Dynamic Multi-Vehicle Routing with Multiple Classes of Demands

Marco Pavone   Stephen L. Smith   Francesco Bullo   Emilio Frazzoli

Abstract— In this paper we study a dynamic vehicle routing problem in which there are multiple vehicles and multiple classes of demands. Demands of each class arrive in the environment randomly over time and require a random amount of on-site service that is characteristic of the class. To service a demand, one of the vehicles must travel to the demand location and remain there for the required on-site service time. The quality of service provided to each class is given by the expected delay between the arrival of a demand in the class, and that demand's service completion. The goal is to design a routing policy for the service vehicles which minimizes a convex combination of the delays for each class. First, we provide a lower bound on the achievable values of the convex combination of delays. Then, we propose a novel routing policy and analyze its performance under heavy load conditions (i.e., when the fraction of time the service vehicles spend performing on-site service approaches one). The maximum deviation of the policy's performance from the lower bound depends only on the number of classes, and is independent of the number of vehicles, the arrival rates of demands, the on-site service times, and the convex combination coefficients.

I. INTRODUCTION

Consider a bounded environment $\mathcal{E}$ in the plane which contains n service vehicles. Demands arrive in $\mathcal{E}$ sequentially over time and each demand is a member of one of m classes. Upon arrival, a demand assumes a location in $\mathcal{E}$, and requires a class dependent amount of on-site service time. To service a demand, one of the n vehicles must travel to the demand location and perform the on-site service. If we specify a policy by which the vehicles serve demands, then the expected delay for demands of class α, denoted D_α, is the expected amount of time between a demands arrival and its service completion. Then, given coefficients $c_1, \ldots, c_m > 0$, the goal is to find the vehicle routing policy that minimizes the convex combination

$$c_1 D_1 + \cdots + c_m D_m.$$

By increasing the coefficients for certain classes, a higher priority level can be given to their demands. This problem has important applications in areas such as UAV surveillance, where targets are given different priority levels based on their urgency or potential importance.

In classical queuing theory (i.e., queuing systems in which the demands are not spatially distributed), the problem of priority queues has received much attention, [1]. In [2] the authors characterize the region of delays that are realizable by a single server. This analysis is performed under the

M. Pavone and E. Frazzoli are with the Laboratory for Information and Decision Systems, Aeronautics and Astronautics Department, Massachusetts Institute of Technology, Cambridge, MA 02139, USA; email: {pavone,frazzoli}@mit.edu. S. L. Smith and F. Bullo are with the Center for Control, Dynamical Systems and Computation, Department of Mechanical Engineering, University of California, Santa Barbara, CA 93106, USA {stephen,bullo}@engineering.ucsb.edu.

assumption that the customer (demand) interarrival times and service times are distributed exponentially. In [3] the achievable delays are studied in queuing networks.

If the demands are spatially distributed then the problem becomes one of vehicle routing. With only one class of demands, the problem in this paper is known as the Dynamic Traveling Repairperson Problem (DTRP), which was first introduced by Bertsimas and van Ryzin [4], [5], [6]. These papers study the expected delay of demands and propose several policies that perform within a constant factor of the optimal in both heavy load (i.e., when the fraction of time the service vehicles spend performing on-site service approaches one), and in light load (i.e., when the fraction of time the service vehicles spends performing on-site service approaches zero). They also study vehicles with finite service capacity, and extend their results to arbitrary renewal arrival processes, and nonuniform demand location distributions. In [7], and [8], decentralized policies are developed for the DTRP. Spatial queuing problems have also been studied in the context of urban operations research [9], where approximations are used to cast the problems in the traditional queuing framework. In our previous paper [10], we introduced and studied the multi-class DTRP for the case of two classes and one vehicle. For this case we derived a lower bound on the achievable delay values and proposed the Randomized Priority policy, which performed within a constant factor of the lower bound, for all convex combination coefficients.

The contributions of this paper are as follows. We extend the two-class DTRP to n service vehicles and m classes of demands. The extension of our previous analysis to multiple classes of demands is very nontrivial. We derive a new lower bound on the achievable values of the convex combination of delays, and propose a new policy in which each class of demands is served separately from the others. We show that the maximum deviation of the policy's performance from the lower bound is $2m^2$. Thus, the maximum deviation is independent of the number of vehicles, the arrival rates of demands, the on-site service times, and the convex combination coefficients. We also comment on the source of the gap between the upper and lower bounds.

The paper is organized as follows. In Section II we give some asymptotic properties of the traveling salesperson tour. In Section II-B we formalize the problem and in Section III we derive a lower bound, and in Section IV we introduce and analyze the Separate Queues policy. Finally, in Section V we present simulation results.

II. BACKGROUND AND PROBLEM STATEMENT

In this section we summarize the asymptotic properties of the Euclidean traveling salesperson tour, and formalize the multi-class dynamic traveling repairperson problem.
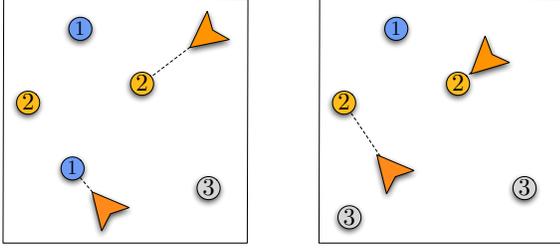
Fig. 1. A depiction of the problem for two vehicles and three classes. Left figure: One vehicle is moving to a class 1 demand, and the other to a class 2 demand. Right figure: The bottom vehicle has serviced the class 1 demand is moving to a class 2 demand. A new class 3 demand has arrived.

A. The Euclidean Traveling Salesperson Problem

Given a set Q of N points in $\mathbb{R}^d$, the Euclidean traveling salesperson problem (TSP) is to find the minimum-length tour of Q (i.e., the shortest closed path through all points). Let $\text{TSP}(Q)$ denote the minimum length of a tour through all the points in Q. Assume that the locations of the N points are random variables independently and identically distributed, uniformly in a compact set $\mathcal{E}$; in [11] it is shown that there exists a constant $\beta_{\text{TSP},d}$ such that, almost surely,

$$\lim_{N \to +\infty} \frac{\text{TSP}(Q)}{N^{1-1/d}} = \beta_{\text{TSP},d} |\mathcal{E}|^{1/d}. \tag{1}$$

The current estimate of the constant for $d = 2$ is $\beta_{\text{TSP},2} \simeq 0.7120$, [12]. The bound in Eq. (1) holds for all compact sets $\mathcal{E}$ and the shape of $\mathcal{E}$ only affects the convergence rate to the limit. If $\mathcal{E}$ is a "fairly compact and fairly convex" set in the plane, then Eq. (1) provides an adequate estimate of the optimal TSP tour length for values of N as low as 15, [9].

