Operations Research Letters 48 (2020) 599–606

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

Approximation algorithms for the a priori traveling repairman

Fatemeh Navidi [a], Inge Li Gørtz [b], Viswanath Nagarajan [a,*]

[a] *University of Michigan, Ann Arbor, MI, USA*
[b] *Technical University of Denmark, DTU Compute, Denmark*

ARTICLE INFO

Article history:
Received 4 June 2020
Accepted 12 July 2020
Available online 22 July 2020

Keywords:
Traveling repairman problem
A priori optimization
Approximation algorithms

ABSTRACT

We consider the *a priori* traveling repairman problem, which is a stochastic version of the classic traveling repairman problem. Given a metric (V, d) with a root $r \in V$, the traveling repairman problem (TRP) involves finding a tour originating from r that minimizes the sum of arrival-times at all vertices. In its *a priori* version, we are also given independent probabilities of each vertex being active. We want to find a master tour τ originating from r and visiting all vertices. The objective is to minimize the expected sum of arrival-times at all active vertices, when τ is shortcut over the inactive vertices. We obtain the first constant-factor approximation algorithm for *a priori* TRP under independent non-uniform probabilities. Our result provides a general reduction from non-uniform to uniform probabilities, and uses (in black-box manner) an existing approximation algorithm for *a priori* TRP under uniform probabilities.

1. Introduction

A priori optimization [5] is an elegant model for stochastic combinatorial optimization, that is particularly useful when one needs to repeatedly solve instances of the same optimization problem. The basic idea here is to reduce the computational overhead of solving repeated problem instances by performing suitable pre-processing using distributional information. More specifically, in an *a priori* optimization problem, one is given a probability distribution Π over inputs and the goal is to find a "master solution" τ. Then, after observing the random input A (drawn from the distribution Π), the master solution τ is modified using a simple rule to obtain a solution τ_A for that particular input. The objective is to minimize the expected value of the master solution. For a problem with objective function ϕ, we are interested in:

$$\min_{\tau:\text{master solution}} \mathbb{E}_A\left[\phi(\tau_A)\right].$$

This paper studies the *a priori* traveling repairman problem. The traveling repairman problem (TRP) is a fundamental vehicle routing problem that involves computing a tour originating from a depot/root that minimizes the sum of latencies (i.e. the distance from the root on this tour) at all vertices. The TRP is also known as the traveling deliveryman or minimum latency problem, and has been studied extensively, see e.g. [11,12,16]. In the *a priori* TRP, the master solution τ is a tour visiting all vertices, and for any random input (i.e. subset A of vertices), the solution τ_A is simply obtained by visiting the vertices of A in the order given by τ.

An *a priori* solution is advantageous in settings when we repeatedly solve instances of the TRP that are drawn from a common distribution. For example, we may need to solve one TRP instance on each day of operations, where the distribution over instances is estimated from historical data. Using an *a priori* solution saves on computation time as we do not have to solve each instance from scratch. Moreover, for vehicle routing problems (VRPs) there are also practical advantages to have a pre-planned master tour, e.g. drivers have familiarity with the route followed each day. See [7,17], and [10] for more discussion on the benefits of a pre-planned VRP solution.

1.1. Problem definition

The traveling repairman problem (TRP) is defined on a finite metric (V, d) where V is a vertex set and $d : V \times V \to \mathbb{R}_+$ is a distance function. We assume that the distances are symmetric and satisfy triangle inequality. There is also a designated root vertex $r \in V$. The goal is to find a tour τ originating from r that visits all vertices. The *latency* of any vertex v in tour τ is the length of the path from r to v along τ. The objective in TRP is to minimize the sum of latencies of all vertices.

In the *a priori* TRP, in addition to the above input we are also given activation probabilities $\{p_v\}_{v\in V}$ at all vertices; we use Π to denote this distribution. In this paper, as in most prior works on *a priori* optimization, we assume that the input distribution Π is independent across vertices. So the active subset $A \subseteq V$ contains

* Corresponding author.
E-mail addresses: navidi@umich.edu (F. Navidi), inge@dtu.dk (I.L. Gørtz), viswa@umich.edu (V. Nagarajan).

https://doi.org/10.1016/j.orl.2020.07.009

each vertex $v \in V$ independently with probability p_v. A solution to *a priori* TRP is a master tour τ originating from r and visiting all vertices. Given an active subset $A \subseteq V$, we restrict tour τ to vertices in A (by shortcutting over $V \setminus A$) to obtain tour τ_A, again originating from r. For each $v \in A$, we use $\mathsf{LAT}^A_\tau(v)$ to denote the latency of v in tour τ_A. We also set $\mathsf{LAT}^A_\tau(v) = 0$ for $v \notin A$. We also use $\mathsf{LAT}^A_\tau = \sum_{v\in A} \mathsf{LAT}^A_\tau(v)$ for the total latency under active subset $A \subseteq V$. The objective is to minimize

$$\mathsf{ELAT}_\tau = \mathbb{E}_A\left[\mathsf{LAT}^A_\tau\right] = \mathbb{E}_A\left[\sum_{v\in A} \mathsf{LAT}^A_\tau(v)\right].$$

1.2. Results

Our main result in this note is the first constant-factor approximation for the *a priori* TRP.

