A Stochastic and Dynamic Vehicle Routing Problem with Time Windows and Customer Impatience

M. Pavone · N. Bisnik · E. Frazzoli · V. Isler

Abstract In this paper, we study the problem of designing motion strategies for a team of mobile agents, required to fulfill request for on-site service in a given planar region. In our model, each service request is generated by a spatio-temporal stochastic process; once a service request has been generated, it remains active for a certain deterministic amount of time, and then expires. An active service request is fulfilled when one of the mobile agents visits the location of the request. Specific problems we investigate are the following: what is the minimum number of mobile agents needed to ensure that a certain fraction of service requests is fulfilled before expiration? What strategy should they use to ensure that this objective is attained? This problem can be viewed as the stochastic and dynamic version of the well-known vehicle routing problem with time windows. We also extend our analysis to the case in which the time service requests remain active is itself a random variable, describing customer impatience. The customers' impatience is only known to the mobile agents via prior statistics. In this case, it is desired to minimize the fraction of service requests missed because of impatience. Finally, we show how the routing strategies presented in the paper can be executed in a distributed fashion.

M. Pavone (✉) · E. Frazzoli
Laboratory for Information and Decision Systems,
and the Aeronautics and Astronautics Department,
Massachusetts Institute of Technology,
Cambridge, MA 02139, USA
e-mail: pavone@mit.edu

E. Frazzoli
e-mail: frazzoli@mit.edu

N. Bisnik · V. Isler
Rensselaer Polytechnic Institute, Troy, NY 12180, USA

N. Bisnik
e-mail: bisnin@cs.rpi.edu

V. Isler
e-mail: isler@cs.rpi.edu

1 Introduction

The vehicle routing problem with time windows (VRPTW) is an extension of the classic vehicle routing problem (VRP) and is defined as follows: given a set of depots, a homogeneous fleet of vehicles and a set of known demand locations, find a set of closed routes, originating and ending at the depots, that service all demands and minimize the travel cost; in addition, the service at each demand must start within an associated *time window*. All problem parameters, such as demand locations and time windows, are assumed to be known with certainty. Time windows constraints are indeed common in many applications, including bank deliveries, postal deliveries, grocery distribution, dial-a-ride service, bus routing, and repairmen scheduling. The VRPTW has generated significant research interest over the years (see, for example, [1–5]), resulting in major contributions in the area of combinatorial optimization. However, paralleling the observations in [6] about the VRP, the VRPTW, as a model for routing

problems, is static and deterministic, whereas many routing problems in practice are inherently *dynamic* and *stochastic*. In fact, requests for service often arrive sequentially in time, and these arrival epochs may be stochastic; moreover, locations of future demands may be unknown or known only probabilistically.

Optimization problems with these characteristics arise frequently in sensor network settings. As an example, imagine a sensor network composed of a large number of nodes is deployed over a vast field, for example to study the behaviors of elusive animals, or to detect suspicious activity in a protected region as, for example, home burglaries, or insurgents placing improvised explosive devices (IEDs). Typically, network nodes contain inexpensive sensors, such as motion detectors, which are susceptible to false alarms. In addition to the sensor network, suppose a team of unmanned aerial vehicles (UAVs) is also available, which are equipped with more sophisticated on-board sensors. Each time a sensor detects an event, a UAV is sent to the location to investigate the cause of the alarm, i.e., to verify the presence of the animal or the intruders. Then, the UAV mission control is a *continuous* process of collecting alarms, planning routes and sending UAVs; moreover, *timeliness* is paramount: should the UAV take too long to reach the location of the event, its cause may have already left the premises, and be hard to track. Another possible scenario is a sensor network in which sensors are triggered by external events, and then remain active to upload data to a UAV for a certain known amount of time. After this time expires, sensors return to a power-saving "sleep" mode.

1.1 The dynamic and stochastic VRPTW

In this paper, we introduce and study the following dynamic and stochastic version of the VRPTW. We assume that demands for service arrive according to a Poisson process with rate λ to a bounded Euclidean service region Q with area $|Q|$. Upon arrival, demands assume an independent and uniformly distributed location in Q. Each demand has an associated *deterministic* time window $[r_i, d_i]$, where $d_i \geq r_i, r_i, \ d_i \in \mathbb{R}$; the *release time* r_i is the demand's arrival time, while the *deadline* is $r_i + T$, where $T \in \mathbb{R}_+$ is a constant *independent* of the demand. Notice the assumption of equal width for the time windows (although release times and deadlines clearly differ from demand to demand). Demands are serviced by a fleet of vehicles that travel at constant velocity v. We assume that the on-site service times are equal to zero (i.e., we assume that the on-site service times are negligible compared to the travel times).

Given $\varepsilon \in (0, \ 1]$, the objective is to find routing policies that, with the minimum possible number of agents, ensure that each demand has a probability of being visited before expiring greater than $1 - \varepsilon$.

This problem can be viewed as an extension of our complementary previous works [7] and [8].

1.2 Related work

The static and deterministic VRPTW has been the subject of intensive research efforts for both heuristic and exact optimization approaches (see, for example, [1–5]). Indeed, the VRPTW is NP-hard; even finding a feasible solution to the VRPTW when the number of vehicles is fixed is itself an NP-complete problem [9]. Chapter 7 in [3] provides a comprehensive survey on exact (exponential-time) solution techniques. Because of the difficulty of the VRPTW and its wide applicability to real-life situations, many heuristic solution techniques capable of producing high-quality solutions in limited time have been proposed; a recent thorough survey on heuristics for the VRPTW can be found in [4, 5].

To the best of our knowledge, this is the first time that a dynamic and stochastic version of the VRPTW is introduced. The dynamic and stochastic VRPTW is closely related to the dynamic traveling repairman problem (DTRP) [6, 7, 10–12], in which m identical vehicles must service demands whose time of arrival, location and on-site service time are stochastic, and the objective is to find service policies that minimize the expected waiting time of the demands. In [8] the authors studied a similar vehicle routing problem where demands expire. They presented approximation algorithms for the case where vehicles' motion is restricted to a planar curve.

The dynamic and stochastic VRPTW is also related to coverage problems of sensor networks. Considerable research effort has been invested in studying coverage properties of static sensor networks [13–17]. More recently, there has been growing interest in understanding how the coverage properties of a sensor network may be improved by introducing mobility to the sensor devices. The problem of relocating sensors to improve coverage has been studied in [18]. In this formulation, the sensors can individually estimate the positions of the targets. However, the quality of coverage decreases with increasing distance. The average area covered by mobile sensors over a period of time has been characterized in [19]. It is shown that for a mobile sensor network with spatial density λ, with each sensor moving according to a mobility model similar to a random walk with expected velocity $E[V_s]$, the

expected area covered in time interval $(0, t)$ is given by $1 - \exp\left(-\lambda(\pi r^2 + 2rE[V_s]t)\right)$. Finally, another related problem is the orienteering problem (see, for example, [20]): the input is an edge-weighted graph $G = (V, E)$ (directed or undirected), two nodes $s, t \in V$ and a non-negative budget B. The goal is to find an $s - t$ walk of total length at most B so as to maximize the number of distinct nodes visited by the walk.

1.3 Statement of contributions and paper organization

Since we use a variety of results from several areas, we present a brief overview in Section 2. In Section 3 we formally define the dynamic and stochastic VRPTW. In Section 4 we study the special case when $\varepsilon \to 0^+$ and λ is large; setting $\varepsilon \to 0^+$ implies that almost all demands are required to be serviced before expiring. First, we compute a lower bound on the minimum number of vehicles needed to ensure that almost all demands are serviced before expiring; second, we propose and analyze a service policy whose performance provably approximates that of optimal policies. Then, in Section 5, we study a provably correct service policy for the general case with ε and λ arbitrary. In Section 6 we extend our analysis to the case in which the time service requests remain active is itself a random variable, describing customer impatience; the customers' impatience is only known to the mobile agents via prior statistics. In Section 7 we present simulation results. In Section 8 we present a distributed strategy for assigning demands to the vehicles, and to route them in an efficient way; finally, in Section 9, we draw some conclusions and discuss some directions for future work.

2 Preliminaries

In this section, we briefly describe some known concepts from probability, geometry and combinatorial optimization, on which we will rely extensively later in the paper.

