilibrium when all used cycles have the same mean loop delay and this common delay exceeds that of any unused cycles.*

It is expected that for cases where a uniform payment policy is in effect, an economic, rational dispatcher will try to attain a Wardrop-like equilibrium. However, the payment policy is not uniform in practice. Because these prior definitions assume that the net profit gained by making the trip along any path (or cycle) is identical, the more complete notion of *economic Wardrop-like equilibrium* is introduced.

Definition 2. *A closed network is said to be in economic Wardrop-like equilibrium when all used cycles have the same expected per-loop profit and this common profit exceeds that of any unused cycles.*

It is well known that rational agents over-congest queueing systems compared to a socially-optimal operating point [39, 3]. This phenomenon occurs generally due to *negative externalities* — that is, the fact that each joining individual adds collectively to the delays of *all* existing individuals. Pricing controls (e.g., [39]), sequencing policies (e.g., [10]), and network topology (e.g., [40]) have been derived for various system topologies in order to influence self-interested agents to conform to the social optimum.

The first known analysis of equilibrium behavior of rational agents in queueing systems is presented by Naor [39], which shows that agents join a single-server queue more readily than is socially optimal. Many extensions and generalizations for the single station queue system have been made [34]. Equilibrium behavior has also been explored in parallel, open, and closed networks. These topologies are discussed next in turn.

In one example of equilibrium routing in parallel systems [3], arriving agents can know only the holding costs and service distribution of each parallel server. Agents tend to prefer servers with small cost per expected service time more than is socially optimal. Finally, a modified pricing scheme is developed for parallel M/M/1, pre-emptive LIFO queues in which an arriving individual is charged explicitly for causing others to wait. It is shown that this new market incentive is the same as the optimality conditions, thus inducing socially optimal behavior from rational individual behavior. This final result has been extended to any service distribution and any number of servers [18].

In the context of a particular well-studied open network configuration, Parlaktürk and Kumar [10] show that both the join-shortest-queue and first-in-first-out queue discipline are unstable with rational agents. New disciplines are developed which stabilize the system in equilibrium, though no claims are made regarding the optimality of these new queue disciplines. More recently, Xia and Shihada [11] have explored the optimality of a decentralized control mechanism for wireless data networks by using network approximations to find closed-form optimality conditions.

Finally, Adelman [5] addresses decentralized routing of agents in a closed queueing network by formulating the problem as a Markov decision process for each entity. Several value function approximations using (non)linear programming are presented. A method is also provided for determining the optimal number of agents circulating in the network.

The contributions of the research presented are three-fold. First, a novel method is presented which provides approximate routing probabilities that maximize the average network throughput for general closed bipartite networks. Closed networks are, in general, difficult to analyze; thus, three open-network approximations from the literature are used to formulate a non-linear program and the resulting solutions compared. All of the approximations are shown to provide average network throughput comparable to existing iterative, optimal methods in the literature, though their robustness and accuracy varies. The advantage of this new method over the existing methods is that the use of approximations allows for the development of a closed-form representation of the optimality conditions in terms of entity flows. These then give significant insight into optimal routing in general closed bipartite networks (e.g., the effect of travel times between nodes in the network and service rate heterogeneity).

Secondly, insight gained from the closed-form optimality conditions also provides a means of determining effective market incentives (i.e., price controls) for the case in which the customers are autonomous rational decision makers in the sense that they may choose their own routing in order to maximize individual profit. It is shown that a uniform market context (i.e., one which uniformly rewards visits to each node) will produce an equilibrium which will produce sub-optimal average network throughput in general.

Finally, while it is well known that a routing strategy in which each customer follows a deterministic path (called vertex routing) maximizes average network throughput, no known methods exist for finding a partition of customers among possible paths in the network. In this work, the open-network approximations are further used to develop a partition allocation algorithm which provides such a vertex routing solution. This solution is shown to have significantly higher average

network throughput than the best probabilistic routing. Further, the empirical optimality gap is shown to be small on average for the 3000 randomly generated networks considered.

3. Problem Formulation and Methodology

As described previously, much work has been done in the area of optimal design of queueing systems [1]. However, very little work in optimal routing of agents in closed networks has been done. In one notable example, Kobayashi and Gerla present an iterative method for finding optimal routing probabilities based on mean value analysis [7]. The primary drawback of this work is that no closed-form optimality conditions can be derived using this methodology. Further, while it is known that the optimal routing strategy is one in which each entity follows a deterministic routing policy [4, 6, 9], there is no known method for finding these individual routing strategies. Others have looked at the dual problem, that is, designing the network capacities to maximize throughput given fixed routing probabilities (e.g., [15, 41]).

To evaluate the effectiveness of a payment policy (which influences the resulting routing behavior of rational agents operating in this network), closed-form optimality conditions are required. It is apparent that these existing iterative methods do not satisfy these requirements and thus a system approximation and alternate method are needed. The work presented here provides a methodology for determining routing probabilities which exhibits optimality conditions and for determining a deterministic routing policy from these probabilities.

Approximations of queueing networks include system approximations (i.e., to obtain a product-form solution), decomposition methods (i.e., assume nodes of network are independent), and process approximations (i.e., continuous fluid flows for heavy traffic) [36]. Of particular interest for this work is Whitt's *fixed population mean* (FPM) method and other variants of this FPM approximation [42]. In this closed network approximation, the nodes are assumed to be independent queues in an open network in which the sum of the mean number of agents at each node is equal to the number of agents in the closed network being approximated. This FPM approximation along with optimization has been previously used in the context of workload allocation among server pools [43].

The general problem of interest for this work is that of allocating agents circulating in a bipartite closed network to maximize system throughput. The particular network structure of interest is shown in Figure 1. Here agents repeatedly travel from any number of parallel nodes to a (fewer) number of central nodes, here called a *parallel-cycle* network. Each node has an associated congestion (latency) function which is increasing with traffic flow and each path has some travel delay.

As an example, for the debris removal example, the parallel nodes can represent the pickup sites, whereas the central nodes can represent the TDSR sites.

Figure 1: General parallel-cycle network with m central nodes and n parallel nodes.

Let $\mathcal{C}$ be the set of central nodes ($|\mathcal{C}| = m$) and $\mathcal{P}$ be the set of parallel nodes ($|\mathcal{P}| = n$). Then, let $i \in \mathcal{P}$ be a parallel node and $j \in \mathcal{C}$ be a central node. Consider the following parallel-cycle partition (PCP) problem formulation in which the continuous flows, λ_{ij} , are the decision variables, the constraint is the number of agents in the system, and the objective is to maximize the total system throughput. The relevant variables are shown in Table 1 along with additional notation.