B. Problem Statement

Consider a bounded environment $\mathcal{E}$ in the plane with area $|\mathcal{E}|$. The environment contains n vehicles, each with maximum speed v. Demands of type $\alpha \in \{1, \ldots, m\}$ arrive in the environment according to a Poisson process with rate λ_α. Upon arrival, demands assume an independently and uniformly distributed location in $\mathcal{E}$. A demand of type α is serviced when the vehicle spends an on-site service time that is generally distributed with finite mean $\bar{s}_\alpha$.

Consider the arrival of the ith α-demand. The service delay for the ith demand, $D_\alpha(i)$, is the time elapsed between its arrival and its service completion. The wait time is then given by $W_\alpha(i) := D_\alpha(i) - s_\alpha(i)$, where $s_\alpha(i)$ is the on-site service time required by demand i. Given a stable policy P (i.e., a policy for which the α and β queue lengths remain finite), the steady-state service delay is defined as $D_\alpha(P) := \lim_{i \to +\infty} \mathbb{E}[D_\alpha(i)]$, and the steady-state wait is $W_\alpha(P) := D_\alpha(P) - \bar{s}_\alpha$. Then, given a stable policy P, the *average delay per demand* is

$$D(P) = \frac{1}{\Lambda} \sum_{\alpha=1}^m \lambda_\alpha D_\alpha,$$

where $\Lambda := \sum_{\alpha=1}^m \lambda_\alpha$. The average delay per demand is the standard cost functional for queuing systems with multiples classes of demands. Notice that we can write

$D(P) = \sum_{\alpha=1}^m c_\alpha D_\alpha$ with $c_\alpha = \lambda_\alpha / \Lambda$. Then, a possible way to model priority is to allow any convex combination of $D_1, \ldots, D_m$. We are now ready to state our problem.

Problem Statement: Determine the vehicle routing policy P which minimizes

$$D(P) := \sum_{\alpha=1}^m c_\alpha D_\alpha(P), \tag{2}$$

where $c_\alpha > 0$, $\alpha \in \{1, \ldots, m\}$, and $\sum_{\alpha=1}^m c_\alpha = 1$. Notice that if $c_\alpha > \lambda_\alpha / \Lambda$, then the delay of α-demands is being weighted more heavily than in the average case. Thus, the quantity $c_\alpha \Lambda / \lambda_\alpha$ gives the priority of α demands compared to that given in the average delay case. Hence, without loss of generality we can assume the demand classes are labeled so that

$$\frac{c_1}{\lambda_1} \geq \frac{c_2}{\lambda_2} \geq \cdots \geq \frac{c_m}{\lambda_m}, \tag{3}$$

implying that if $\alpha < \beta$ for some $\alpha, \beta \in \{1, \ldots, m\}$, then α-demands are of a higher priority than β-demands.

A necessary condition for there to exist a policy which yields a finite $D(P)$ is

$$\varrho := \frac{1}{n} \sum_{\alpha=1}^m \lambda_\alpha \bar{s}_\alpha < 1, \tag{4}$$

The *load factor* ϱ captures the fraction of time the n service vehicles must be busy in any stable policy.

III. LOWER BOUND IN HEAVY LOAD

In this section we present two lower bounds on the delay in Eq. (2). The first holds only in heavy load (i.e., as $\varrho \to 1^-$), while the second (less tight) bound holds for all ϱ.

Theorem III.1 (Heavy load lower bound) In heavy load $(\varrho \to 1^-)$, for every routing policy P,

$$D(P) \geq \frac{\beta_{\text{TSP}}^2 |\mathcal{E}|}{2n^2 v^2 (1 - \varrho)^2} \sum_{\alpha=1}^m \left(c_\alpha + 2 \sum_{j=\alpha+1}^m c_j \right) \lambda_\alpha. \tag{5}$$

where $\beta_{\text{TSP}} := \beta_{\text{TSP},2}$, and $c_1, \ldots, c_m$ satisfy Eq. (3).

Proof: Consider a tagged demand i of type α, and let us quantify its total service requirement. The demand requires on-site service time $s_\alpha(i)$. let us denote by $d_\alpha(i)$ the distance from the location of the demand served prior to i, to i's location. In order to compute a lower bound on the wait time, we will allow "remote" servicing of some of the demands; in other words, we introduce the variables $r_\alpha \in \{0, 1\}$ for each $\alpha \in \{1, \ldots, m\}$. If $r_\alpha = 1$, then α-demands can be serviced remotely (i.e., a service vehicle can service the ith α-demand from any location, by simply stopping for the on-site service time $s_\alpha(i)$). If $r_\alpha = 0$ then α-demands must be serviced on location.

Thus, the total service requirement of α-demand i is $r_\alpha d_\alpha(i) + s_\alpha(i)$. The steady-state expected service requirement is $r_\alpha \bar{d}_\alpha + \bar{s}_\alpha$, where $\bar{d}_\alpha := \lim_{i \to +\infty} \mathbb{E}[d_\alpha(i)]$. In order to maintain stability of the system we must require that

$$\frac{1}{n} \sum_{\alpha=1}^m \lambda_\alpha \left(\frac{r_\alpha \bar{d}_\alpha}{v} + \bar{s}_\alpha \right) < 1. \tag{6}$$

Applying the definition of ϱ in Eq. (4), we write Eq. (6) as

$$\sum_{\alpha=1}^{m} r_\alpha \lambda_\alpha \bar{d}_\alpha < (1-\varrho)nv. \qquad (7)$$

For a stable policy P, let N_α represent the number of demands of type α in the queue. From a key result in the DTRP literature (see [13], page 23), we have in heavy load ($\varrho \to 1^-$) the following

$$\bar{d}_\alpha \geq \frac{\beta_{\mathrm{TSP}}}{\sqrt{2}} \sqrt{\frac{|\mathcal{E}|}{\sum_\alpha r_\alpha N_\alpha}} =: \bar{d}, \qquad (8)$$

for each $\alpha \in \{1, \ldots, m\}$. Combining Eqs. (7) and (8),

$$\frac{\sum_\alpha r_\alpha \lambda_\alpha}{nv(1-\varrho)} < \frac{1}{\bar{d}}.$$

Applying the definition of $\bar{d}$, squaring both sides, and rearranging we obtain

$$\frac{\beta_{\mathrm{TSP}}^2}{2} \frac{|\mathcal{E}|(\sum_\alpha r_\alpha \lambda_\alpha)^2}{n^2 v^2 (1-\varrho)^2} < \sum_\alpha r_\alpha N_\alpha.$$

From Little's law $N_\alpha = \lambda_\alpha W_\alpha$ for each $\alpha \in \{1, \ldots, m\}$, and thus

$$\sum_\alpha r_\alpha \lambda_\alpha W_\alpha > \frac{\beta_{\mathrm{TSP}}^2}{2} \frac{|\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} \left(\sum_\alpha r_\alpha \lambda_\alpha\right)^2. \qquad (9)$$

Recall that $W_\alpha = D_\alpha - \bar{s}_\alpha$ and $r_\alpha \in \{0,1\}$ for each $\alpha \in \{1, \ldots, m\}$., Eq. (9) gives us $2^m - 1$ constraints on the feasible values of the delays $D_1(P), \ldots, D_m(P)$.