2.1 Notation

Let $\mathbb{N}_0$ be the set of nonnegative integers and $\mathbb{R}_+$ be the set of positive real numbers. Let $\|\cdot\|$ denote the Euclidean norm. Let Q be a compact, convex subset of $\mathbb{R}^d$. We denote the Lebesgue measure of Q as $|Q|$. We define $J_m \doteq \{1, 2, \cdots, m\}$. Let $G = (g_1, \cdots, g_m) \in Q^m \subset (\mathbb{R}^d)^m$ denote the location of m points in Q. A *partition* (or tessellation) of Q is a collection of m closed subsets $\mathcal{Q} = \{Q_1, \cdots, Q_m\}$ with disjoint interiors whose

union is Q. The partition of Q is *convex*, if each Q_j, $j \in J_m$, is convex.

2.2 Convergence of random variables

We refer the reader to [21, 22] for comprehensive treatment of convergence of random variables. A sequence of random variables $X_r, r \in \mathbb{N}_0$, converges almost surely to X ($\lim_{r \to \infty} X_r \overset{a.s.}{=} X$) if $\lim_{r \to \infty} X_r(\omega) = X(\omega)$ for all sample functions $\omega \in \Omega$ where $\mathbb{P}[\Omega] = 1$. (In other words, $\mathbb{P}[\lim_{r \to \infty} X_r = X] = 1$.) The sequence of random variables X_r converges almost surely to X if and only if, for each $\varepsilon > 0$,

$$\lim_{r \to \infty} \mathbb{P}\left[\sup_{s \geq r}\{|X_s - X|\} > \varepsilon\right] = 0. \tag{1}$$

2.3 Inequalities for random variables

If X and Y are defined on the same probability space, then X is almost surely larger that Y, written $X \overset{a.s.}{\geq} Y$, if and only if $X(\omega) \geq Y(\omega)$ for all $\omega \in \Omega$ where $\mathbb{P}[\Omega] = 1$.

Moreover, we define the following notions for asymptotic almost sure inequalities. We say that a sequence of random variables X_r, $r \in \mathbb{N}_0$, is asymptotically almost surely upper bounded by the random variable X (which we write as $\lim_{r \to \infty} X_r \overset{a.s.}{\leq} X$) if, for any $\varepsilon > 0$,

$$\lim_{r \to \infty} \mathbb{P}\left[\sup_{s \geq r}\{X_s - X\} > \varepsilon\right] = 0. \tag{2}$$

Similarly, we say that X_r is asymptotically almost surely lower bounded by X (that is, $\lim_{r \to \infty} X_r \overset{a.s.}{\geq} X$) if, for any $\varepsilon > 0$,

$$\lim_{r \to \infty} \mathbb{P}\left[\sup_{s \geq r}\{X - X_s\} > \varepsilon\right] = 0. \tag{3}$$

Notice that neither Eq. 2 nor 3 implies any form of convergence for the sequence X_r. On the other hand, if both hold at the same time, it is clear that Eq. 1 is recovered.

2.4 Asymptotic and worst-case properties of the traveling salesperson problem in the Euclidean plane

The Euclidean traveling salesperson problem (TSP) is formulated as follows: given a set D of n points in $\mathbb{R}^d$, find the minimum-length tour (i.e., closed path

that visits all points) of D. Let TSP(D) denote the minimum length of a tour through all the points in D; by convention, TSP($\emptyset$) = 0. Suppose the set D is composed of n points whose locations are independently chosen at random from a distribution f, supported on a compact set Q. In [23] it is shown that there exists a constant $\beta_{\text{TSP},d}$ such that, almost surely,

$$\lim_{n \to +\infty} \frac{\text{TSP}(D)}{n^{1-1/d}} \overset{a.s.}{=} \beta_{\text{TSP},d} \int_Q \bar{f}(q)^{1-1/d}\, dq, \qquad (4)$$

where $\bar{f}$ is the density of the absolutely continuous part of the point distribution f. In other words, the optimal cost of stochastic TSP tours approaches a deterministic limit. In the planar case, the cost of stochastic TSP tours grows as the square root of the number of points in D. Moreover, the current best estimate of the constant in the case $d = 2$ is $\beta_{\text{TSP},2} \simeq 0.7120$ (see, e.g., [24]). For short, we let $\beta \doteq \beta_{\text{TSP},2}$.

Notice that the bound (4) holds for all compact sets: the shape of the set only affects the convergence rate to the limit. According to [25], if Q is a "fairly compact and fairly convex" set in the plane, then Eq. 4 provides an adequate estimate of the optimal TSP tour length for values of n as low as 15.

Remarkably, the asymptotic cost of the stochastic TSP for uniform point distributions is an upper bound on the asymptotic cost for general point distributions, i.e.,

$$\lim_{n \to +\infty} \frac{\text{TSP}(D)}{n^{1-1/d}} \leq \beta_{\text{TSP},d} |Q|^{1/d},$$

where $|Q|$ is the area of Q. This follows directly from an application of Jensen's inequality for concave functions to the right hand side of Eq. 4

$$\int_Q \bar{f}(q)^{1-\frac{1}{d}}\, dq \leq |Q|^{1/d} \left(\int_Q \bar{f}(q)\, dq \right)^{1-\frac{1}{d}} \leq |Q|^{1/d}.$$

Finally, if the support of the point distribution f is a compact set $Q \subseteq \mathbb{R}^d$, the following (deterministic) bound holds on the length of the TSP tour [26]:

$$\text{TSP}(D) \leq \beta_Q n^{1-1/d} |Q|^{1/d}. \qquad (5)$$

The price of determinism is that the constant β_Q depends on the set Q, and is generally much larger than $\beta_{\text{TSP},d}$. For example, if Q is a unit square, $\beta_Q = 2$.

2.5 Voronoi diagrams

We refer the reader to [27] for a comprehensive treatment of Voronoi diagrams. The *Voronoi Diagram* $\mathcal{V}(G) = (V_1(G), \cdots, V_m(G))$ of Q generated by points $G = (g_1, \cdots, g_m)$ is defined by

$$V_i(G) = \left\{ x \in Q \mid \|x - g_i\| \leq \|x - g_j\|,\ \forall j \neq i,\ j \in J_m \right\} \qquad (6)$$

We refer to G as the set of *generators* of $\mathcal{V}(G)$, and to $V_i(G)$ as the Voronoi cell or region of dominance of the i-th generator. Each Voronoi cell is a convex set. Indeed, a Voronoi Diagram is a convex partition of Q. For simplicity, we will refer to $V_i(G)$ as V_i. When the two Voronoi cells V_i and V_j are adjacent (i.e., they share an edge), g_i is called a *Voronoi neighbor* of g_j (and vice-versa).

Finally, we define an equitable Voronoi diagram as a Voronoi diagram where all Voronoi cells have the same Lebesgue measure, with respect to the distribution f.

2.6 The VRPTW

The classic VRPTW is defined on a complete graph $G = (V, A)$, where $V = \{v_0, v_1, \ldots, v_n\}$ is the vertex set and $A = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ is the arc set. The first p vertices in V represent depot locations at which one or more of the $m \geq p$ available vehicles are based, while the remaining vertices of V represent demand locations to be serviced. With each vertex $v_i \in V$ is associated a nonnegative on-site service time s_i, and a time window $[r_i, d_i]$, where $d_i \geq r_i, r_i, d_i \in \mathbb{N}_0$. We refer to r_i as a *release time*, to d_i as a *deadline*, and to $d_i - r_i$ as the *width* of the time window. Each arc (v_i, v_j) has an associated nonnegative cost c_{ij} (usually the travel time between v_i and v_j). The VRPTW consists of designing m vehicle routes on G such that: (1) every route starts and ends at the same depot; (2) every demand belongs to exactly one route; (3) the service at demand i begins in the interval $[r_i, d_i]$, and each vehicle j leaves its depot and returns to its depot in the interval $[r_j, d_j]$, (4) the total travel cost of all vehicles is minimized. There are variants of this problem where other constraints are added, for example the total load and duration of route j are required not to exceed certain thresholds. Moreover, the fleet size m can be a variable and a usual additional objective is to minimize m.

All problem parameters are assumed to be known with *certainty*. Note that finding a feasible solution to the VRPTW is itself an NP-complete problem [9].

3 Problem definition

We focus on the case where the environment Q is a compact, convex subset of $\mathbb{R}^2$.[1] Without loss of generality we will assume that the measure of Q (denoted as $|Q|$) is 1.