Table 1: Variables for PCP problem.

<i>Variable</i>	<i>Description</i>
λ_{ij}	Flow assigned to cycle ij (agents/hr)
$\hat{\lambda}_j$	Total flow to central node j ($\sum_i \lambda_{ij}$)
$\bar{\lambda}_i$	Total flow to parallel node i ($\sum_j \lambda_{ij}$)
μ_i	Average service rate of node i
N	Total number of agents in the system
ρ_i	Average utilization of node i (λ_i/μ_i)

Now, let these nodes be single server queues with known service time distributions. Note that the terms *node* and *server* will be used interchangeably throughout this paper. Using the FPM approximation, Little's Law (expected number of agents at each server equals average arrival rate times the expected wait time [36]) can be used to formulate a necessary constraint on the total number of agents. The PCP problem can then be written as the following nonlinear program:

$$\max \quad T = \sum_{i=1}^n \sum_{j=1}^m \lambda_{ij},$$

$$\text{s.t.} \quad \sum_{j=1}^m \hat{\lambda}_j C_j(\hat{\lambda}_j) + \sum_{i=1}^n \bar{\lambda}_i D_i(\bar{\lambda}_i) + \sum_{i=1}^n \sum_{j=1}^m \lambda_{ij} E_{ij}(\lambda_{ij}) \leq N, \quad (1)$$

$$\sum_{j=1}^m \lambda_{ij} \leq \mu_i \quad \text{for } i = 1, \dots, n, \quad (2)$$

$$\sum_{i=1}^n \lambda_{ij} \leq \mu_j \quad \text{for } j = 1, \dots, m, \quad (3)$$

$$\lambda_{ij} \geq 0 \quad \text{for } i = 1, \dots, n, \quad j = 1, \dots, m. \quad (4)$$

Here $C_j(\lambda), D_j(\lambda), E_{ij}(\lambda): \mathcal{R}^+ \mapsto [0, \infty]$ are general mean congestion (i.e., latency) functions at the central and parallel servers and for the round trip travel delay between i and j , respectively.

Note that the first constraint could be written as an inequality because of the form of the objective (i.e., sum of positive terms strictly increasing in λ_{ij}). The remaining constraints simply imply non-negative, feasible flows. The complexity of solving the PCP problem largely depends on the properties of the congestion and travel delay functions, $C_j(\lambda), D_i(\lambda), E_{ij}(\lambda)$. If the functions are all convex, which is the case of the the FPM approximation [42], problem can be solved using standard convex programming methods. If the congestion or travel delay functions are not convex, then the PCP could be hard to solve, and it is not clear if it can be solved in polynomial time (may depend on the exact structure of these functions). That is the case for one of the FPM approximation variants from [42] that we study later in this paper. Our focus here is not on finding the optimal solution of the PCP problem computationally, but on deriving insights from its optimality conditions. While the optimality conditions (which can sometimes be expressed in closed form) can also help in efficient computation of the optimal solutions, exploration of that issue is beyond the scope of this work.

For open networks, assuming (external) Poisson arrivals and exponential servers, these congestion functions have a complete closed-form representation since each server is known to be independent [36]. On the other hand, for closed networks a normalization constant must be found. The two most common approaches to approximating this constant are through Buzen's convolution algorithm [44] and mean value analysis [45]. Both of these are non-polynomial algorithms and neither admits a general closed-form congestion function suitable for use in solving the PCP problem in the current framework.

Figure 2: Example parallel-cycle queueing network with two parallel nodes and a single central node.

Figure 3: Feasible region along with stability constraints and objective gradient for two parallel cycles.

The FPM approximation assumes that these servers are indeed independent and uses the open network congestion functions. This approximation is asymptotically correct for a large number of nodes and agents, and the ratio of any two server utilizations in the approximate model is equal to that in the actual closed system [42]. Thus, we expect the ratio of optimal flows to also be equal for large number of nodes and agents (as $\rho = \lambda/\mu$).

The objective is linear (and thus also convex) while the feasible region using M/M/c congestion functions is known to be convex [46, 47]. Therefore, the first-order KKT conditions are both necessary and sufficient to find globally optimal flows under this Markovian assumption. To illustrate these optimality conditions, a small example is described before providing the general result.

3.1. Two Parallel Cycles Example

Consider the system composed of a single central node and two parallel nodes as shown in Figure 2. Let these nodes have exponentially distributed service times which are known to produce central and parallel node delay functions given by [36]

$$C(\lambda_1, \lambda_2) = \frac{1}{\mu - \lambda_1 - \lambda_2} \text{ and } D_i(\lambda_i) = \frac{1}{\mu_i - \lambda_i}, i \in \{1, 2\}.$$

Suppose also that there are no congestion effects on the travel delay. Then E_{ij} is a constant, denoted by e_1 and e_2 for the two cycles. The constraints are shown generically in Figure 3 along with the stability boundaries ($\lambda < \mu$) and objective gradient. Note that when assuming M/M/1 nodes, constraints (2) and (3) are never at equality as the number of agents would approach ∞ , thus violating Constraint (1). Therefore, as redundant constraints, they are removed for the remaining analysis for this example.

Let f denote the objective $(\lambda_1 + \lambda_2)$, and g denote the set of three constraints (Constraint (1), and Constraint (4) for each of the two decision variables). Then taking derivatives with respect to both decision variables, the gradients are obtained as

$$\begin{aligned}\nabla f &= \begin{bmatrix} 1 & 1 \end{bmatrix} \\ \nabla g &= \begin{bmatrix} \frac{\mu}{(\mu - \lambda_1 - \lambda_2)^2} + \frac{\mu_1}{(\mu_1 - \lambda_1)^2} + e_1 & -1 & 0 \\ \frac{\mu}{(\mu - \lambda_1 - \lambda_2)^2} + \frac{\mu_2}{(\mu_2 - \lambda_2)^2} + e_2 & 0 & -1 \end{bmatrix}^T.\end{aligned}$$