To simplify notation, let us define

$$A := \frac{\beta_{\mathrm{TSP}}^2}{2} \frac{|\mathcal{E}|}{n^2 v^2 (1-\varrho)^2}.$$

Then, a lower bound is the solution to the linear program:

$$\textbf{minimize} \quad \sum_{\alpha=1}^{m} c_\alpha W_\alpha,$$

$$\textbf{subject to} \quad \sum_{\alpha=1}^{m} r_\alpha \lambda_\alpha W_\alpha > A \left(\sum_\alpha r_\alpha \lambda_\alpha\right)^2,$$

$$r_\alpha \in \{0,1\} \quad \forall \alpha \in \{1, \ldots, m\}.$$

With the class labeling in Eq. (3), only m of the $2^m - 1$ constraints are non-redundant and the program can be rewritten as

$$\textbf{minimize} \quad \sum_{\alpha=1}^{m} c_\alpha W_\alpha,$$

$$\textbf{subject to}$$

$$\begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ \lambda_1 & \lambda_2 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & 0 \\ \lambda_1 & \lambda_2 & \lambda_3 & \cdots & \lambda_m \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_m \end{bmatrix} \geq A \begin{bmatrix} \lambda_1^2 \\ (\lambda_1 + \lambda_2)^2 \\ \vdots \\ (\lambda_1 + \cdots + \lambda_m)^2 \end{bmatrix}.$$

The solution $(W_1^*, \ldots, W_m^*)$ of the linear program is

$$W_\alpha^* = A(\lambda_1 + \cdots + \lambda_\alpha)^2 - A(\lambda_1 + \cdots + \lambda_{\alpha-1})^2$$

$$= A \left(\lambda_\alpha + 2 \sum_{j=1}^{\alpha-1} \lambda_j\right).$$

The optimal value of the cost function is

$$\sum_{\alpha=1}^{m} c_\alpha W_\alpha^* = A \sum_{\alpha=1}^{m} c_\alpha \left(\lambda_\alpha + 2 \sum_{j=1}^{\alpha-1} \lambda_j\right)$$

$$= A \sum_{\alpha=1}^{m} \left(c_\alpha + 2 \sum_{j=\alpha+1}^{m} c_j\right) \lambda_\alpha.$$

Applying the definition of A and noting that for every policy P, $D_\alpha(P) \geq W_\alpha^* + \bar{s}_\alpha$, as $\varrho \to 1^-$ we obtain the desired result. ∎

Remark III.2 (Lower bound for all $\varrho \in [0,1)$) With slight modifications, it it possible to obtain a less tight lower bound valid for all values of ϱ. In the above derivation, the assumption that $\varrho \to 1^-$ is used in Eq. (8). It is possible to use, instead, a lower bound valid for all $\varrho \in [0,1)$ (see [5]):

$$\bar{d}_\alpha \geq \gamma \sqrt{\frac{|\mathcal{E}|}{\sum_\alpha r_\alpha N_\alpha + n/2}},$$

where $\gamma = 2/(3\sqrt{2\pi}) \approx 0.266$. Using this bound we obtain the same linear program as in the proof of Theorem III.1, with the difference that A is now a function given by

$$A(x) := \frac{\gamma^2 |\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} x - \frac{n}{2}.$$

Following the procedure in the proof of Theorem III.1

$$W_1^* = \frac{\gamma^2 |\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} \lambda_1 - \frac{n}{2\lambda_1}$$

$$W_\alpha^* = \frac{\gamma^2 |\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} \left(\lambda_\alpha + 2 \sum_{j=1}^{\alpha-1} \lambda_j\right),$$

for each $\alpha \in \{2, \ldots, m\}$. Finally, for every policy P, $D_\alpha(P) \geq W_\alpha^* + \bar{s}_\alpha$, and thus

$$D(P) \geq \frac{\gamma^2 |\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} \sum_{\alpha=1}^{m} \left(\left(c_\alpha + 2 \sum_{j=\alpha+1}^{m} c_j\right) \lambda_\alpha\right)$$

$$- \frac{nc}{2\lambda_1} + \sum_{\alpha=1}^{m} c_\alpha \bar{s}_\alpha, \qquad (10)$$

for all $\varrho \in [0,1)$ under the labeling in Eq. (3). □

IV. SEPARATE QUEUES POLICY

In this section we introduce and analyze the Separate Queues (SQ) policy. We show that this policy is within a factor of $2m^2$ of the lower bound in heavy load.

To present the SQ policy we need some notation. We assume vehicle $k \in \{1, \ldots, n\}$ has a service region $R^{[k]} \subset$

$\mathcal{E}$, such that $\{R^{[1]}, \ldots, R^{[n]}\}$ form a partition of the environment $\mathcal{E}$. In general the partition could be time varying, but for the description of the SQ policy this will not be required. We assume that information on outstanding demands of type $\alpha \in \{1, \ldots, m\}$ in region $R^{[k]}$ at time t is summarized as a finite set of demand positions $Q_\alpha^{[k]}(t)$ with $N_\alpha^{[k]}(t) := \mathrm{card}(Q_\alpha^{[k]}(t))$. Demands of type α with location in $R^{[k]}$ are inserted in the set $Q_\alpha^{[k]}$ as soon as they are generated. Removal from the set $Q_\alpha^{[k]}$ requires that service vehicle k moves to the demand location, and provides the on-site service. With this notation we describe the policy as follows:

Separate Queues (SQ) Policy

Assumes: A probability distribution $\mathbf{p} = [p_1, \ldots, p_m]$.
1 Partition $\mathcal{E}$ into n equal area regions and assign one vehicle to each region.
2 **foreach** vehicle-region pair k **do**
3   **if** the set $\cup_\alpha Q_\alpha^{[k]}$ is empty **then**
4   Move vehicle toward the median of its own region until a demand arrives.
5   **else**
6   Select $Q \in \{Q_1^{[k]}, \ldots, Q_m^{[k]}\}$ according to $\mathbf{p}$.
7   **if** Q is empty **then**
8   Reselect until Q is nonempty.
9   Compute TSP tour through all demands in Q.
10   Service Q following the TSP tour, starting at the demand closest to the vehicle's current position.
11   Repeat.
12 Optimize over $\mathbf{p}$.