Demands are serviced by a team of m holonomic vehicles, modeled as point masses. The vehicles are free to move, with bounded velocity v, within the environment Q; without loss of generality, we will assume that the velocity magnitude is unitary. The vehicles are identical, and have unlimited fuel and demand servicing capacity. For simplicity, vehicles are not required to stop or to loiter in proximity of demands (i.e., we assume zero on-site service time).

Service operations start at time 0; the initial number of demands is a positive integer random variable. Demands arrive at Q according to a Poisson process with intensity λ, and their locations are independent and uniformly distributed over Q. In other words, the number of demands generated over time within a region $S \subseteq Q$ can be described as a homogeneous Poisson process with rate

$$\lambda_S = \lambda \cdot |S|.$$

When the number of vehicles is a function of λ, we denote such number as $m(\lambda)$. It is assumed that the initial number of demands is *independent* of λ. The term *heavy load* is used to denote the condition $\lambda \to \infty$. We will label demands in increasing order with respect to time of arrival, by using the index $i \in \mathbb{N}_0$. Each demand has an associated *deterministic* time window: the release time r_i is the demand's arrival time, while the deadline is $r_i + T$, where $T \in \mathbb{R}_+$ is a constant *independent* of the demand. Notice the simplifying assumption of equal width for the time windows (although release times and deadlines clearly differ from demand to demand).

If one of the vehicles visits the location of the i-th demand at a time t_i such that $r_i \leq t_i < d_i$, then the i-th demand is considered serviced; otherwise the i-th demand is considered expired. Let W_i be a random variable expressing the *sojourn* time in the system for the i-th demand. If the i-th demand is serviced, then $W_i = t_i - r_i$, otherwise $W_i = T$. Notice that, by definition, all random variables W_i are surely bounded below and above; in particular, we have $0 \leq W_i \leq T$.

Let $D(t)$ be the set of locations of demands generated up to time t. Information on outstanding demands (i.e., arrived demands that have neither been serviced nor expired) at time t is summarized as a finite set of demand positions $D_o(t) \subseteq D(t)$. Demands are inserted in both D and D_o as soon as they are generated; they are removed from D_o either upon servicing—as a vehicle visits the demand's location— or upon expiration. We assume that information contained in $D(t)$ and $D_o(t)$ is available to all vehicles.

Given $\varepsilon \in (0, 1]$, the objective is to find routing policies that ensure

$$\lim_{i \to \infty} \mathbb{P}[W_i < T] \geq 1 - \varepsilon, \qquad (7)$$

with the minimum possible number of agents. (Notice that $W_i = T$ if and only if the i-th demand expired.) We will refer to ε as the "reliability" of the system (notice that ε small implies high reliability).

Define W_i^λ as the sojourn time of the i-th demand when the arrival rate is λ. Given $\varepsilon \in (0, 1]$, the objective, *in heavy load*, is to find routing policies that ensure

$$\lim_{i \to \infty} \mathbb{P}\left[\lim_{\lambda \to \infty} W_i^\lambda < T\right] \geq 1 - \varepsilon. \qquad (8)$$

The heavy load assumption will allow us to find some interesting asymptotic results.

4 The dynamic and stochastic VRPTW in heavy load

In this section we study the special case when $\varepsilon \to 0^+$ and $\lambda \to \infty$. Setting $\varepsilon \to 0^+$ implies that, in steady state, almost all demands are required to be serviced before expiring. Our strategy is the following: First, we establish a heavy-load lower bound on the minimum number of agents required to ensure Eq. 8 when $\varepsilon \to 0^+$. Second, we analyze a heavy-load policy with a provable performance guarantee.

4.1 Heavy-load lower bound

We have the following

Theorem 4.1 *The minimum number $m^*(\lambda)$ of vehicles required to ensure Eq.* 8 *when $\varepsilon \to 0^+$ satisfies:*

$$\lim_{\lambda \to \infty} \frac{m^*(\lambda)}{\sqrt{\lambda}} \geq \frac{\gamma}{\sqrt{T}}, \qquad (9)$$

where $\gamma \geq 2/(3\sqrt{2\pi}) \approx 0.266$.

---

[1]Extensions to higher dimensions are in principle straightforward, but the constants appearing, e.g., in Eq. 4, are less well known.

Proof When $\varepsilon \to 0^+$, we require that, in the limit $i \to \infty$, all demands (except possibly a negligible set) receive service before expiration. Thus, in the limit $i \to \infty$, our problem resembles the dynamic vehicle routing problem [6]. In [6] it is shown that, for any policy, the limiting expected sojourn time satisfies (assuming zero on-site service time):

$$\lim_{i \to \infty} \frac{\mathbb{E}\left[W_{i,\text{DTRP}}^{\lambda}\right] m^2(\lambda)}{\lambda} \geq \gamma^2,$$

where $W_{i,\text{DTRP}}^{\lambda}$ is the sojourn time of the *i*-th demand in the DTRP when the arrival rate is λ, and $\gamma \geq 2/(3\sqrt{2\pi}) \approx 0.266$. Therefore, in heavy load

$$\lim_{\lambda \to \infty} \lim_{i \to \infty} \frac{\mathbb{E}\left[W_{i,\text{DTRP}}^{\lambda}\right] m^2(\lambda)}{\lambda} \geq \gamma^2.$$

Since all demands, except possibly a negligible set, receive service within a time T, we have eventually $\mathbb{E}\left[W_{i,\text{DTRP}}^{\lambda}\right] \leq T$. It follows that

$$\lim_{\lambda \to \infty} \lim_{i \to \infty} \frac{T m^2(\lambda)}{\lambda} \geq \gamma^2. \tag{10}$$

Therefore, to satisfy Eq. 10, we need

$$\lim_{\lambda \to \infty} \frac{m(\lambda)}{\sqrt{\lambda}} \geq \frac{\gamma}{\sqrt{T}}. \qquad\qquad \square$$

Thus, the required number of vehicles in heavy load and when $\varepsilon \to 0^+$ can not be less then $m^*(\lambda) \doteq \left\lceil \gamma \sqrt{\lambda/T} \right\rceil$.

4.2 A provably good heavy load policy

In this section, we propose a policy (which we call TSP Policy) that satisfies Eq. 8 and requires a number of vehicles that is within a constant factor from the optimum. A pseudo-code description of the TSP Policy is provided in Algorithm 1. The requirement that the location of each demand is visited even if the demand expired is introduced to simplify the analysis.

Algorithm 1 TSP Policy
Require: m, the number of vehicles.
Ensure: Routing policy to service demands.
At start-up, the environment Q is partitioned into m service regions Q_j, $j \in J_m$, of equal area $1/m$ (recall that $|Q| = 1$), and each agent is assigned to a distinct service region. Then, each agent executes in its own service region:
while TRUE **do**
  if there are no unvisited demands **then**
    Move at unit velocity toward the median of the service region.
  else
    Compute the TSP tour through all demands. Service demands by following the TSP tour (start from the closest demand and randomly select one of the two possible orientations of the tour).
    Do not skip demands that expired.
  end if
end while

We assume that each vehicle spends a fixed time $c > 0$, $c \ll T$, to compute a new TSP tour; the computation time c *does not* depend on the number of outstanding demands.[2]

The behavior of the TSP policy is summarized in the following theorem.

Theorem 4.2 *Assume that the TSP policy is used with a number of vehicles $m(\lambda)$ (i.e., m depends on λ) that satisfies*

$$\beta\sqrt{\frac{2}{T}} < \lim_{\lambda \to \infty} \frac{m(\lambda)}{\sqrt{\lambda}} < +\infty. \tag{11}$$

Then

$$\lim_{i \to \infty} \mathbb{P}\left[\lim_{\lambda \to \infty} W_i^{\lambda} < T \right] = 1.$$

In other words, the TSP policy, with a number of vehicles that satisfies Eq. 11, solves, in heavy load and when

---

[2]The assumption of a *small* and *constant* computation time is indeed common in the DTRP literature; e.g., in [6], the computation time is assumed to be zero.