These gradients then give the following two equations from the KKT first-order condition:

$$u_1 \left(\frac{\mu}{(\mu - \lambda_1 - \lambda_2)^2} + \frac{\mu_1}{(\mu_1 - \lambda_1)^2} + e_1 \right) - u_{4,1} = 1$$

$$u_1 \left(\frac{\mu}{(\mu - \lambda_1 - \lambda_2)^2} + \frac{\mu_2}{(\mu_2 - \lambda_2)^2} + e_2 \right) - u_{4,2} = 1$$

where u_1 is the Lagrange multiplier for Constraint (1) and $u_{4,1}, u_{4,2}$ are the multipliers for Constraint 4. For both of these equations to hold with non-trivial flows (i.e., $u_{4,1} = u_{4,2} = 0$), it must be true that

$$\frac{\mu_1}{(\mu_1 - \lambda_1)^2} - \frac{\mu_2}{(\mu_2 - \lambda_2)^2} = e_2 - e_1$$

while the first constraint is at equality (i.e., $u_1 > 0$). Further, one can see that this takes the form of a hyperbola in the parallel congestion (i.e., wait) times, given by

$$\frac{D_1^2}{A^2} - \frac{D_2^2}{B^2} = 1$$

where $A = \sqrt{\frac{e_2 - e_1}{\mu_1}}$ and $B = \sqrt{\frac{e_2 - e_1}{\mu_2}}$. As $e_2 - e_1 \rightarrow 0$ (i.e., the mean travel times of each possible route are the same), the first quadrant solution tends to

$$D_2 = \sqrt{\frac{\mu_1}{\mu_2}} D_1. \quad (5)$$

Thus, the maximum throughput allocation of flows is such that the mean wait time of the slower server, say server 2 without loss of generality, is larger than the mean wait time at the faster server by a factor of $\sqrt{\frac{\mu_1}{\mu_2}} > 1$.

3.1.1. Price Controls

When this is compared to the rational Wardrop-like equilibrium for this case ($D_1 + e_1 = D_2 + e_2$, i.e., the expected wait times at each pickup site plus travel time are the same), it is clear that for $\mu_1 \neq \mu_2$, the rational equilibrium is *not* optimal. Thus, a uniform incentive structure will not result in an efficient equilibrium in general. Assuming an economic Wardrop-like equilibrium as given in Definition 2, the market context which will induce an efficient equilibrium would be one such that

$$\frac{r_1}{D_1 + e_1} = \frac{r_2}{D_2 + e_2}$$

where r_1, r_2 are the per-cycle rewards for parallel servers 1 and 2, respectively. For the case considered above, where $e_2 - e_1 \rightarrow 0$, the optimal rewards are given by

$$r_2 = r_1 \left(\frac{\sqrt{\frac{\mu_1}{\mu_2}} + \zeta}{1 + \zeta} \right)$$

where $\zeta = e_1/D_1 = e_2/D_1$ (i.e., the ratio between travel time and mean wait time at queue 1). Thus, the optimal flows are induced by an incentive structure that rewards the slower server proportionally more than the faster server. Further, this proportion decreases as travel time dominates queue wait time (i.e. $\zeta \rightarrow \infty$). Thus, uniform rewards are optimal for cases with homogeneous parallel servers or relatively high travel time.

3.2. General Optimality Condition

For an arbitrary number of parallel cycles, n , and central servers, m , the general gradients are given by

$$\nabla f = \mathcal{E}_{nm}^T$$

$$\nabla g = \begin{bmatrix} \mathcal{A}_1 & \mathcal{B}_1 & \mathcal{I}_n & -\mathcal{I}_n & 0 & \cdots & 0 \\ \mathcal{A}_2 & \mathcal{B}_2 & \mathcal{I}_n & 0 & -\mathcal{I}_n & & \vdots \\ \vdots & \vdots & \vdots & \vdots & & \ddots & 0 \\ \mathcal{A}_m & \mathcal{B}_n & \mathcal{I}_n & 0 & \cdots & 0 & -\mathcal{I}_n \end{bmatrix}^T.$$

where $\mathcal{E}_{nm}$ is a column vector of ones of length nm (the number of flow decision variables), $\mathcal{I}_n$ is an n -dimensional identity matrix, $\mathcal{B}_i$ is a $m \times n$ matrix whose entries are all zero except for the i^{th} column, which is all ones, and each element, i , in the column vector $\mathcal{A}_j$ is

$$\frac{\partial \hat{\lambda}_j C_j}{\partial \lambda_{ij}} + \frac{\partial \bar{\lambda}_i D_i}{\partial \lambda_{ij}} + \frac{\partial \lambda_{ij} E_{ij}}{\partial \lambda_{ij}}, \quad i = 1, \dots, n.$$

The first-order KKT conditions require that all non-zero, feasible flows (i.e. those flows for which only Constraint (1) is active) satisfy

$$\frac{\partial \hat{\lambda}_j C_j}{\partial \lambda_{ij}} + \frac{\partial \bar{\lambda}_i D_i}{\partial \lambda_{ij}} + \frac{\partial \lambda_{ij} E_{ij}}{\partial \lambda_{ij}} = \frac{\partial \hat{\lambda}_l C_l}{\partial \lambda_{kl}} + \frac{\partial \bar{\lambda}_k D_k}{\partial \lambda_{kl}} + \frac{\partial \lambda_{kl} E_{kl}}{\partial \lambda_{kl}}, \quad \forall (i, j), (k, l) \text{ where } \lambda_{ij}, \lambda_{kl} > 0,$$

and the optimal point intersects with the first constraint at equality.

These conditions for each of the three exponential server approximations presented by Whitt [42] are now considered. Travel congestion effects are neglected throughout for clarity of presentation. Instead, constant mean travel times of $E_{ij}(\lambda_{ij}) = e_{ij}$ will be used.

3.2.1. Infinite Buffer Queues

The *fixed population mean* (FPM) approximation of Whitt [42] is first applied to our model. This first approximation assumes that the network is open (i.e., each node can be treated as an independent queue) and that the total number of entities in the open network approximation is constrained to be equal to the number circulating in the closed system. When the nodes are approximated as $M/M/1$ queues, the expected queue length at each of the central nodes is simply

$$\hat{\lambda}_j C_j(\hat{\lambda}_j) = \frac{\hat{\lambda}_j}{\mu_j - \hat{\lambda}_j}.$$

The expected queue length at the parallel nodes can be expressed similarly. The constraint in the approximation on the total number of entities on average is captured in Constraint 1 of the PCP problem. This approximation then gives the following first-order KKT conditions which all non-zero, feasible flows must satisfy:

$$\mu_i D_i^2 + \mu_j C_j^2 + e_{ij} = \mu_k D_k^2 + \mu_l C_l^2 + e_{kl}, \quad \forall (i, j), (k, l) \text{ where } \lambda_{ij}, \lambda_{kl} > 0.$$

As mentioned earlier, due to the convexity of the congestion (latency) functions, the first-order KKT conditions are both necessary and sufficient for globally optimality. For the case in which all travel times are equal,

$$\sqrt{\mu_j} C_j = \sqrt{\mu_l} C_l, \quad \forall j, l \in \mathcal{C} \text{ and}$$

$$\sqrt{\mu_i} D_i = \sqrt{\mu_k} D_k, \quad \forall i, k \in \mathcal{P}.$$

Again, it is clear that the Wardrop-like rational equilibrium ($C_j + D_i = C_l + D_k$, i.e., all possible routes have equal delays) is *not* optimal in general. Therefore, any market context which encourages this kind of equilibrium where wait times are equal among all possible options will lead to suboptimal performance for cases in which the servers are heterogeneous.