A. Stability Analysis of the SQ Policy in Heavy Load

In this section we will analyze the SQ policy in heavy load, i.e., as $\varrho \to 1^-$. In the SQ policy each region $R^{[k]}$ has equal area, and contains a single vehicle. Thus, the n vehicle problem in a region of area $|\mathcal{E}|$ has been turned into n independent 1 vehicle problems, each in a region of area $|\mathcal{E}|/n$, with arrival rates λ_α/n. To determine the performance of the policy we need only study the performance in a single region k. For simplicity of notation we omit the label k. We refer to the time instant t_i in which the vehicle computes a new TSP tour as the epoch i of the policy; we refer to the time interval between epoch i and epoch $i+1$ as the ith iteration and we will refer to its length as T_i. Finally, let $N_\alpha(t_i) := N_{\alpha,i}$, $\alpha \in \{1, \ldots, m\}$, be the number of outstanding α-demands at beginning of iteration i.

The following straightforward lemma, proved in [10], will be essential in deriving our main results.

Lemma IV.1 (Number of outstanding demands) In heavy load (i.e., $\varrho \to 1^-$), after transients, the number of demands serviced in a single tour of the vehicle in the SQ policy is very large with high probability (i.e., with probability that tends to 1 as ϱ approaches 1).

Let TS_j be the event that Q_j is selected for service at iteration i of the SQ policy. By the total probability law

$$\mathbb{E}[N_{\alpha,i+1}] = \sum_{j=1}^m p_j \mathbb{E}(N_{\alpha,i+1}|TS_j), \quad \alpha \in \{1, \ldots, m\},$$

where the conditioning is with respect to the task being performed at iteration i. During iteration i of the policy, demands arrive according to independent Poisson processes. Call $N_{\alpha,i}^{\mathrm{new}}$ the α-demands ($\alpha \in \{1, \ldots, m\}$) newly arrived during iteration i; then, by definition of the SQ policy

$$\mathbb{E}(N_{\alpha,i+1}|TS_j) = \begin{cases} \mathbb{E}(N_{\alpha,i}^{\mathrm{new}}|TS_j), & \text{if } \alpha = j \\ \mathbb{E}(N_{\alpha,i}|TS_j) + \mathbb{E}(N_{\alpha,i}^{\mathrm{new}}|TS_j), & \text{o.w.} \end{cases}$$

By the law of iterated expectation, we can write $\mathbb{E}(N_{\alpha,i}^{\mathrm{new}}|TS_j) = (\lambda_\alpha/n)\mathbb{E}(T_i|TS_j)$. Moreover, since the number of demands outstanding at the beginning of iteration i is independent of the task that will be chosen, we have $\mathbb{E}(N_{\alpha,i}|TS_j) = \mathbb{E}[N_{\alpha,i}]$. Thus we obtain

$$\mathbb{E}(N_{\alpha,i+1}|TS_j) = \begin{cases} \frac{\lambda_\alpha}{n}\mathbb{E}(T_i|TS_j), & \text{if } \alpha = j \\ \mathbb{E}[N_{\alpha,i}] + \frac{\lambda_\alpha}{n}\mathbb{E}(T_i|TS_j), & \text{o.w.} \end{cases}$$

Therefore, we are left with computing the conditional expected values of T_i. The length of T_i is given by the time needed by the vehicle to travel along the TSP tour plus the time spent to service demands. Assuming i large enough, Lemma (IV.1) holds, and we can apply Eq. (1) to estimate from the quantities $N_{\alpha,i}$, $\alpha \in \{1, \ldots, m\}$, the length of the TSP tour at iteration i. Conditioning on task j (when only demands of type j are serviced), we have

$$\mathbb{E}(T_i|TS_j) = \frac{\beta_{\mathrm{TSP}}\sqrt{|\mathcal{E}|/n}}{v}\mathbb{E}\left(\sqrt{N_{j,i}}|TS_j\right) +$$
$$+ \mathbb{E}\left(\sum_{k=1}^{N_{j,i}} s_{j,k}|TS_j\right)$$
$$\leq \frac{\beta_{\mathrm{TSP}}\sqrt{|\mathcal{E}|/n}}{v}\sqrt{\mathbb{E}[N_{j,i}]} + \mathbb{E}[N_{j,i}]\bar{s}_j,$$

where we have

- applied Eq. (1);
- applied Jensen's inequality for concave functions, in the form $\mathbb{E}\left[\sqrt{X}\right] \leq \sqrt{\mathbb{E}[X]}$;
- removed the conditioning on TS_j, since random variables $N_{\alpha,i}$ are independent from *future* events, and in particular from the choice of the task at iteration i;
- used the *crucial* fact that the on-site service times are independent from the number of outstanding demands.

Collecting the above results (for short $\mathbb{E}[X]$ is denoted by $\bar{X}$, where X is any random variable), we have

$$\bar{N}_{\alpha,i+1} \leq (1 - p_\alpha)\bar{N}_{\alpha,i} +$$
$$\sum_{j=1}^m p_j \frac{\lambda_\alpha}{n}\left[\frac{\beta_{\mathrm{TSP}}\sqrt{|\mathcal{E}|}}{\sqrt{n}v}\sqrt{\bar{N}_{j,i}} + \bar{N}_{j,i}\bar{s}_j\right], \quad (11)$$

for each $\alpha \in \{1, \ldots, m\}$. The m inequalities above describe a system of recursive relations for an upper bound on $\bar{N}_{\alpha,i}$,

$\alpha \in \{1, \ldots, m\}$. The following theorem (see Appendix for its proof) bounds the values to which they converge.

Theorem IV.2 (Queue length) For every set of initial conditions $\{\bar{N}_{\alpha,0}\}_{\alpha \in \{1,\ldots,m\}}$, the trajectories $i \mapsto \bar{N}_{\alpha,i}$, $\alpha \in \{1, \ldots, m\}$, resulting from Eqs. (11), satisfy

$$\limsup_{i \to +\infty} \bar{N}_{\alpha,i} \le \frac{\beta_{\text{TSP}}^2 |\mathcal{E}|}{n^3 v^2 (1-\varrho)^2} \frac{\lambda_\alpha}{p_\alpha} \left(\sum_{j=1}^m \sqrt{\lambda_j p_j} \right)^2 .$$

B. Delay of the SQ Policy in Heavy Load

From Theorem IV.2, and using Little's law, the delay of α-demands is

$$D_\alpha(SQ) \le \frac{n}{\lambda_\alpha} \limsup_{i \to +\infty} \bar{N}_{\alpha,i} + \bar{s}_\alpha$$

$$= \frac{\beta_{\text{TSP}}^2 |\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} \frac{1}{p_\alpha} \left(\sum_{j=1}^m \sqrt{\lambda_j p_j} \right)^2 .$$

Thus, the delay (as defined in Eq. (2)) of the SQ policy, is

$$D(SQ) \le \frac{\beta_{\text{TSP}}^2 |\mathcal{E}|}{n^2 v^2 (1-\varrho)^2} \sum_{\alpha=1}^m \frac{c_\alpha}{p_\alpha} \left(\sum_{i=1}^m \sqrt{\lambda_i p_i} \right)^2 . \quad (12)$$

With this expression we prove out main result on the performance of the SQ policy.