$\varepsilon \to 0^+$, *the dynamic and stochastic VRPTW. Moreover, if we define, with $\delta > 0$ an arbitrarily small constant,*

$$m_{\text{TSP}}(\lambda) \doteq \left\lceil \sqrt{\lambda} \left(\beta \sqrt{\frac{2}{T}} + \delta \right) \right\rceil \quad (12)$$

($m_{\text{TSP}}(\lambda)$ *clearly satisfies Eq. 11*) *we obtain*

$$\lim_{\lambda \to \infty} \frac{m_{\text{TSP}}(\lambda)}{m^*(\lambda)} \le 3.78. \quad (13)$$

Equation 13 shows that the number of vehicles required by the TSP policy is within a constant factor from the optimal value.

Before proving Theorem (4.2), we need some notation and two lemmas.

To simplify the analysis, we write $\lambda = kl$ where $k \in \mathbb{N}_0$ and $l > 0$ is an arbitrary constant: thus, heavy load is obtained for $k \to \infty$. With a slight abuse of notation, let $W_{i,j}^\lambda$ be the sojourn time of the i-th demand arriving in service region j. By definition of the policy, what happens in a service region $j \in J_m$ is independent of what happens in any other service region $h \ne j$. Thus, under the TSP policy, ensuring Eq. 8 when $\varepsilon \to 0^+$ calls for ensuring that

$$\lim_{i \to \infty} \mathbb{P}\left[\lim_{\lambda \to \infty} W_{i,j}^\lambda < T \right] = 1, \ \forall j \in J_m.$$

Our analysis will thus concentrate on the (independent) sequences of random variables $W_{i,j}^\lambda$. In the following, to avoid cumbersome notation, we will drop the index j, with the understanding that all the following analysis refers to a service region j (and *not* to the whole environment Q).

The arrival rate in a service region of area $1/m(\lambda)$, is $\bar{\lambda} = \lambda/m(\lambda)$. We refer to the time instant t_r, $r \in \mathbb{N}_0$, at which the vehicle computes a new TSP tour (recall that each service region is serviced by exactly one vehicle) as the epoch r of the policy; we refer to the time interval between epoch r and epoch $r + 1$ as the r-th iteration. Finally, we define

- n_r^k, $r \ge 1$: number of demands arrived in between epochs $r - 1$ and r, when the overall arrival process has intensity $\lambda = kl$;
- C_r^k, $r \ge 0$: time interval between epochs r and $r + 1$, when the overall arrival process has intensity $\lambda = kl$. This time interval is the sum of (1) the computation time (equal, deterministically, to c time units), and of (2) the time required to service all demands

arrived in between epochs $r - 1$ and r following a TSP tour;
- W_i^k, $i \ge 0$: sojourn time of the i-th demand, when the overall arrival process has intensity $\lambda = kl$;
- $m(k)$: number of vehicles used when $\lambda = kl$.

Notice that $C_r^k - c$ is also the length of the TSP tour through the demands arrived in between epochs $r - 1$ and r; indeed, velocity is unitary, and in heavy load we can safely neglect the travel component between the agent's current position and the closest demand in the TSP tour [see also part (1) in Lemma 4.3 below].

In the next Lemma, we provide some almost sure convergence results concerning random variables C_r^k and n_r^k.

Lemma 4.3 *Assume*

$$\beta \sqrt{\frac{2}{T}} < \lim_{k \to \infty} \frac{m(k)}{\sqrt{kl}} < +\infty.$$

Then, at each epoch $r \ge 1$:

(1) $\quad \lim_{k \to \infty} n_r^k \overset{a.s}{=} \infty;$

(2) $\quad \lim_{k \to \infty} \frac{C_r^k - c}{\sqrt{n_r^k}} \sqrt{m(k)} \overset{a.s}{=} \beta;$

(3) $\quad \lim_{k \to \infty} \frac{n_r^k}{C_{r-1}^k} \frac{1}{kl/m(k)} \overset{a.s}{=} 1.$

Indeed, limit (1) is intuitive; limit (2) is an application of the TSP Theorem (Eq. 4), and limit (3) is a consequence of the strong law for renewal processes. A detailed proof is provided in the Appendix.

The next Lemma characterizes the length of an iteration.

Lemma 4.4 *Assume*

$$\beta \sqrt{\frac{2}{T}} < \lim_{k \to \infty} \frac{m(k)}{\sqrt{kl}} < +\infty.$$

Then

$$\limsup_{r \to \infty} \limsup_{k \to \infty} C_r^k \overset{a.s.}{<} T/2.$$

Proof Recall that $C_{r+1}^k - c$ is the length of the TSP tour through the demands arrived in between epochs r and $r + 1$. Then, using Lemma 4.3, we can write

$$
\begin{aligned}
\limsup_{k \to \infty} C_{r+1}^k &= c + \limsup_{k \to \infty} \left(C_{r+1}^k - c \right) \\
&= c + \limsup_{k \to \infty} \frac{C_{r+1}^k - c}{\sqrt{n_{r+1}^k}} \frac{\sqrt{m(k)}}{\sqrt{m(k)}} \sqrt{n_{r+1}^k} \\
&\overset{a.s}{=} c + \limsup_{k \to \infty} \frac{\beta}{\sqrt{m(k)}} \sqrt{n_{r+1}^k} \\
&= c + \limsup_{k \to \infty} \frac{\beta}{\sqrt{m(k)}} \sqrt{\frac{n_{r+1}^k}{C_r^k} \frac{kl/m(k)}{kl/m(k)} C_r^k} \\
&\overset{a.s.}{=} c + \limsup_{k \to \infty} \beta \frac{\sqrt{kl}}{m(k)} \sqrt{C_r^k} \\
&< c + \sqrt{\frac{T}{2}} \limsup_{k \to \infty} \sqrt{C_r^k}.
\end{aligned}
\tag{14}
$$

Define the random variable $X_r \doteq \limsup_{k \to \infty} C_r^k$; Eq. 14 allows to determine an almost sure upper bound on $\limsup_{r \to \infty} X_r$. It is straightforward to verify

$$
\limsup_{r \to \infty} X_r \overset{a.s.}{<} \frac{1}{4} \left(\sqrt{\frac{T}{2}} + \sqrt{\frac{T}{2} + 4c} \right)^2 ;
\tag{15}
$$

since, by assumption, $c \ll T$, we get the claim. □

Proof (Theorem 4.2) Notice that, for demands arriving in between epochs r and $r + 1$, the sojourn time W_i^k is bounded above by $(C_r^k + C_{r+1}^k)$; therefore, applying Lemma 4.4, we can write

$$
\limsup_{i \to \infty} \limsup_{k \to \infty} W_i^k \leq \limsup_{r \to \infty} \limsup_{k \to \infty} \left(C_r^k + C_{r+1}^k \right) \overset{a.s.}{<} T.
\tag{16}
$$

For convenience, define $\bar{T}$ as

$$
\bar{T} \doteq \limsup_{i \to \infty} \limsup_{k \to \infty} W_i^k.
$$

From the definition, $\bar{T}$ is a random variable (it might not be a constant) almost surely smaller than T. Recalling the definition of almost sure convergence, we have for any $\delta > 0$

$$
\lim_{i \to \infty} \mathbb{P} \left[\sup_{p \geq i} \left\{ \left| \sup_{q \geq p} \left(\limsup_{k \to \infty} W_q^k \right) - \bar{T} \right| \right\} > \delta \right] = 0.
$$

Thus, we have *a fortiori*

$$
\lim_{i \to \infty} \mathbb{P} \left[\left(\limsup_{k \to \infty} W_i^k - \bar{T} \right) > \delta \right] = 0.
$$

The claim follows from the fact that δ is arbitrary and $\bar{T} \overset{a.s.}{<} T$. □

5 A service policy for the non-asymptotic case

The previous results hold only in the limit $\lambda \to \infty$ and when $\varepsilon \to 0^+$. In this section, we study the TSP policy for arbitrary values of $\varepsilon \in (0, 1]$ and in general load conditions. In particular, we study the number of vehicles that are sufficient, for a given ε, to ensure Eq. 7. Here, we assume that vehicles are allowed to skip the expired demands. We have the following

Theorem 5.1 *For any given $\varepsilon \in (0, 1]$ and any λ, a number of vehicles sufficient for the TSP policy to ensure Eq. 7 is*

$$
m = \left\lceil \beta_Q \sqrt{\frac{2\lambda}{\varepsilon T}} \right\rceil,
$$

where β_Q is a constant that depends on the shape *of the service regions.*

Proof As in Section 4.2, we analyze the sequence of sojourn times within a given cell $j \in J_m$, and we drop the index j, with the understanding that all the following analysis refers to a service region j (and *not* to the whole environment Q).