3.2.2. Finite Buffer Queues

The second approximation suggested by Whitt is called FPM with *finite waiting room* (FWR) [42]. In this approximation, the M/M/1 state probability distribution at each node is conditioned on the maximum population in the closed network (i.e., $\pi_i = p(x = i | x \leq N)$). This truncation ensures that there is zero probability that any one node has more entities in the approximated open system than what exist in the closed system of interest. The expected queue length is then

$$\mathbb{E}\{L | L \leq N\} = \sum_{i=0}^N i \left(\frac{(1-\rho)\rho^i}{\sum_{j=0}^N (1-\rho)\rho^j} \right).$$

where $\rho = \lambda/\mu$. Evaluating these sums, and invoking L'Hôpital's rule twice for the $\rho = 1$ case, gives the expression for the expected length at the central server j as

$$\hat{\lambda}_j C_j(\hat{\lambda}_j) = \begin{cases} \left(\frac{1-(N+1)\rho_j^N + N\rho_j^{N+1}}{1-\rho_j^{N+1}} \right) \left(\frac{\rho_j}{1-\rho_j} \right) & \rho_j \neq 1 \\ \frac{N}{2} & \rho_j = 1 \end{cases} \quad (6)$$

where $\rho_j = \hat{\lambda}_j/\mu_j$, and similarly for the parallel servers.

Note that this approximation allows $\rho > 1$, though its accuracy is known to degrade in such circumstances [42]. This is avoided by Constraints (2) and (3). However, it is then possible for some networks to obtain a solution which has fewer than N agents. This is discussed further in the numerical results.

The required partial derivatives for the KKT conditions, after algebraic manipulation, can be shown to be

$$\begin{aligned} \frac{\partial \hat{\lambda}_j C_j}{\partial \lambda_{ij}} &= \mu_j \left(\frac{1}{(\hat{\lambda}_j - \mu_j)^2} - \frac{(N+1)^2 \hat{\lambda}_j^N \mu_j^N}{(\hat{\lambda}_j^{N+1} - \mu_j^{N+1})^2} \right) \\ &= \mu_j \left(\frac{1}{(\hat{\lambda}_j - \mu_j)^2} \right) + \xi_j(\hat{\lambda}_j, \mu_j, N). \end{aligned}$$

The first-order KKT conditions require that all non-zero flows then satisfy

$$\begin{aligned} \mu_i W_i^2 + \xi_i + \mu_j W_j^2 + \xi_j + e_{ij} = \\ \mu_k W_k^2 + \xi_k + \mu_l W_l^2 + \xi_l + e_{kl}, \\ \forall (i, j), (k, l) \text{ where } \lambda_{ij}, \lambda_{kl} > 0, \end{aligned}$$

where W is the M/M/1 waiting function for comparison with the previous results. It can be shown that the congestion (latency) functions in this case are not convex in general. The expression for the expected queue length as in (6) is typically convex in the range $0 \leq \rho_j \leq 1$, and concave when $\rho_j > 1$. Therefore, the above first-order KKT conditions need not represent sufficient conditions for globally optimality. This fact, along with the complexity of the partial derivative expressions, imply that computing the globally optimal solution of the PCP problem under the FWR approximation may be non-trivial.

3.2.3. Deterministic Arrivals

The final approximation (called DM1) suggested by Whitt to further reduce variability of the distribution of total entities in the open network approximation is to assume deterministic arrivals at each node in the approximation [42]. This assumption, one commonly made in analysis of appointment systems [48], requires an iterative solution method. First, we find the complex root of the characteristic equation, given by

$$z = e^{-\rho^{-1}(1-z)}$$

where $\rho = \lambda/\mu$. For $\rho \leq 1$, it is known that there exists exactly one real root, $r_0 \in (0, 1)$ [36]. Further, this can be found iteratively because of the convexity of the characteristic equation to arbitrarily small tolerance (here 0.001 is used) [36].

The expected wait time for central server j is then given by

$$\hat{\lambda}_j C_j(\hat{\lambda}_j) = \frac{\hat{\lambda}_j}{\mu_j} \left(\frac{1}{1 - r_0(\hat{\lambda}_j/\mu_j)} \right).$$

The expression for the parallel servers follows similarly. Each node in the network is treated as an independent D/M/1 queue, but the total number of entities in this open network approximation is constrained to be equal to the number circulating in the closed system.

3.3. Networks Containing some Fixed Probabilistic Routing

Our framework can be easily extended to the case in which the network has some routing probabilities which cannot be changed. For example, in the system of interest, the central processing node is selected based on the type of material loaded. The routing to the central nodes from the parallel nodes must then be assumed to be probabilistic, as they cannot be controlled. The total flow to each parallel node can, however, still be controlled.

To handle this case, the PCP formulation is modified as follows. The decision variables are now only the total flow for each parallel node, λ_i . The optimal flows are then those which solve the following program:

$$\begin{aligned} \max \quad & T = \sum_{i=1}^n \lambda_i, \\ \text{s.t.} \quad & \sum_{j=1}^m \hat{\lambda}_j C_j(\hat{\lambda}_j) + \sum_{i=1}^n \lambda_i D_i(\lambda_i) + \sum_{i=1}^n \sum_{j=1}^m e_{ij} \lambda_i p_{ij} \leq N, \\ & \lambda_i \leq \mu_i \quad \forall i, \quad \sum_{i=1}^n \lambda_i p_{ij} \leq \mu_j \quad \forall j, \quad \lambda_i \geq 0 \quad \forall i, \end{aligned}$$

where p_{ij} is the pre-determined routing probability from parallel node i to central node j , and $\hat{\lambda}_j$, the total flow to central node j , is now $\sum_i \lambda_i p_{ij}$. Notice that this formulation dramatically reduces the dimensionality of the problem from mn variables to only n variables. In order to utilize the partition allocation algorithm which follows, the individual flows λ_{ij} can be reconstructed from this modified formulation simply by setting $\lambda_{ij} = \lambda_i p_{ij}$. However, whether the problem is easily solvable still depends on the structural properties (such as convexity) of the congestion (latency) functions C_j , D_i , as in the general version of the PCP problem posed earlier.