Theorem IV.3 (SQ policy performance) In heavy load, the delay of the SQ policy is within $2m^2$ of the optimal, independent of the arrival rates $\lambda_1, \ldots, \lambda_m$, coefficients $c_1, \ldots, c_m$, service times $\bar{s}_1, \ldots, \bar{s}_m$, and the number of vehicles n.

Proof: We would like to compare the performance of this policy with the lower bound. To do this, consider setting

$$p_\alpha := c_\alpha \quad \text{for each } \alpha \in \{1, \ldots, m\}.$$

Defining $B := \beta_{\text{TSP}}^2 |\mathcal{E}| / (n^2 v^2 (1 - \varrho)^2)$, Eq. (12) can be written as

$$D(SQ) \le Bm \left(\sum_{i=1}^m \sqrt{c_i \lambda_i} \right)^2 .$$

Next, the lower bound in Eq. (5) is

$$D^* \ge \frac{B}{2} \sum_{i=1}^m \left(c_i + 2 \sum_{j=i+1}^m c_j \right) \lambda_i \ge \frac{B}{2} \sum_{i=1}^m (c_i \lambda_i) .$$

Thus, comparing the upper and lower bounds

$$\frac{D(SQ)}{D^*} \le 2m \frac{\left(\sum_{i=1}^m \sqrt{c_i \lambda_i} \right)^2}{\sum_{i=1}^m (c_i \lambda_i)} . \quad (13)$$

Letting $x_i := \sqrt{c_i \lambda_i}$, and $\mathbf{x} := [x_1, \ldots, x_m]$, the numerator of the fraction in Eq. (13) is $\|\mathbf{x}\|_1^2$, and the denominator is $\|\mathbf{x}\|_2^2$. But the one- and two-norms of a vector $\mathbf{x} \in \mathbb{R}^m$ satisfy $\|\mathbf{x}\|_1 \le \sqrt{m} \|\mathbf{x}\|_2$. Thus, in heavy load we obtain

$$\frac{D(SQ)}{D^*} \le 2m \left(\frac{\|\mathbf{x}\|_1}{\|\mathbf{x}\|_2} \right)^2 \le 2m^2 ,$$

and the policy is a $2m^2$-factor approximation. ∎

Remark IV.4 (Relation to RP policy in [10]) For $m = 2$ the SQ policy is within a factor of 8 of the optimal. This improves on the factor of 12 obtained for the Randomized Priority (RP) policy in [10]. However, it appears that the RP policy bound is not tight, since for two classes, simulations indicate it performs no worse than the SQ policy. □

V. SIMULATIONS AND DISCUSSION

In this section we discuss, through the use of simulations, the performance of the SQ policy with the probability assignment $p_\alpha := c_\alpha$, $\alpha \in \{1, \ldots, m\}$. In particular, we study (i) the tightness of the upper bound in Eq. (12), (ii) conditions for which the gap between lower bound in Eq. (5) and upper bound in Eq. (12) is maximized, (iii) the suboptimality of the probability assignment $p_\alpha = c_\alpha$, and, finally, (iv) how different the cost function in Eq. (2) may be, in general, for the SQ policy and a policy that services demands all together irrespective of priorities. Simulations of the SQ policy were performed using `linkern` [1] as a solver to generate approximations to the optimal TSP tour.

A. Tightness of the Upper Bound

We consider $n = 1$, $m = 4$, and $\varrho = 0.75, 0.8, 0.85, 0.9$ and 0.95. For each value of ϱ we perform 100 runs. In each run we randomly (uniformly in $[0, 1]$) generate $\lambda_1, \ldots, \lambda_m$, $c_1, \ldots, c_m$, and $\bar{s}_1, \ldots, \bar{s}_m$, with the constraints $\sum_{\alpha=1}^m \lambda_\alpha \bar{s}_\alpha = \varrho$ and $\sum_{\alpha=1}^m c_\alpha = 1$, we iterate the SQ policy 4000 times, and compute the value of cost function in Eq. (2) by considering the number of demands in the last 1000 iterations. For each value of ϱ, we record the mean value, standard deviation, maximum value and minimum value of the ratio (that we call χ) between experimental results and theoretical upper bound. This is summarized in Table I. One can see that the upper bound provides a reasonable approximation for load factors as low as $\varrho = 0.75$.

Load factor (ϱ)	$\mathbb{E}[\chi]$	σ_χ	max χ	min χ
0.75	0.803	0.092	1.093	0.354
0.8	0.778	0.108	0.943	0.256
0.85	0.773	0.111	1.150	0.417
0.9	0.733	0.159	1.162	0.203
0.95	0.716	0.131	0.890	0.257

TABLE I

RATIO χ BETWEEN EXPERIMENTAL RESULTS AND UPPER BOUND FOR VARIOUS VALUES OF ϱ.

B. Unfavorable Conditions for the SQ Policy

One may question if for some sets $\{\lambda_\alpha\}$ and $\{c_\alpha\}$, $\alpha \in \{1, \ldots, m\}$, the ratio between upper bound (12) and lower bound (5) is indeed close to $2m^2$. The answer is affirmative: consider, e.g., the case $\lambda_1 \ll \lambda_2 \ll \ldots \ll \lambda_m$ and $c_1 \gg c_2 \gg \ldots \gg c_m$, with $\lambda_\alpha c_\alpha = a$, for some positive constant a. Then, the upper bound is equal to $Bm^3 a$ and the lower
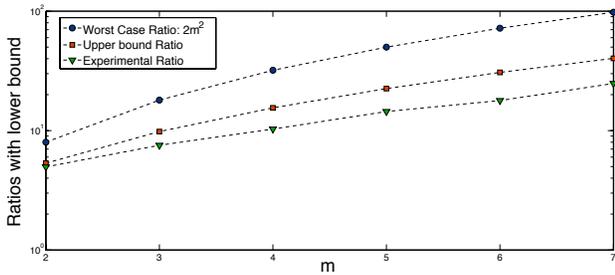
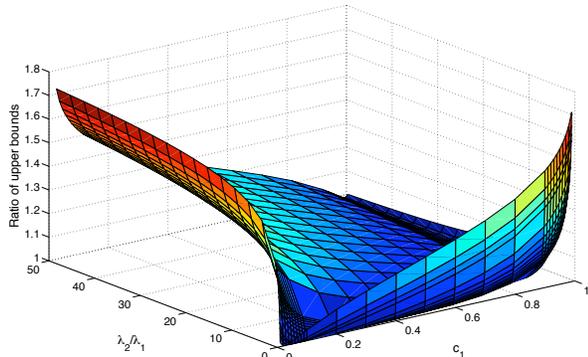Fig. 2. Experimental results for the SQ policy in worst-case conditions; $\varrho = 0.85$ and $\lambda_1 = 1$.