To avoid any confusion, we restate some of the notation. The arrival rate in a service region, whose area is $1/m$, is $\bar{\lambda} = \lambda/m$. We refer to the time instant t_r, $r \in \mathbb{N}_0$, in which the vehicle computes a new TSP tour as the epoch r of the policy; we refer to the time interval between epoch r and epoch $r + 1$ as the r-th iteration. We define n_r as the number of demands arrived in between epochs $r - 1$ and r. Finally, we define C_r as the time interval between epochs r and $r + 1$. For simplicity we assume the computation time to be zero; extension to the case $c > 0$ is straightforward but cumbersome.

First, we study the sequence of expected values $\mathbb{E}[C_r]$. By the deterministic inequality for the TSP tour through n points, we have (recall that the area of each service region is $1/m$)

$$
C_{r+1} \leq \beta_Q \sqrt{n_{r+1}/m},
$$

where the contribution of the initial travel to the closest demand in the TSP tour is included in β_Q. By applying Jensen's inequality for concave functions, in the form $\mathbb{E}\left[\sqrt{X}\right] \leq \sqrt{\mathbb{E}[X]}$, we get

$$\mathbb{E}[C_{r+1}] \leq \beta_Q \sqrt{\mathbb{E}[n_{r+1}]/m}.$$

During iteration r of the policy, demands arrive according to a Poisson process. Call n_r^{new} the number of demands arrived during iteration r. Thus, we have $\mathbb{E}[n_{r+1}] = \mathbb{E}[n_r^{\text{new}}] = \bar{\lambda}\mathbb{E}[C_r]$; therefore

$$\mathbb{E}[C_{r+1}] \leq \beta_Q \sqrt{\bar{\lambda}\mathbb{E}[C_r]/m}.$$

The above inequality describes a recurrence relation that allows to bound the value to which $\mathbb{E}[C_r]$ converges as $r \to \infty$. It is straightforward to verify that

$$\limsup_{r\to\infty} \mathbb{E}[C_r] \leq \beta_Q^2 \frac{\lambda}{m^2}.$$

Since, for demands arriving in between epochs r and $r + 1$, the sojourn time W_i is bounded above by $(C_r + C_{r+1})$, we can write

$$\limsup_{i\to\infty} \mathbb{E}[W_i] \leq 2\beta_Q^2 \frac{\lambda}{m^2}.$$

Furthermore, since W_i is a non-negative random variable with finite mean, we can apply Markov inequality to obtain

$$\mathbb{P}[W_i = T] = \mathbb{P}[W_i \geq T] \leq \frac{\mathbb{E}[W_i]}{T};$$

therefore we have

$$\limsup_{i\to\infty} \mathbb{P}[W_i = T] \leq \limsup_{i\to\infty} \frac{\mathbb{E}[W_i]}{T} \leq 2\beta_Q^2 \frac{\lambda}{m^2 T}.$$

If we choose m such that $2\beta_Q^2\lambda/(m^2 T) < \varepsilon$, we can guarantee that $\limsup_{i\to\infty} \mathbb{P}[W_i = T] < \varepsilon$, and we obtain the claim. □

Remark 5.2 Some remarks are in order.

1) The constant β_Q depends on the shape of the service regions; for example, if service regions are approximately square, then $\beta_Q \approx 2$ (recall that we are including in β_Q the contribution of the initial travel to the closest demand in the TSP tour).

2) Markov inequality is often a loose upper bound for the cumulative distribution function of a random variable; thus, we believe that the result of Theorem 5.1 is very conservative.

3) Notice, moreover, that the TSP policy does not rely on the knowledge of demands' release times; in other words, it does not give priority to demands that are about to expire. It is clear that, to lower the number of needed vehicles, the knowledge of the releases times should be exploited.

6 The dynamic and stochastic vehicle routing problem with customer impatience

In this section, we extend our analysis to the case in which the time service requests remain active is itself a random variable, describing customer impatience. In this case, it is desired to minimize the fraction of service requests missed because of impatience. A preliminary version of these results appeared in [28].

6.1 Problem definition

The problem definition is similar to that of Section 3. The new fact is that, now, each demand i has an associated *random* impatience time L_i; should the i-th demand not be visited within time L_i from its arrival, it will expire. The impatience times L_i are independent and identically distributed according to a density $f_L : \mathbb{R}_+ \to \mathbb{R}$. We assume that demands' impatience is known to the vehicles *only* via prior statistics. This is in sharp contrast to the standard vehicle routing problem with time windows, where the time windows are known by the vehicles.

Similarly as before, we define W_i as the sojourn time in the system for the i-th demand. The random variable W_i is the elapsed time between the arrival of demand i and the time when either one of the vehicles visits its location or such demand departs from the system due to impatience - whichever is smaller. A demand is considered serviced if $W_i < L_i$.

Given $\varphi \in (0, 1]$, the objective is to ensure that no more than a fraction φ (where $\varphi \in (0, 1]$ is a control parameter) out of all the arrived demands departs impatiently before service. Notice that we rule out the case $\varphi = 0$. We want to answer the questions: what is a sufficient number of mobile agents needed to ensure that each service request is fulfilled before expiring, with probability at least $1 - \varphi$? What strategy should they use to ensure this objective is attained?

We will restrict our analysis to the heavy load case.

6.2 From customer impatience to deterministic time windows

Our approach is to transform the problem with customer impatience into a problem with deterministic time windows, and then to apply the results of the previous sections. Recalling that the impatience times are *identically* distributed, define the critical time T_{crit} as

$$T_{\text{crit}} \doteq \max \left\{ T \in \mathbb{R}_+ : \int_T^\infty f_L(t)dt = \mathbb{P}\left[L > T\right] \geqslant 1 - \varphi \right\}.$$

Clearly, if a routing policy is able to ensure that *each* demand location (regardless of its impatience) is visited within time T_{crit} from its arrival, then this policy ensures that no more than a fraction φ out of all the arrived demands departs impatiently before service. Through the concept of critical time we can, therefore, address the problem of servicing demands *with* impatience as the problem of visiting *all* demands' locations, regardless of their impatience (i.e., even if they depart impatiently), within a constant time. In other words, through the concept of critical time, we cast the problem of servicing demands with impatience into the previous dynamic and stochastic VRPTW, in which the time windows have length T_{crit} and we require a system reliability $\varepsilon \to 0^+$.

This approach on the one hand introduces some degree of conservatism, but on the other hand it simplifies considerably the mathematical analysis.

6.3 A provably correct heavy load policy

From the previous discussion, we argue that, in heavy load, the TSP policy (see Algorithm 1) solves the dynamic and stochastic vehicle routing problem with customer impatience for any $\varphi \in (0, 1]$. An upper bound on the number of vehicles needed by the TSP policy is

$$m = \left\lceil \beta \sqrt{\frac{2\lambda}{T_{\text{crit}}}} \right\rceil. \tag{17}$$

7 Simulation results for the TSP policy

In this section we verify by simulations the correctness of the previous results. All simulations are performed using linkern[3] as a solver to generate approximations to the optimal TSP tour. This powerful solver

---

[3] linkern is written in ANSI C and is freely available for academic research use at http://www.tsp.gatech.edu//concorde.html.

Table 1 Fraction ρ of expired demands

max ρ	$\mathbb{E}[\rho]$	$\sigma(\rho)$
$4.1 \cdot 10^{-3}$	$3.5 \cdot 10^{-3}$	$7.5 \cdot 10^{-4}$

yields approximations in the order of 10% of the optimal tour cost very quickly for many instances. For example, in our numerical experiments on a 2.4 GHz Pentium machine, approximations of random TSPs with 1,000 points typically required about two seconds of CPU time.

7.1 TSP policy in heavy load

The analysis of the TSP policy in heavy load culminates in Eq. 12, which dictates how many vehicles are needed to ensure that almost no demand is missed.

We consider the scenario $\lambda = 1000$ and $T = 5$. Then, Eq. 12 prescribes $m_{\text{TSP}} = 15$ vehicles. We run, starting from random initial conditions, 100 simulations; each run consists of 200 iterations of the TSP policy. For each run, we record the fraction ρ of demands that expire before service. Table 1 summarizes simulation results; we report the worst case value of ρ, the mean value of ρ and its standard deviation (mean value and standard deviation are over 100 realizations).