3.4. Price Controls

The goal of setting prices is to induce an optimal economic Wardrop-like equilibrium. In the simple example presented previously, it was straightforward to write an expression for these prices. For larger systems, particularly those with travel delays, it becomes more cumbersome to do so. Assuming an economic Wardrop-like equilibrium as given in Definition 2, the rewards which will induce an efficient equilibrium would be such that

$$\frac{r_{ij}}{C_j + D_i + E_{ij}} = \frac{r_{kl}}{C_l + D_k + E_{kl}}, \quad \forall (i, j), (k, l) \quad (7)$$

where r_{ij} , r_{kl} are two general per-cycle rewards. While it is not clear whether an explicit expression for the optimal rewards can be written in general, there is a simple procedure to compute the optimal rewards. After solving the PCP problem, choose a reference cycle which will receive some fixed reward. All other rewards can then be determined from this reference cycle by simply evaluating the expected wait times and applying Equation (7). While the expected wait times could be determined from the approximation, it is beneficial to evaluate the desired probabilities (or vertex allocation) using MVA to obtain more accurate wait times than those provided by the approximation.

3.5. Partition Allocation Method

The solution to the problem as posed gives continuous flows, λ_{ij} , for each cycle. These flows now need to be converted to routing probabilities, to allow for comparison of the FPM method (using all three approximations) with prior work, and a partition of the agents into deterministic routes, which is known to be optimal as discussed previously. A probabilistic routing strategy can be obtained by normalizing the flows:

$$p_{ij} = \frac{\lambda_{ij}}{\sum_j \lambda_{ij}} \quad \text{and} \quad p_{ji} = \frac{\lambda_{ij}}{\sum_i \lambda_{ij}} \quad \forall i \in \mathcal{P}, j \in \mathcal{C}.$$

Meanwhile, a novel rounding strategy is required to obtain the deterministic routing assignments (i.e., a partition) which is described next.

Obtaining an optimal integer solution from a relaxed solution is commonly done by either branch-and-bound or by adding cutting planes when the decision variables themselves are desired to take integer values [49]. However, since the number of agents (N) can be large, such approaches can be very computationally demanding if applied to our case. The method proposed to obtain a partition routing strategy is shown in Algorithm 1. First, calculate the mean lengths from the optimal flows (lines 2–4), and then calculate the initial integral solution by rounding down (flooring) the number of agents on each path thus calculated (lines 5–8). The number of agents unallocated, k , is then calculated (line 9). Finally, increase the entity count of the k partitions with the largest non-integer parts by one (line 10) to complete the allocation.

It is important to note that the result of this allocation method is dependent not only on the resulting flows (i.e., routing probabilities), but also of the underlying delay model.

Note that the loop in lines 5–8 and the step in line 10 both run in $O(mn)$ time. Therefore, assuming that the optimal continuous flows λ_{ij} have been computed (the complexity of which will depend on the convexity of the congestion and travel delay functions, C_j , D_i and E_{ij} , as discussed

Algorithm 1 Partition Assignment Method

```

1: procedure ASSIGN( $\lambda, \mu, N$ )
2:    $L_j \leftarrow \bar{\lambda}_j C_j(\bar{\lambda}_j, \mu)$  ▷ Mean central length
3:    $L_i \leftarrow \hat{\lambda}_i D_i(\hat{\lambda}_i, \mu)$  ▷ Mean parallel length
4:    $L_{ij} \leftarrow \lambda_{ij} E_{ij}(\lambda_{ij})$  ▷ Mean travel length
5:   for  $i = 1, n; j = 1, m$  do
6:      $\hat{n}_{ij} \leftarrow L_{ij} + \frac{L_j \bar{\lambda}_j}{\lambda_{ij}} + \frac{L_i \hat{\lambda}_i}{\lambda_{ij}}$ 
7:      $n_{ij} \leftarrow \lfloor \hat{n}_{ij} \rfloor$ 
8:   end for
9:    $k \leftarrow N - \sum_i \sum_j n_{ij}$ 
10:   $n_{ij} \leftarrow n_{ij} + 1$  for  $k$  largest  $\hat{n}_{ij} - n_{ij}$ 
11: end procedure

```

earlier), Algorithm 1 runs in $O(mn)$ time. Since the number of pick up and drop off sites (m, n) would typically be small, Algorithm 1 is quite efficient. In particular, it is worth noting that the complexity of Algorithm 1 is independent of the number of agents, N (which can be large).

4. Numerical Results

To assess the performance of the throughput optimization method using the FPM, FWR, and DM1 approximations, this section presents two numerical studies. The first set of results compares these approximations with methods from the existing literature. Because Whitt's approximations are known to improve in their accuracy with increasing network size and number of entities, the second part of the numerical study seeks to determine an empirical optimality gap and quantify how this gap changes with growing number of agents and network size. In both cases, the problem formulation is implemented in MATLAB and solved with `fmincon`'s interior-point method. This continuous solution of desired network flows was first converted into a probability routing matrix before partitioning the agents (i.e., into fixed routing groups) according to the method described in Section 3.5.

4.1. Comparison with Prior Work

In this portion of the numerical results, the routing probabilities resulting from the proposed method are compared with those presented in Kobayashi and Gerla's Example 1 [7]. This example is a central server system similar to that shown in Figure 2 with three parallel cycles where $\mu = 4$, $\mu_1 = 2$, $\mu_2 = 1$, and $\mu_3 = 0.5$. All travel times between nodes are zero in this example. The resulting routing probabilities from both the proposed method for all three approximations and from Kobayashi and Gerla's MVA method are shown in Figure 4.

Figure 4: Comparison of routing probability results. All MVA results are from Table V in [7]. Left axis shows routing probability to the first parallel server while the right axes show probability to the remaining two parallel servers.

The left axis shows the routing probability to the first parallel server, indicated with “+” markers. The known optimal routing proportion with 1 entity is 100% while the three approximations give 88%-92%. This optimal proportion decreases as the number of entities increases. The right axis shows the routing proportions for the second (indicated with “o” markers) and third servers (indicated with “x” markers). These routing proportions increase with increasing entities in the system, with the second server being favored in general.

The FPM and DM1 approximations lead to allocations which significantly under-allocate flow to the fastest parallel server while over-allocating to the slower two. The converse is generally true for the FWR approximation.