Fig. 3. The ratios $\text{upbd}_c/\text{upbd}_{opt}$ for 2 queues.

bound is approximately equal to $Bma/2$, thus their ratio is (arbitrarily) close to $2m^2$. Then, we simulated the SQ policy for the case $\lambda_m = a\lambda_{m-1} = a^2\lambda_{m-1} = \ldots = a^{m-1}\lambda_1$ and $c_1 = ac_2 = \ldots = a^{m-1}c_m$ with $a = 2$. Fig. 2 shows that the experimental value of the cost function (averaged over 10 simulation runs) indeed increases proportionally to m^2.

C. Suboptimality of Choice $p_\alpha = c_\alpha$

To prove Theorem IV.3 we used the probability assignment

$$p_\alpha := c_\alpha \quad \text{for each } \alpha \in \{1, \ldots, m\}. \quad (14)$$

However, one would like to select the $\mathbf{p}$ that minimizes the right-hand side of Eq. (12). This is a constrained multi-variable nonlinear optimization problem over $\mathbf{p}$. However, for two classes the optimization is over a single variable p_1. A comparison of optimized upper bound (upbd_{opt}) to the upper bound (upbd_c) obtained with the probability assignment in Eq. (14) is shown in Fig. 3. In this figure the ratio of upper bounds is bounded by two.

For $m > 2$ we make the comparison as follows. For each value of m we perform 1000 runs. In each run we randomly generate $\lambda_1, \ldots, \lambda_m$, $c_1, \ldots, c_m$, and five sets of initial $\mathbf{p}$ values, $\mathbf{p}_1, \ldots, \mathbf{p}_5$. For each initial condition we use a line search to find a local optimum $\mathbf{p}$ value. We take the ratio between upbd_c and the least upper bound upbd_{opt} from the five initial conditions. We also record the % variation between the largest and the smallest values of upbd_{opt} obtained starting from the five initial conditions. This is summarized in Table II. The second column shows the largest ratio obtained over the 1000 runs. The third column shows the largest % variation in the 1000 runs. The

Number of classes (m)	max $\text{upbd}_c/\text{upbd}_{opt}$	Max. % variation
3	1.60	0.12
4	1.51	0.04
5	1.51	0.08
6	1.74	0.02
7	1.88	0.08
8	1.63	0.15

TABLE II

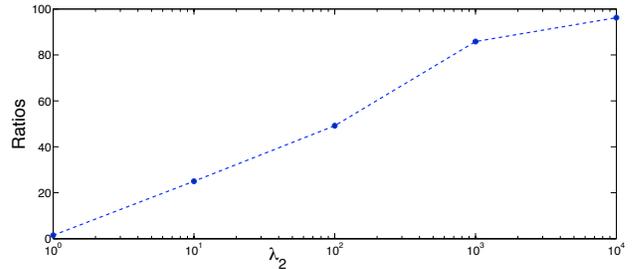RATIO OF UPPER BOUND WITH $p_\alpha = c_\alpha$ AND UPPER BOUND WITH OPTIMIZED $\mathbf{p}$.

Fig. 4. Ratio of experimental delays between Merge policy and SQ policy as a function of λ_2, with $m = 2$, $\lambda_1 = 1$, c = 0.995 and $\varrho = 0.9$.

assignment in Eq. (14) seems to perform within a factor of two of the optimized assignment, and the optimization appears to converge to values close to a global optimum since all five random conditions converge to values that are within $\sim 0.1\%$ of each other on every run.

D. The Merge Policy

The simplest possible policy for our problem would be to ignore priorities and service demands all together, by repeatedly forming TSP tours of outstanding demands (i.e., by using the SQ policy as though there were only one class). We call such a policy the Merge policy. However, the performance of the SQ and the Merge policy can be arbitrarily far apart. Indeed, by defining the overall arrival rate $\Lambda := \sum_{\alpha=1}^{m} \lambda_\alpha$ and overall mean on-site service $\bar{S} := \sum_{\alpha=1}^{m} \lambda_\alpha$, and by using the upper bounds in [4], we immediately obtain as an upper bound for the Merge policy: $D(\text{Merge}) \leq \frac{\beta_{\text{TSP}}^2 |\mathcal{E}|\Lambda}{n^2 v^2 (1-\varrho)^2}$. Then, we see that $D(\text{Merge})/D(SQ)$ can be arbitrarily large by choosing $\lambda_m \gg \lambda_\alpha$ and $c_m \ll c_\alpha$, with $\alpha \in \{1, \ldots, m-1\}$. This behavior is confirmed by experimental results, as depicted in Fig. 4 where we show the experimental ratios of delays between Merge and SQ policy (the ratios are averaged values over 10 simulation runs).

VI. CONCLUSIONS

In this paper we introduced a dynamic multi-vehicle routing problem with multiple classes of demands. For every set of coefficients, we determined a lower bound on the achievable convex combination of the class delays. We presented the Separate Queues (SQ) policy and showed that its deviation from the lower bound depends only on the number of the classes. We believe that there is room for improvement in the lower bound, and thus the SQ policy's performance may be significantly better than is indicated by its deviation from the current lower bound. Thus, our main

thrust of future work will be in trying to raise the lower bound. We are also interested in combining the aspects of multi-class vehicle routing with problems in which demands require teams of vehicles for their service.

ACKNOWLEDGMENTS

REFERENCES

[1] L. Kleinrock, *Queueing Systems. Volume II: Computer Applications.* New York, NY: John Wiley and Sons, 1976.

[2] E. G. Coffman Jr. and I. Mitrani, "A characterization of waiting time performance realizable by single-server queues," *Operations Research*, vol. 28, no. 3, pp. 810–821, 1980.

[3] D. Bertsimas, I. C. Paschalidis, and J. N. Tsitsiklis, "Optimization of multiclass queueing networks: Polyhedral and nonlinear characterizations of achievable performance," *The Annals of Applied Probability*, vol. 4, no. 1, pp. 43–75, 1994.

[4] D. J. Bertsimas and G. J. van Ryzin, "A stochastic and dynamic vehicle routing problem in the Euclidean plane," *Operations Research*, vol. 39, pp. 601–615, 1991.

[5] ——, "Stochastic and dynamic vehicle routing in the Euclidean plane with multiple capacitated vehicles," *Operations Research*, vol. 41, no. 1, pp. 60–76, 1993.

[6] ——, "Stochastic and dynamic vehicle routing with general interarrival and service time distributions," *Advances in Applied Probability*, vol. 25, pp. 947–978, 1993.

[7] E. Frazzoli and F. Bullo, "Decentralized algorithms for vehicle routing in a stochastic time-varying environment," in *Proc CDC*, Paradise Island, Bahamas, Dec. 2004, pp. 3357–3363.