From Table 1, the worst case fraction of missed demands is $\rho = 4.1 \cdot 10^{-3}$, that is 4 demands every 1000 demands are missed in the worst case; recalling that our analysis holds in the limit $\lambda \to \infty$, and that we are using an approximate TSP solver, we conclude that simulation results match our previous analysis.

The central idea in the analysis of the TSP policy in heavy load is that each iteration should last a time interval less than $T/2$. Figure 1 shows the worst case iteration length versus the number of vehicles employed. We consider the case $T = 5$ and $\lambda = 400$. Equation 12 prescribes 10 vehicles, and $m = 10$ is exactly the minimum number of vehicles to obtain a worst case
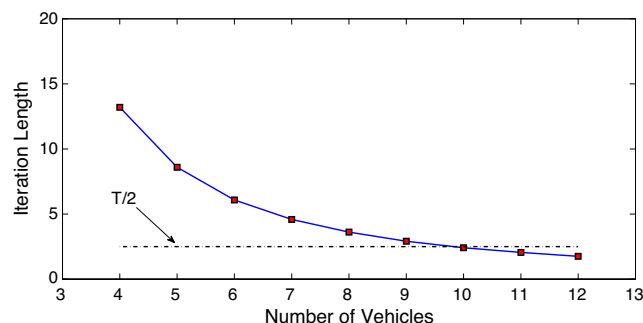

Figure 1 Iteration length as a function of m; $\lambda = 400$, $T = 5$, $m_{\text{TSP}} = 10$

iteration length smaller than $T/2$. This shows that our analysis of the TSP policy is rather tight.

7.2 TSP policy in normal load

We now test the TSP policy in moderate load, with the number of vehicles prescribed by Theorem 5.1. We compute the maximum fraction ρ of missed demands as a function of ε and λ. The maximum is over 100 sample paths, with 200 iterations each. We consider $T = 5$, $\varepsilon \in \{0.1, 0.4, 0.6, 0.9\}$ and $\lambda = \{1, 5, 10, 20, 30, 40, 50, 60, 70, 80\}$. The number of vehicles is dictated by Theorem 5.1.

For all pairs (ε, λ), we obtained max $\rho = 0$. This result is somewhat expected, since, as discussed before, Theorem 5.1 provides a very loose upper bound on the number of vehicles needed to guarantee ε-reliability.

It is interesting to verify if the results obtained for the heavy load case, in particular Eq. 12, remain valid for moderate values of λ. Figure 2 shows the worst case fraction of missed demands for various values of λ (the worst case is with respect to 100 sample paths). The number of vehicles is prescribed by Eq. 12. It can be seen that for $\lambda > 350$ $\rho \approx 0$, and that for $\lambda \leq 350$ ρ is never larger than 0.1.

7.3 TSP policy with customer impatience

First, we consider the scenario where demands have an impatience uniformly distributed in $[0, 90]$, i.e.

$$f_L(t) = \begin{cases} 1/90 & \text{if } t \in [0, 90]; \\ 0 & \text{otherwise.} \end{cases}$$

We set $\varphi = 0.05$; therefore $T_{\text{crit}} = 4.5$ seconds. We consider $\lambda = \{10, 20, 40, 50, 75, 100\}$. For each value of λ, we run, starting from random initial conditions, 100 simulations. For each λ, we record the fraction ρ of demands that are missed in the worst case run. For
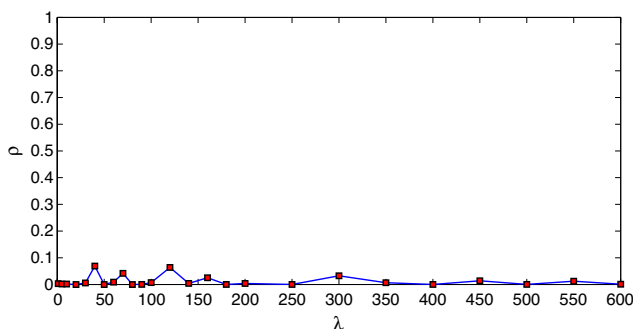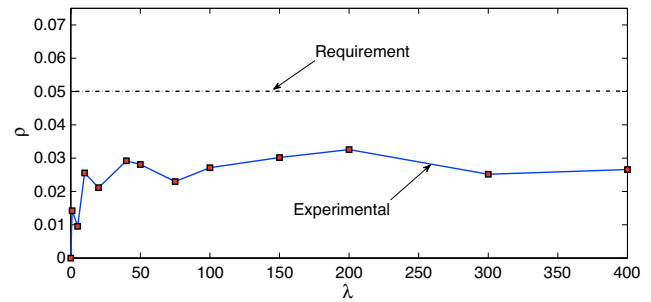

Figure 3 Fraction of demands that depart impatiently as a function of λ; the number of agents is respectively $m = \{1, 1, 2, 2, 3, 4, 4, 5, 5, 6, 7, 9, 10\}$. The impatience time follows a uniform distribution

every value of λ, the number of vehicles is dictated by Eq. 17.

Figure 3 shows the results. The TSP policy always satisfies the requirement that no more than $\varphi = 5\%$ of demands depart impatiently. The requirement is satisfied with a considerable safety margin, and this is a consequence of the conservatism that characterizes our approach. Recall that Eq. 17 was derived under the heavy load assumption. These simulations show that Eq. 17 seems to be applicable for every value of λ.

Then, we consider an impatience that follows an exponential distribution with the same mean of the previous uniform distribution, i.e.:

$$f_L(t) = \begin{cases} \delta e^{-\delta t} & \text{if } t \geq 0; \\ 0 & \text{otherwise.} \end{cases}$$

where $\delta = 1/45$ seconds. We repeat the previous simulations with this new customer impatience model. The TSP policy, also in this case, always satisfies the requirement, with a considerable safety margin (see Fig. 4).
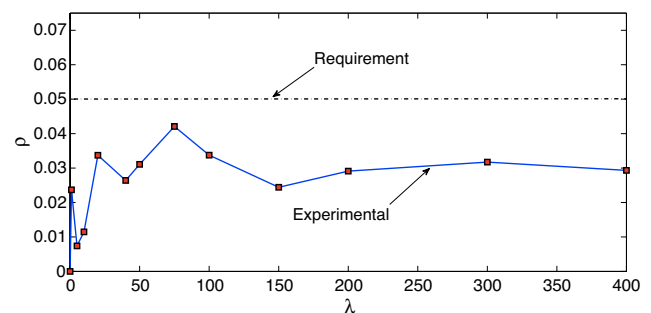

Figure 2 Fraction ρ of missed demands as a function of λ. The number of vehicles is prescribed by Eq. 12

Figure 4 Fraction of demands that depart impatiently as a function of λ; the number of agents is respectively $m = \{1, 1, 2, 3, 3, 5, 5, 6, 7, 9, 10, 12, 14\}$. The impatience time follows an exponential distribution
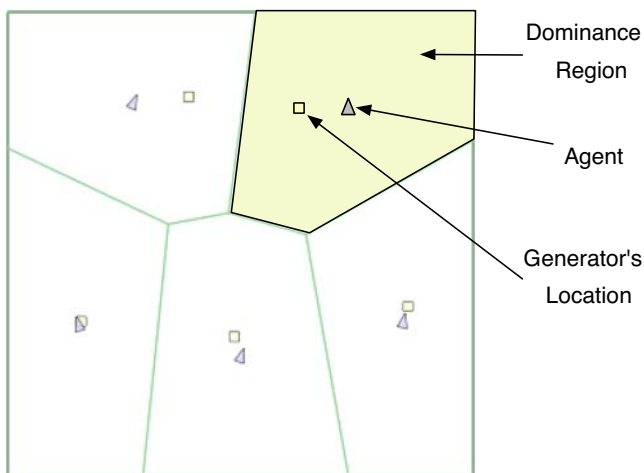
Figure 5 Agents, virtual generators and regions of dominance

8 Decentralized equitable partitioning

The TSP policy is centralized, since it requires a centralized assignment of the service regions. As introduced in [7, 29], assuming that Q is convex, m agents can achieve a configuration with equitable (i.e. with the same area) service regions in a *decentralized* way. The first step is to associate to each vehicle i a *virtual generator* g_i. We define the region of dominance for vehicle i as the Voronoi cell $V_i = V_i(G)$, where $G \doteq (g_1, \cdots, g_m)$. We refer to the partition into regions of dominance induced by the set of virtual generators G as $V(G)$. A virtual generator g_i is simply an artificial variable locally controlled by the i-th agent; in particular, g_i is a virtual point (see Fig. 5).