The differences in routing probabilities between the approximations and the MVA results are generally decreasing as the number of agents increases. Finally, as N increases, the difference between the routing probabilities given by the three approximations and the known optimal probabilities decrease as expected. Further, note that the proposed method converges to the *balanced* probabilities for all three approximations in which the utilizations are equalized (i.e., the well-known *proportional routing* asymptotic result [50]).

To understand the impact of these small differences in routing probabilities on resulting system throughput, simulations were performed for the $N = 8$ case with the four sets of routing proba-

Figure 5: Histogram of system throughput using the three approximations' probabilistic routing (white) and partition routing (black) strategies compared to that given by the MVA method for 100 simulation runs.

bilities shown in Figure 4 (MVA: $[0.644, 0.260, 0.097]$, FPM: $[0.621, 0.271, 0.108]$, FWR: $[0.715, 0.219, 0.066]$, and DM1: $[0.597, 0.276, 0.127]$) as well as for the resulting partition routing allocations (FPM: $[4, 3, 1]$, FWR: $[5, 2, 1]$, and DM1: $[4, 2, 2]$). This case was chosen as it is the first one in which the three approximations give distinct partition allocations. Each configuration was subjected to 100 simulation runs with a duration of 5000 time units (with a 1000 unit warm-up period).

The histograms of the resulting average system throughput are shown in Figure 5. These histograms indicate the distributions of the average system throughput over the 100 simulation runs. Little difference is observed in terms of mean system throughput between the probabilistic routing based on the three approximation methods relative to the MVA (FPM: -0.50%, FWR: -2.54%, DM1: -1.71%). While the FPM approximation is the simplest, it nonetheless provides the best routing probabilities for this example.

The partition routing strategy (obtained using Algorithm 1) significantly out-performs the MVA probabilistic routing strategy, for all the FPM, FWR and DM1 approximations. The DM1 approximation gives the highest mean throughput (3.13). This throughput is 15% higher than the MVA

Figure 6: Network configurations under study.

probabilistic routing. The FPM and FWR approximations also provide competitive partitions having throughput within 0.24% and 0.53% of the DM1 partition, respectively.

4.2. Empirical Optimality Gap

To evaluate the optimality gap of the proposed partitioning method, a large set of small networks was created and optimal partitions found empirically via complete enumeration of the solution space. For this test, three configurations were considered: 3 nodes and 2 decision variables (DVs; i.e., flows), 4 nodes and 4 DVs, and 5 nodes and 6 DVs, as shown in Figure 6. All possible assignments were evaluated using mean value analysis and an optimal assignment was selected. The number of assignments grows rapidly with increasing number of decision variables, reaching over 1.2 million for the 6 DV case with 40 agents as shown in Figure 7.

Figure 7: Number of possible allocations.

One thousand random networks were generated for each of the three configurations. The number of agents was chosen randomly between five and forty. The server rates were chosen uniformly between 1/2 and 6, while travel times were chosen uniformly between 1/12 and 1. For each random instance generated, the partition allocation was determined for each of the three approximations

Figure 8: Cumulative distribution of optimality gap for all approximations and system configurations.

Figure 9: Cumulative distribution of optimality gap for each approximations by system configurations and number of agents.

using the method described previously. Only those which are solved to optimality are included (i.e., those which exceeded the maximum number of iterations, change in solution between iterations fell below some threshold, or which did not have any feasible solution are excluded). All system throughput results were generated using PDQ (v.6.1.1, approximate MVA method) in the C programming language [51].

The empirical optimality gap itself is calculated by considering the total system throughput of the system under the calculated partition compared to the known optimal partition from complete enumeration. The total results for all configurations are shown as a cumulative distribution function (CDF) for each of the three approximations in Figure 8. The FPM and DM1 approximations are shown to provide similar performance, with an optimality gap of less than 1% for nearly 80% of the 3000 random instances considered. The FWR approximation performs worse, achieving similar performance only 60% of the time. However, this approximation has significantly better worst-case performance. The mean optimality gaps are 1.29%, 1.13%, and 1.29% for FPM, FWR, and DM1 approximations respectively. The medians are significantly less at 0.07%, 0.37%, and 0.09%. Thus, the approximations perform quite well for the majority of cases. However, the worst case optimality gaps are 26.78%, 16.86%, and 26.76%. So while FWR performs best on average and in the worst-case, FPM performs best in the median. The DM1 approximation has a small advantage over FPM for larger systems.

The results are now broken out according to both the number of decision variables and the number of agents. The CDFs, shown in Figure 9, indicate that the proposed method degrades more rapidly with increasing number of decision variables than agents. The FWR approximation, while not performing well in general, experiences *better* performance with increasing agents. The max, mean, and median optimality gaps for each category of network size and agents are shown in Tables 2 and 3. All three statistics increase with network size and number of agents for all but the FWR approximation. Even for the 6 DV cases, the mean and median optimality gaps are reasonably small: 2.86% and 0.66% for FPM, 2.03% and 1.01% for FWR, and 2.85% and 0.61% for DM1.

4.3. Discussion

The numerical results presented above show that the new partition method clearly out-performs existing probabilistic routing methods by producing a mean system throughput 15% higher from prior literature in the instance that was examined. In addition, this new formulation provides in-

Table 2: Empirical optimality gap results by number of decision variables (%).

<i>Approx</i>		<i>Overall</i>	<i>2DV_s</i>	<i>4DV_s</i>	<i>6DV_s</i>
FPM	max	26.78	2.63	21.28	26.78
	mean	1.29	0.15	0.88	2.86
	median	0.07	0.00	0.07	0.66
FWR*	max	16.85	8.53	15.33	16.85
	mean	1.13	0.35	1.10	2.03
	median	0.37	0.18	0.43	1.01
DM1 [†]	max	26.76	2.63	21.25	26.76
	mean	1.29	0.15	0.86	2.85
	median	0.09	0.00	0.07	0.61

Notes: * 57.3% and [†] 62.8% of cases solved to optimality

Table 3: Empirical optimality gap results by number of agents (%).

<i>Approx</i>		<i>Overall</i>	<i>5–16</i>	<i>17–28</i>	<i>29–40</i>
FPM	max	26.78	18.36	25.31	26.78
	mean	1.29	0.39	1.23	2.28
	median	0.07	0.01	0.08	0.19
FWR*	max	16.85	15.33	14.09	16.85
	mean	1.13	1.39	0.81	1.11
	median	0.37	0.51	0.30	0.32
DM1 [†]	max	26.76	18.33	25.08	26.76
	mean	1.29	0.36	1.24	2.27
	median	0.09	0.01	0.11	0.21

Notes: * 57.3% and [†] 62.8% of cases solved to optimality

sights into the properties of the market context which induces efficient rational dispatcher behavior. A uniform reward policy is preferred only when travel times are much larger than queue wait times or when travel times are equal and the servers are homogeneous.