[8] M. Pavone, E. Frazzoli, and F. Bullo, "Decentralized algorithms for stochastic and dynamic vehicle routing with general target distribution," in *Proc CDC*, New Orleans, LA, Dec. 2007, pp. 4869–4874.

[9] R. C. Larson and A. R. Odoni, *Urban Operations Research.* Englewood Cliffs, NJ: Prentice Hall, 1981.

[10] S. L. Smith, M. Pavone, F. Bullo, and E. Frazzoli, "Dynamic vehicle routing with heterogeneous demands," in *Proc CDC*, Cancun, Mexico, Dec. 2008, to appear.

[11] J. M. Steele, "Probabilistic and worst case analyses of classical problems of combinatorial optimization in Euclidean space," *Mathematics of Operations Research*, vol. 15, no. 4, p. 749, 1990.

[12] G. Percus and O. C. Martin, "Finite size and dimensional dependence of the Euclidean traveling salesman problem," *Physical Review Letters*, vol. 76, no. 8, pp. 1188–1191, 1996.

[13] H. Xu, "Optimal policies for stochastic and dynamic vehicle routing problems," Dept. of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA., 1995.

APPENDIX

In this appendix we prove Theorem IV.2. Henceforth, we consider the relation "$\leq$" in $\mathbb{R}^m$ as the product order of m copies of $\mathbb{R}$ (in other words, given two vector $v, w \in \mathbb{R}^m$, we interpret $v \leq w$ component-wise).

Proof: [Proof of Theorem IV.2] Define $q_j := 1 - p_j$ and let $\hat{\lambda}_\alpha$ denote the arrival rate in region $R^{[k]}$. Thus $\hat{\lambda}_\alpha := \lambda_\alpha / n$ for each $\alpha \in \{1, \ldots, m\}$. Let $x(i) :=$

$(\bar{N}_{1,i}, \bar{N}_{2,i}, \ldots, \bar{N}_{m,i}) \in \mathbb{R}^m$ and define two matrices

$$A := \begin{bmatrix} \hat{\lambda}_1 p_1 \bar{s}_1 + q_1 & \hat{\lambda}_1 p_2 \bar{s}_2 & \ldots & \hat{\lambda}_1 p_m \bar{s}_m \\ \hat{\lambda}_2 p_1 \bar{s}_1 & \hat{\lambda}_2 p_2 \bar{s}_2 + q_2 & \ldots & \hat{\lambda}_2 p_m \bar{s}_m \\ \vdots & & \ddots & \vdots \\ \hat{\lambda}_m p_1 \bar{s}_1 & \hat{\lambda}_m p_2 \bar{s}_2 & \ldots & \hat{\lambda}_m p_m \bar{s}_m + q_m \end{bmatrix},$$

and

$$B := \frac{\beta_{\text{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{nv}} \begin{bmatrix} \hat{\lambda}_1 p_1 & \hat{\lambda}_1 p_2 & \ldots & \hat{\lambda}_1 p_m \\ \hat{\lambda}_2 p_1 & \hat{\lambda}_2 p_2 & \ldots & \hat{\lambda}_2 p_m \\ \vdots & & \ddots & \vdots \\ \hat{\lambda}_m p_1 & \hat{\lambda}_m p_2 & \ldots & \hat{\lambda}_m p_m \end{bmatrix},$$

Then Eqs. (11) can be written as

$$x(i+1) \leq Ax(i) + B \begin{bmatrix} \sqrt{x_1(i)} \\ \sqrt{x_2(i)} \\ \vdots \\ \sqrt{x_m(i)} \end{bmatrix} =: f(x(i)) \qquad (15)$$

where $f : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$, and $x_j(i)$, $j \in \{1, \ldots, m\}$, are the components of vector $x(i)$. We refer to the discrete system in Eq. (15) as System-X. Next we define two auxiliary systems, System-Y and System-Z. We define System-Y as

$$y(i+1) = f(y(i)). \qquad (16)$$

System-Y is, therefore, equal to System-X, with the exception that we replaced the inequality with an equality.

Pick, now, any $\varepsilon > 0$. From Young's inequality

$$\sqrt{a} \leq \frac{1}{4\varepsilon} + \varepsilon a, \quad \text{for all } a \in \mathbb{R}_{\geq 0}. \qquad (17)$$

Hence, for $i \mapsto y(i) \in \mathbb{R}_{\geq 0}^m$, the Eq. (16) becomes

$$y(i+1) \leq Ay(i) + B\left(\frac{1}{4\varepsilon}\mathbf{1}_m + \varepsilon\, y(i)\right)$$
$$= \left(A + \varepsilon B\right)y(i) + \frac{1}{4\varepsilon}B\mathbf{1}_m.$$

where $\mathbf{1}_m$ is the vector $(1, 1, \ldots, 1)^{\mathrm{T}} \in \mathbb{R}^m$. Next, define System-Z as

$$z(i+1) = \left(A + \varepsilon B\right)z(i) + \frac{1}{4\varepsilon}B\mathbf{1}_m =: g(z(i)). \qquad (18)$$

The proof now proceeds as follows. First, we show that if $x(0) = y(0) = z(0)$, then

$$x(i) \leq y(i) \leq z(i), \quad \text{for all } i \geq 0 \qquad (19)$$

Second, we show that the trajectories of System-Z are bounded; this fact, together with Eq. (19), implies that also trajectories of System-Y and System-X are bounded. Third, and last, we will compute $\limsup_{i \to +\infty} y(i)$; this quantity, together with Eq. (19), will yield the desired result.

Let us consider the first issue. We have $y(1) = f(y(0))$ and $z(1) = g(z(0))$. Since, by assumption $z(0) = y(0)$, we have that $g(z(0)) = g(y(0)) \geq f(y(0))$, where the last inequality follows from Eq. (17) and by definition of f and g. Therefore, we get $y(1) \leq z(1)$. Then, we have $y(2) = f(y(1))$ and $z(2) = g(z(1))$. Since $z(1), y(1) \in$

$\mathbb{R}_{\geq 0}^m$, and the elements in matrices A and B are all non-negative, then $y(1) \leq z(1)$ implies $g(y(1)) \leq g(z(1))$. Using same arguments as before, we can write $z(2) \geq g(y(1)) \geq f(y(1)) = x(2)$; therefore, we get $y(2) \leq z(2)$. Then, it is immediate by induction that $y(i) \leq z(i)$ for all $i \geq 0$.

Similarly, we have $x(1) \leq f(x(0)) = f(y(0)) = y(1)$, where we have used the assumption $x(0) = y(0)$. Then, we get $x(1) \leq y(1)$. Since $x(1), y(1) \in \mathbb{R}_{\geq 0}^m$, the elements in matrices A and B are nonnegative, and by the monotonicity of $\sqrt{\cdot}$, then $x(1) \leq y(1)$ implies $f(x(1)) \leq f(y(1))$. Therefore, we can write $x(2) \leq f(x(1)) \leq f(y(1)) = y(2)$; thus, we get $x(2) \leq y(2)$. Then, it is immediate to show by induction that $x(i) \leq y(i)$ for all $i \geq 0$, and Eq. (19) holds.