We shall assume that each vehicle has sufficient information available to determine: (1) its Voronoi cell, and (2) the locations of all outstanding events in its
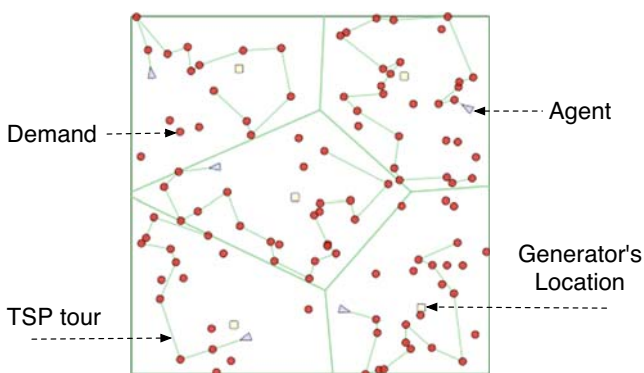
Voronoi cell. A control policy that relies on information (1) and (2) is *Voronoi-distributed* in the sense that the behavior of each vehicle depends only on the location of the other agents with contiguous Voronoi cells (the number of Voronoi neighbors of each generator is on average less than or equal to 6 [27]). Accordingly, Voronoi-distributed policies are *spatially distributed* and *scalable* in the number of agents. A spatially distributed algorithm for the local computation and maintenance of Voronoi cells is provided in [30].

The key idea, then, is to enable virtual generators to follow a (Voronoi-distributed) gradient descent law such that an equitable partition is reached (see [7] for the details). Figure 6 shows a typical partition that arises with such algorithm.

9 Conclusion

In this paper, we introduced and studied a dynamic and stochastic version of the well-known vehicle routing problem with time windows. In our model, each service request is generated by a spatio-temporal stochastic process; once a service request has been generated, it remains active for a certain deterministic amount of time, and then expires. Given $\varepsilon \in (0, 1]$, the objective is to find routing policies that, with the minimum possible number of agents, ensure that each demand has a probability of being visited before expiring greater than $1 - \varepsilon$.

We first presented a lower-bound on the minimum number of vehicles for the case where $\varepsilon \to 0^+$ and the arrival rate of events is large (heavy load). Next, we presented a heavy-load routing strategy for servicing almost all demands before they expire and showed that the number of vehicles is within a small constant from the optimal value. We then studied the case where the arrival rate is arbitrary and an arbitrary fraction of the events must be serviced; also for this case we provided a provably correct algorithm. Finally, we extended these results to the case where the expiration time of an event is itself a random variable. In addition, we showed how the routing strategies presented in the paper can be executed in a distributed fashion. Our theoretical results are confirmed by simulations.

Several important research directions will be the objective of future work. First of all, the results in this paper assume that the time windows are either deterministic and uniform for all demands, or stochastic and unknown to the agents; it is of interest to extend our work to a more general setup. Second, we will investigate the extension of the work presented in this paper to the case in which the demands are generated

Figure 6 Each vehicle services outstanding demands inside its own region of dominance by following a TSP tour

according to non-uniform spatial distributions. Finally, some of the bounds in this paper are conservative, thus we plan to search for tighter bounds.

Appendix

In this appendix, we restate and prove Lemma 4.3.

Lemma 4.3 *Assume*

$$\beta\sqrt{\frac{2}{T}} < \lim_{k\to\infty}\frac{m(k)}{\sqrt{kl}} < +\infty.$$

Then, at each epoch $r \geq 1$:

(1) $\displaystyle\lim_{k\to\infty} n_r^k \stackrel{a.s}{=} \infty$;

(2) $\displaystyle\lim_{k\to\infty}\frac{C_r^k - c}{\sqrt{n_r^k}}\sqrt{m(k)} \stackrel{a.s}{=} \beta$;

(3) $\displaystyle\lim_{k\to\infty}\frac{n_r^k}{C_{r-1}^k}\frac{1}{kl/m(k)} \stackrel{a.s}{=} 1.$

Proof The arrival rate in a service region is $kl/m(k)$. Consider an arbitrary deterministic time interval $c > 0$ and let $n^k(c)$ be the number of Poisson arrivals, with rate $kl/m(k)$, in such time interval; we start by proving that $\lim_{k\to\infty} n^k(c) \stackrel{a.s.}{=} \infty$. From Section 2, we have

$$\lim_{k\to\infty} n^k(c) \stackrel{a.s.}{=} \infty \Leftrightarrow \forall N > 0 \lim_{k\to\infty}\mathbb{P}\left[\bigcup_{p=k}^{\infty}[n^p(c) < N]\right]=0.$$

Therefore, we want to show that

$$\forall \varepsilon > 0 \quad \exists \bar{k} : \forall k > \bar{k} \quad \mathbb{P}\left[\bigcup_{p=k}^{\infty}[n^p(c) < N]\right] < \varepsilon. \tag{18}$$

By assumption

$$L < \lim_{k\to\infty}\frac{m(k)}{\sqrt{kl}} < U,$$

where L and U are two positive constants (their values are of no concern here). Thus, there exists $k_1 > 0$ such that for all $k \geq k_1$

$$L < \frac{m(k)}{\sqrt{kl}} < U.$$

Let k_2 be the smallest integer k such that $\sqrt{kl}c/L > 1$. Now, by using the union bound and assuming $k > \max(k_1, k_2)$, we have

$$\mathbb{P}\left[\bigcup_{p=k}^{\infty}[n^p(c) < N]\right] \leqslant \sum_{p=k}^{\infty}\mathbb{P}[n^p(c) < N]$$

$$= \sum_{p=k}^{\infty}\sum_{n=0}^{N-1} e^{-\frac{plc}{m(p)}}\frac{(plc/m(p))^n}{n!}$$

$$\leq \sum_{p=k}^{\infty}\sum_{n=0}^{N-1} e^{-\frac{\sqrt{pl}c}{U}}\frac{(\sqrt{pl}c/L)^n}{n!}$$

$$\leq N\sum_{p=k}^{\infty} e^{-\frac{\sqrt{pl}c}{U}}(\sqrt{pl}c/L)^{N-1}.$$

The series $\sum_{p=0}^{\infty} e^{-\frac{\sqrt{pl}c}{U}}(\sqrt{pl}c/L)^{N-1}$ is convergent (as it can be easily verified with the comparison test); therefore, $\lim_{k\to\infty}\sum_{p=k}^{\infty} e^{-\frac{\sqrt{pl}c}{U}}(\sqrt{pl}c/L)^{N-1} = 0$. Let k_3 be the smallest integer such that, for all $k > k_3$, $\sum_{p=k}^{\infty} e^{-\frac{\sqrt{pl}c}{U}}(\sqrt{pl}c/L)^{N-1} < \varepsilon/N$. Then, by letting $\bar{k} \doteq \max(k_1, k_2, k_3)$, we prove Eq. 18.

Now, the time interval between epochs $r - 1$ and r (call it τ_{r-1}) is at least as large as the computation time $c > 0$. Thus, if Ω is the set of sample functions ω for which $\lim_{k\to\infty} n^k(c) = \infty$, we have $\lim_{k\to\infty} n^k(\tau_{r-1}) = \infty$ for all ω in Ω. Since $\mathbb{P}[\Omega] = 1$ (and $n^k(\tau_{r-1}) = n_r^k$ by definition), part (1) is proven.

We now prove part (2). By Eq. 4, we know that, given a set D_n of n points that are independent and uniformly distributed in a region of unit area, we have $\lim_{n\to\infty} \text{TSP}(D_n)/\sqrt{n} \stackrel{a.s.}{=} \beta$. From part (1) of this Lemma, we also have that $\lim_{k\to\infty} n_r^k \stackrel{a.s}{=} \infty$. Assume, now, that we scale by a factor $\sqrt{m(k)}$ the coordinates of the demands that arrive in between epochs $r - 1$ and r, when the overall arrival rate is kl. Let F_r^k be the length of the tour through such scaled demands (the scaled demands are uniformly distributed in a region with unit area). Thus, for any sample function (except possibly for a set of probability zero), $F_r^k/\sqrt{n_r^k}$ runs through the same sequence of values with increasing k as $\text{TSP}(D_n)/\sqrt{n}$ runs through with increasing n. Thus if Ω is the set of sample functions ω for which both $\lim_{n\to\infty} \text{TSP}(D_n)/\sqrt{n} = \beta$ and $\lim_{k\to\infty} n_r^k \stackrel{a.s.}{=} \infty$, we have $\lim_{k\to\infty} F_r^k/\sqrt{n_r^k} = \beta$ for all sample functions in Ω. By Eq. 4 and part (1) of the Lemma we have $\mathbb{P}[\Omega] = 1$. Thus

$$\lim_{k\to\infty}\frac{F_r^k}{\sqrt{n_r^k}} \stackrel{a.s.}{=} \beta.$$

By scaling, we have $F_r^k = (C_r^k - c)\sqrt{m(k)}$, and thus we get the limit in part (2).