The second numerical study determined empirical optimality gaps when using this new partition strategy for a large set of problem instances in which provably optimal solutions could be obtained. The partition routing method proposed was observed to provide allocations using the FWR approximation having throughput within 1.13% of the known optimal throughput on average and within 16.85% in the worst case for the three thousand instances considered. The FPM and DM1 approximations performed slightly worse, both having throughput within 1.29% of the known optimal throughput on average and 26.78% and 26.76% in the worst case, respectively. The performance of all three approximations degrades with increasing network size (i.e., number of decision

variables) and increasing number of agents in the system. This is a counter-intuitive result given that some of the approximations have been shown to be asymptotically correct [42]. It may be the case that performance improves for even larger systems as those evaluated in this paper are relatively small. Exploring performance for larger systems remains as future work.

The differences in performance among the three approximations generally decreases as the systems get larger. This implies that the choice of approximation matters less for large systems, but remains critical for smaller systems. One important aspect for choosing an approximation is the robustness of the approximation (i.e., the likelihood of solving the problem to optimality). The FPM approximation is the simplest and most robust of the approximations considered. It was able to solve all of the random instances considered. The FWR approximation was able to solve 57.3%, while the DM1 approximation did slightly better at 62.8%. There are two reasons for this: the first is numerical (i.e., too many iterations), while the second is conceptual. The numerical issues may be due to lack of convexity in the FWR and DM1 approximations. As mentioned earlier, it can be shown that the PCP problem under the FPM approximation is a convex program, while it is non-convex under the FWR approximation. For the DM1 approximation, we could not ascertain whether the PCP problem is convex or not. However, the numerical issues can likely be solved on a case-by-case basis through the selection of algorithm parameters (the defaults were used here). Conceptually, both the FWR and DM1 approximations can have a null feasible set when servers are fully utilized. This is particularly common in smaller systems with a larger number of agents. One way to alleviate this conceptual gap is to relax Constraints (2) and (3) which require the flows to be less than the service rates. While this will provide routing probabilities and a partition as desired, the performance is likely to be poor [42].

Finally, we return to our motivating problem. In the case of debris removal, the network topology and number of entities is slowly changing over the course of the clean up efforts. These changes occur on the time scale of days or weeks. The pricing strategy developed here using a steady-state approximation has been shown to work better than any other strategy with a slowly changing network in another simulation study [52].

5. Concluding Remarks

This paper presents a new methodology for determining routing probabilities for closed bipartite networks using three open network approximations, and utilizing the method to allocating agents circulating in the network using math programming framework. The approximations are observed

to result in average network throughputs comparable to existing iterative, optimal methods, while allowing for closed-form representation of the optimality conditions. The closed-form optimality conditions can potentially be used for determining price controls (incentives) for situations where the customers are autonomous rational decision makers. Further, a method for determining a partition of the agents into deterministic routes, that performs significantly better than probabilistic routing, is presented. Numerical evaluation over randomly generated networks show that the empirical gap is small.

While we have argued that a uniform market context will in general produce an equilibrium that is sub-optimal in terms of network throughput, performance characterization of the uniform rational equilibrium remains an interesting topic of future research. Furthermore, understanding the performance improvement that can be attained through non-uniform pricing (possibly as a function of network properties) also remains open for future investigation.

Finally, while the motivating domain was debris removal following natural disasters, the proposed framework is general and can be applied to any system for which a closed queueing network model considered here is appropriate. This includes mining, logging, and other material transport systems.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments towards improving this work.

The contributions of J. Brooks and D. Mendonça are based upon work supported by the National Science Foundation under Grant Nos. 1313589 and 1363513. The contribution of K. Kar was partly supported by the National Science Foundation under Grant No. 1218374.

References

- [1] S. Stidham, *Optimal Design of Queueing Systems*. CRC Press, 2009.
- [2] L. Kleinrock, “On the modeling and analysis of computer networks,” *Proceedings of the IEEE*, vol. 81, no. 8, pp. 1179–1191, 1993.
- [3] C. E. Bell and S. Stidham, “Individual versus social optimization in the allocation of customers to alternative servers,” *Management Science*, vol. 29, no. 7, pp. 831–840, 1983.
- [4] S. K. Tripathi and C. M. Woodside, “A vertex-allocation theorem for resources in queueing networks,” *Journal of the ACM*, vol. 35, no. 1, pp. 221–230, 1988.

- [5] D. Adelman, "Price-directed control of a closed logistics queueing network," *Operations Research*, vol. 55, no. 6, pp. 1022–1038, 2007.
- [6] W. C. Cheng and R. R. Muntz, "Optimal routing for closed queueing networks," *Performance Evaluation*, vol. 13, pp. 3–15, 1991.
- [7] H. Kobayashi and M. Gerla, "Optimal routing in closed queueing networks," *ACM Transactions on Computer Systems*, vol. 1, no. 4, pp. 294–310, 1983.
- [8] J. M. Smith, "Optimal routing in closed queueing networks with state dependent queues," *INFOR*, vol. 49, no. 1, pp. 45–62, 2011.
- [9] A. Hordijk and J. A. Loeve, "Optimal static customer routing in a closed queueing network," *Statistica Neerlandica*, vol. 54, no. 2, pp. 148–159, 2000.
- [10] A. K. Parlaktürk and S. Kumar, "Self-interested routing in queueing networks," *Management Science*, vol. 50, no. 7, pp. 949–966, 2004.
- [11] L. Xia and B. Shihada, "A Jackson network model and threshold policy for joint optimization of energy and delay in multi-hop wireless networks," *European Journal of Operational Research*, vol. 242, no. 3, pp. 778–787, 2015.
- [12] T. Roughgarden, "The price of anarchy is independent of the network topology," *Journal of Computer and System Sciences*, vol. 67, no. 2, pp. 341–364, 2003.
- [13] H. Kameda and Y. Zhang, "Uniqueness of the solution for optimal static routing in open BCMP queueing networks," *Mathematical and Computer Modelling*, vol. 22, no. 10-12, pp. 119–130, 1995.
- [14] A. Seidmann, "Optimal dynamic routing in flexible manufacturing systems," *Annals of Operations Research*, vol. 15, pp. 291–311, 1988.
- [15] P. Whittle, "Optimal routing in Jackson networks," *Asia-Pacific Journal of Operational Research*, vol. 1, pp. 32–37, 1984.
- [16] F. P. Kelly and C. N. Laws, "Dynamic routing in open queueing networks: Brownian models, cut constraints, and resource pooling," *Queueing Systems*, vol. 13, pp. 47–86, 1993.
- [17] R. Morabito, M. C. de Souza, and M. Vazquez, "Approximate decomposition methods for the analysis of multicommodity flow routing in generalized queueing networks," *European Journal of Operational Research*, vol. 232, no. 3, pp. 618–629, 2014.
- [18] T. A. Grossman and M. L. Brandeau, "Optimal pricing for service facilities with self-optimizing customers," *European Journal of Operational Research*, vol. 141, pp. 39–57, 2002.
- [19] J. Anselmi, U. Ayesta, and A. Wierman, "Competition yields efficiency in load balancing games," *Performance Evaluation*, vol. 68, no. 11, pp. 986–1001, 2011.
- [20] B. Vocking, "Selfish load balancing," in *Algorithmic Game Theory*, N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, Eds. Cambridge University Press, 2007, pp. 517–542.
- [21] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," in *STACS 99*. Springer, 1999, pp. 404–413.