We now turn our attention to the second issue, namely boundedness of trajectories for System-Z (in Eq. (18)). Notice that System-Z is a discrete-time linear system. The eigenvalues of A are characterized in the following lemma.

Lemma VI.1 The eigenvalues of A are real and with magnitude strictly less than 1 (i.e., A is a stable matrix).

Proof: Let $w \in \mathbb{C}^m$ be an eigenvector of A, and $\mu \in \mathbb{C}$ be the corresponding eigenvalue. Then we have $Aw = \mu w$. Define $r := (p_1 \bar{s}_1, p_2 \bar{s}_2, \ldots, p_m \bar{s}_m)$. Then the m eigenvalue equations are

$$\hat{\lambda}_j \, w \cdot r + q_j w_j = \mu \, w_j, \quad j \in \{1, \ldots, m\}, \qquad (20)$$

where $w \cdot r$ is the scalar product of vectors w and r, and w_j is the jth component of w.

There are two possible cases. If $w \cdot r = 0$, then Eq. (20) becomes $q_j w_j = \mu w_j$, for all j. Since $w \neq 0$, there exists j^* such that $w_j^* \neq 0$; thus, we have $\mu = q_{j^*}$. Since $q_{j^*} \in \mathbb{R}$ and $0 < q_{j^*} < 1$, we have that μ is real and $|\mu| < 1$.

Assume, now, that $w \cdot r \neq 0$. This implies that $\mu \neq q_j$ and $w_j \neq 0$ for all j, thus we can write for all j

$$w_j = \frac{\hat{\lambda}_j}{\mu - q_j} \, w \cdot r \qquad (21)$$

Therefore

$$w_j = \frac{\hat{\lambda}_j}{\hat{\lambda}_1} \frac{\mu - q_1}{\mu - q_j} w_1.$$

Therefore, (21) can be rewritten as

$$\sum_{j=1}^m \frac{r_j \hat{\lambda}_j}{\mu - q_j} = 1. \qquad (22)$$

Eq. (22) implies that the eigenvalues are real. To see this, write $\mu = a + ib$, where i is the imaginary unit: then

$$\sum_{j=1}^m \frac{r_j \hat{\lambda}_j}{a + ib - q_j} = \sum_{j=1}^m \frac{r_j \hat{\lambda}_j [(a - q_j) - ib]}{(a - q_j)^2 + b^2}$$

Thus Eq. (22) implies

$$b \underbrace{\sum_{j=1}^m \frac{r_j \hat{\lambda}_j}{(a - q_j)^2 + b^2}}_{>0} = 0$$

that is, $b = 0$. Eq. (22) also implies that the eigenvalues (that are real) have magnitude strictly less than 1. Indeed, assume, by contradiction, that $\mu \geq 1$, then we would have $\mu - q_j \geq 1 - q_j > 0$ (recall that the eigenvalues are real and $0 < q_j < 1$) and we could write

$$\sum_{j=1}^m \frac{r_j \hat{\lambda}_j}{\mu - q_j} \leq \sum_{j=1}^m \frac{r_j \hat{\lambda}_j}{1 - q_j} = \sum_{j=1}^m \bar{s}_j \hat{\lambda}_j = \varrho < 1,$$

and we get a contradiction. Assume, again by contradiction, that $\mu \leq -1$, then we would trivially get another contradiction $\sum_{j=1}^m r_j \hat{\lambda}_j / (\mu - q_j) < 0$, since $\mu - q_j < 0$. ∎

Hence, $A \in \mathbb{R}^{m \times m}$ has eigenvalues strictly inside the unit disk, and since the eigenvalues of a matrix depend continuously on the matrix entries, there exists a sufficiently small $\varepsilon > 0$ such that the matrix $A + \varepsilon B$ has eigenvalues strictly inside the unit disk. Accordingly, each solution $i \mapsto z(i) \in \mathbb{R}_{\geq 0}^m$ of System-Z converges exponentially fast to the unique equilibrium point

$$z^* = \left(I_m - A - \varepsilon B \right)^{-1} \frac{1}{4\varepsilon} B \mathbf{1}_m. \qquad (23)$$

Combining Eq. (19) with the previous statement, we see that the solutions $i \mapsto x(i)$ and $i \mapsto y(i)$ are bounded. Thus

$$\limsup_{i \to +\infty} x(i) \leq \limsup_{i \to +\infty} y(i) < +\infty. \qquad (24)$$

Finally, we turn our attention to the third issue, namely the computation of $y := \limsup_{i \to +\infty} y(i)$. Taking the $\limsup$ of the left- and right-hand sides of Eq. (16), and noting that

$$\limsup_{i \to +\infty} \sqrt{y_\alpha(i)} = \sqrt{\limsup_{i \to +\infty} y_\alpha(i)} \quad \text{for } \alpha \in \{1, 2, \ldots, m\},$$

since $\sqrt{\cdot}$ is continuous and strictly monotone increasing on $\mathbb{R}_{>0}$, we obtain that

$$y_\alpha = (1 - p_\alpha) y_\alpha + \hat{\lambda}_\alpha \sum_{j=1}^m p_j \left(\frac{\beta_{\text{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{nv}} \sqrt{y_j} + \bar{s}_j y_j \right).$$

Rearranging we obtain

$$p_\alpha y_\alpha = \hat{\lambda}_\alpha \sum_{j=1}^m p_j \left(\frac{\beta_{\text{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{nv}} \sqrt{y_j} + \bar{s}_j y_j \right). \qquad (25)$$

Dividing $p_\alpha y_\alpha$ by $p_1 y_1$ we obtain

$$y_\alpha = \frac{\hat{\lambda}_\alpha p_1}{\hat{\lambda}_1 p_\alpha} y_1. \qquad (26)$$

Combining Eqs. (25) and (26), we obtain

$$p_1 y_1 = \varrho \, p_1 y_1 + \frac{\beta_{\text{TSP}} \sqrt{|\mathcal{E}|}}{\sqrt{nv}} \sqrt{p_1 \hat{\lambda}_1 y_1} \sum_{j=1}^m \sqrt{\hat{\lambda}_j p_j}$$

Thus, recalling that $\hat{\lambda}_\alpha = \lambda_\alpha / n$, we obtain

$$y_\alpha = \frac{\beta_{\text{TSP}}^2 |\mathcal{E}|}{n^3 v^2 (1 - \varrho^2)} \frac{\lambda_\alpha}{p_\alpha} \left(\sum_{j=1}^m \sqrt{\lambda_j p_j} \right)^2.$$

Noting that from Eq. (24), $\limsup_{i \to +\infty} N_{\alpha, i} \leq y_\alpha$, we obtain the desired result. ∎