Finally, we prove part (3). The number of arrivals in between epochs $r-1$ and r is $N^{\frac{kl}{m(k)}}(C_{r-1}^k)$, where $\{N^{\frac{kl}{m(k)}}(t); \ t \geq 0\}$ is a Poisson process with intensity $kl/m(k)$. By the strong law of large numbers for renewal processes (see, for example, [31]) we have

$$\lim_{t \to \infty} N^1(t)/t \overset{a.s.}{=} 1.$$

For every k, consider the time scaling:

$$\tau \doteq \frac{kl}{m(k)} t.$$

Notice that in the new time scale the arrival rate is $\lambda = 1$ for every k. Let $\tilde{C}_{r-1}^k$ be the length of the time interval between epochs $r-1$ and r in the new time scale, i.e. $\tilde{C}_{r-1}^k = \frac{kl}{m(k)} C_{r-1}^k$. Since, by definition, $C_{r-1}^k \geq c > 0$, and since by assumption $\lim_{k\to\infty} kl/m(k) = \infty$, we have

$$\lim_{k \to \infty} \tilde{C}_{r-1}^k \overset{a.s.}{=} \infty.$$

Therefore, with similar arguments as before, we obtain

$$\lim_{k \to \infty} \frac{N^1(\tilde{C}_{r-1}^k)}{\tilde{C}_{r-1}^k} \overset{a.s.}{=} 1. \tag{19}$$

By scaling, we have $N^1(\tilde{C}_{r-1}^k) \equiv N^{\frac{kl}{m(k)}}(C_{r-1}^k)$, therefore Eq. 19 is equivalent to

$$\lim_{k \to \infty} \frac{N^{\frac{kl}{m(k)}}(C_{r-1}^k)}{C_{r-1}^k} \frac{1}{kl/m(k)} \overset{a.s.}{=} 1.$$

Since, by definition, $n_r^k = N^{\frac{kl}{m(k)}}(C_{r-1}^k)$, we obtain the claim. $\qquad\square$

References

1. Solomon MM (1987) Algorithms for the vehicle routing and scheduling problems with time window constraints. Oper Res 35(2):254–265
2. Desrosiers J, Dumas Y, Solomon MM, Soumis F (1995) Time constrained routing and scheduling. In: Ball MO, Magnanti TL, Monma CL, Nemhauser GL (eds) Handbooks in operations research and management science, chapter 8. Elsevier, Amsterdam, The Netherlands, pp 35–139
3. Toth P, Vigo D (2002) The vehicle routing problem. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, PA
4. Bräysy O, Gendreau M (2005) Vehicle routing problem with time windows, part I: route construction and local search algorithms. Transp Sci 39(1):104–118
5. Bräysy O, Gendreau M (2005) Vehicle routing problem with time windows, part II: metaheuristics. Transp Sci 39(1):119–139
6. Bertsimas DJ, van Ryzin GJ (1993) Stochastic and dynamic vehicle routing in the Euclidean plane with multiple capacitated vehicles. Adv Appl Probab 25(4):947–978
7. Pavone M, Frazzoli E, Bullo F (2007) Decentralized algorithms for stochastic and dynamic vehicle routing with general target distribution. In: Proc IEEE conference on decision and control, New Orleans, LA
8. Bisnik N, Abouzeid A, Isler V (2007) Stochastic event capture using mobile sensors subject to a quality metric. IEEE Trans Robot 23:676–692
9. Savelsbergh MWP (1985) Local search in routing problems with time windows. Ann Oper Res 4(1):285–305
10. Bertsimas DJ, van Ryzin GJ (1993) Stochastic and dynamic vehicle routing with general interarrival and service time distributions. Adv Appl Probab 25:947–978
11. Bertsimas DJ, van Ryzin GJ (1991) A stochastic and dynamic vehicle routing problem in the Euclidean plane. Oper Res 39:601–615
12. Frazzoli E, Bullo F (2004) Decentralized algorithms for vehicle routing in a stochastic time-varying environment. In: Proc IEEE conf on decision and control, Paradise Island, Bahamas
13. Huang C-F, Tseng Y-C (2003) The coverage problem in a wireless sensor network. In: 2nd ACM international conference on wireless sensor networks and applications (WSNA). ACM Press, New York, NY, USA, pp 115–121
14. Meguerdichian S, Koushanfar F, Potkonjak M, Srivastava MB (2001) Coverage problems in wireless ad-hoc sensor networks. In: 20th annual IEEE conference on computer communications (INFOCOM), pp 1380–1387
15. Wang X, Xing G, Zhang Y, Lu C, Pless R, Gill C (2003) Integrated coverage and connectivity configuration in wireless sensor networks. In: SenSys '03: proceedings of the 2nd international conference on embedded networked sensor systems. ACM Press, New York, NY, USA, pp 28–39
16. Xing G, Lu C, Pless R, O'Sullivan JA (2004) Co-grid: an efficient coverage maintenance protocol for distributed sensor networks. In: 3rd international symposium on information processing in sensor networks (IPSN). ACM Press, New York, NY, USA, pp 414–423
17. Isler V (2006) Placement and distributed deployment of sensor teams for triangulation based localization. In: Proc IEEE ICRA, pp 3095–3100
18. Cortés J, Martínez S, Karatas T, Bullo F (2004) Coverage control for mobile sensing networks. IEEE Trans Robot Autom 20(2):243–255
19. Liu B, Brass P, Dousse O, Nain P, Towsley D (2005) Mobility improves coverage of sensor networks. In: International symposium on mobile ad hoc networking and computing (MobiHoc). ACM Press, New York, NY, USA, pp 300–308
20. Chekuri C, Korula N, Pál M (2008) Improved algorithms for orienteering and related problems. In: SODA '08: proceedings of the nineteenth annual ACM-SIAM symposium on discrete algorithms. Philadelphia, PA, USA, Society for Industrial and Applied Mathematics, pp 661–670
21. Durrett R (1996) Probability: theory and examples. Duxbury Press, Belmont, CA
22. Stark H, Woods JW (1986) Probability, random processes, and estimation theory for engineers. Prentice-Hall, Inc, Upper Saddle River, NJ

23. Beardwood J, Halton J, Hammersley J (1959) The shortest path through many points. In: Proc of the Cambridge Philoshopy Society, vol 55, pp 299–327
24. Percus G, Martin OC (1996) Finite size and dimensional dependence of the Euclidean traveling salesman problem. Phys Rev Lett 76(8):1188–1191
25. Larson RC, Odoni AR (1981) Urban operations research. Prentice-Hall, Englewood Cliffs, NJ
26. Steele JM (1990) Probabilistic and worst case analyses of classical problems of combinatorial optimization in Euclidean space. Math Oper Res 15(4):749–770
27. Sugihara K, Okabe A, Boots B, Chiu SN (2000) Spatial tessellations: concepts and applications of Voronoi diagrams. Wiley, New York, NY
28. Pavone M, Bisnik N, Frazzoli E, Isler V (2007) Decentralized vehicle routing in a stochastic and dynamic environment with customer impatience. In: Proc Robocomm, Athens, Greece
29. Pavone M, Frazzoli E, Bullo F (2008) Distributed algorithms for equitable partitioning policies: theory and applications. In: Proc IEEE conference on decision and control, Cancun, Mexico
30. Cao M, Hadjicostis CN (2003) Distributed algorithms for Voronoi diagrams and applications in ad-hoc networks. Technical Report UILU-ENG-03-2222, UIUC Coordinated Science Laboratory
31. Gallager RG (1996) Discrete stochastic processes. Kluwer, Dordrecht, The Netherlands