- [22] M. Armony, "Dynamic routing in large-scale service systems with heterogeneous servers," *Queueing Systems*, vol. 51, no. 3-4, pp. 287–329, 2005.
- [23] J. Filipiak, "Dynamic routing in a queueing system with a multiple service facility," *Operations Research*, vol. 32, no. 5, pp. 1163–1180, 1984.
- [24] D. J. Houck, "Comparison of policies for routing customers to parallel queueing systems," *Operations Research*, vol. 35, pp. 306–310, 1987.
- [25] W. Lin and P. R. Kumar, "Optimal control of a queueing system with two heterogeneous servers," *IEEE Transactions on Automatic Control*, vol. 29, no. 8, pp. 696–703, 1984.
- [26] W. F. Piepmeier, "Optimal balancing of I/O requests to disks," *Communications of the ACM*, vol. 18, no. 9, pp. 524–527, 1975.
- [27] J. G. Shanthikumar and S. H. Xu, "Asymptotically optimal routing and service rate allocation in a multiserver queueing system," *Operations Research*, vol. 45, no. 3, pp. 464–469, 1997.
- [28] N. T. Argon, L. Ding, K. D. Glazebrook, and S. Ziya, "Dynamic routing of customers with general delay costs in a multiserver queueing system," *Probability in the Engineering and Informational Sciences*, vol. 23, no. 2, pp. 175–203, 2009.
- [29] M. Delasay, B. Kolfal, and A. Ingolfsson, "Maximizing throughput in finite-source parallel queue systems," *European Journal of Operational Research*, vol. 217, no. 3, pp. 554–559, 2012.
- [30] Y.-C. Teh and A. R. Ward, "Critical thresholds for dynamic routing," *Queueing Systems*, vol. 42, no. 3, pp. 297–316, 2002.
- [31] J. Lui, R. Muntz, and D. Towsley, "Bounding the mean response time of the minimum expected delay routing policy: An algorithmic approach," *IEEE Transactions on Computers*, vol. 44, no. 12, pp. 1371–1382, 1995.
- [32] Z. Liu and D. Towsley, "Optimality of the round-robin routing policy," *Journal of Applied Probability*, vol. 31, no. 2, pp. 466–475, 1994.
- [33] W. Winston, "Optimality of the shortest line discipline," *Journal of Applied Probability*, vol. 14, no. 1, pp. 181–189, 1977.
- [34] R. Hassin and M. Haviv, *To Queue or Not to Queue: Equilibrium Behavior in Queueing Systems*. Norwell, Massachusetts: Kluwer, 2003.
- [35] T. Roughgarden, *Selfish Routing and the Price of Anarchy*. Cambridge, MA: MIT Press, 2005.
- [36] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *Fundamentals of Queueing Theory*. John Wiley & Sons, 2008.
- [37] J. G. Wardrop, "Some theoretical aspects of road traffic research," *Proceedings of the Institution of Civil Engineers*, vol. 1, pp. 325–378, 1952.
- [38] J. R. Correa and N. E. Stier-Moses, "Wardrop equilibria," in *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2010.

- [39] P. Naor, "The regulation of queue size by levying tolls," *Econometrica*, vol. 37, no. 1, pp. 15–24, 1969.
- [40] T. Roughgarden, "On the severity of braess's paradox: Designing networks for selfish users is hard," *Journal of Computer and System Sciences*, vol. 72, no. 5, pp. 922–953, Aug. 2006.
- [41] D. L. Bakuli and J. M. Smith, "Theory and methodology resource allocation in state-dependent emergency evacuation networks," *European Journal of Operational Research*, vol. 89, pp. 543–555, 1996.
- [42] W. Whitt, "Open and closed models for networks of queues," *AT&T Bell Laboratories Technical Journal*, vol. 63, no. 9, pp. 1911–1979, 1984.
- [43] J. Calabrese, "Optimal workload allocation in open networks of multiserver queues," *Management Science*, vol. 38, no. 12, pp. 1792–1802, 1992.
- [44] J. P. Buzen, "Computational algorithms for closed queueing networks with exponential servers," *Communications of the ACM*, vol. 16, no. 9, pp. 527–531, 1973.
- [45] M. Reiser and S. S. Lavenberg, "Mean-value analysis of closed multichain queueing networks," *Journal of the ACM*, vol. 27, no. 2, pp. 313–322, 1980.
- [46] W. Grassmann, "The convexity of the mean queue size of the M/M/c queue with respect to the traffic intensity," *Journal of Applied Probability*, vol. 20, pp. 916–919, 1983.
- [47] H. L. Lee and M. A. Cohen, "A note on the convexity of performance measures of M/M/c queueing systems," *Journal of Applied Probability*, vol. 20, pp. 920–923, 1983.
- [48] B. Jansson, "Choosing a good appointment system - A study of queues of the type (D, M, 1)," *Operations Research*, vol. 14, no. 2, pp. 292–312, 1966.
- [49] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*. Wiley New York, 1988, vol. 18.
- [50] J. Zahorj, K. C. Sevcik, D. L. Eager, and B. Galler, "Balanced job bound analysis of queueing networks," *Communications of the ACM*, vol. 25, no. 2, pp. 134–141, 1982.
- [51] N. J. Gunther, *Analyzing Computer System Performance with Perl::PDQ*. Berlin, Heidelberg: Springer-Verlag, 2005.
- [52] J. D. Brooks, "Dynamic resource allocation in human queueing systems through market-based control," Ph.D. dissertation, Dept. of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, 2014.