Theorem 1.1. *There is a constant-factor approximation algorithm for the* a priori *traveling repairman problem under independent probabilities.*

Previously, [9] obtained such a result under the restriction that all activation probabilities are *identical*, and posed the general case of non-uniform probabilities as an open question — which we resolve. Our result adds to the small list of *a priori* VRPs with provable worst-case guarantees: traveling salesman, capacitated vehicle routing and traveling repairman.

In fact, we obtain Theorem 1.1 by a generic reduction of *a priori* TRP from non-uniform to uniform probabilities, formalized below.

Theorem 1.2. *There is a* (6.27ρ)*-factor approximation algorithm for the* a priori *traveling repairman problem under independent probabilities, where* ρ *denotes the best approximation ratio for the problem under uniform probabilities.*

Clearly, Theorem 1.1 follows by combining Theorem 1.2 with the $O(1)$-approximation algorithm for *a priori* TRP under uniform probabilities by [9]. As the constant factor in [9] for uniform probabilities is quite large, there is the possibility of improving it using a different algorithm: Theorem 1.2 would be applicable to any such future improvement and yield a corresponding improved result for non-uniform probabilities.

1.3. Related work

The *a priori* optimization model was introduced in [15] and [3], see also the survey by [5]. These papers considered the setting where the metric is itself random and carried out asymptotic analysis (as the number of vertices grows large). They obtained such results for the minimum spanning tree, traveling salesman, capacitated vehicle routing and traveling salesman facility location problems.

Approximation algorithms for *a priori* optimization are more recent: these can handle arbitrary problem instances. Such results are known for the traveling salesman problem [13,18,19,21], capacitated VRP [4,14] and traveling repairman [9].

The *a priori* TRP was recently studied in [9], where a constant-factor approximation algorithm was obtained for the case of uniform independent probabilities. They left open the problem under non-uniform probabilities: Theorem 1.2 resolves this positively. The algorithm in [9] was based on many ideas from the deterministic TRP, but it needed stochastic counterparts of various properties. As noted in [9], their proof relied heavily on the probabilities being uniform and it was unclear how to handle non-uniform probabilities.

We note that the deterministic traveling repairman problem (TRP) has been studied extensively, both in exact algorithms [11,16,22] and approximation algorithms [2,6,8,12].

2. *A priori* TRP with non-uniform distribution

Consider an instance $\mathcal{I}$ of *a priori* TRP on metric (V, d) with $|V| = n$ vertices and independent probabilities $\{p_v\}_{v\in V}$. We show how to "reduce" this instance to one with uniform probabilities, which would prove Theorem 1.2. Our approach is natural: we replace each vertex $v \in V$ with a group S_v of co-located vertices, where each new vertex is active with a uniform probability p independent of the other vertices. Let $\mathcal{J}$ denote the new instance and $(\widehat{V}, d)$ the new metric. Intuitively, when p is chosen much smaller than the p_vs and $|S_v| \approx p_v/p$, the scaled uniform instance $\mathcal{J}$ should behave similar to $\mathcal{I}$. However, proving such a result formally requires significant technical work. For example, the master tour found by an algorithm for the scaled (uniform) instance might not visit all the co-located copies consecutively. We define a *consecutive* master tour for $\mathcal{J}$ as one that visits all co-located vertices consecutively. Then, we show an approximate equivalence between (i) master tours in $\mathcal{I}$ and (ii) consecutive master tours in $\mathcal{J}$. This relies on the independence across vertices and the correspondence between the events "vertex v is active in $\mathcal{I}$" and "at least one vertex of S_v is active in $\mathcal{J}$". This is formalized in Section 2.2. Then, we show in Section 2.4 that any master tour for instance $\mathcal{J}$ can be modified to a "consecutive" master tour with the same or better overall expected latency. Finally, in order to maintain a polynomial-size instance $\mathcal{J}$ (this is reflected in the choice of p), we need to take care of vertices with very small probability separately. In Section 2.3 we show that the overall effect of the small-probability vertices is tiny if they are visited in non-decreasing order of distances at the end of our master tour.

Algorithm 1 Reducing non-uniform instance $\mathcal{I}$ to uniform instance $\mathcal{J}$

1: $Y \leftarrow \{v \in V \mid p_v < 1/n^2\}$ denotes the low probability vertices.
2: $X \leftarrow \{v \in V \mid p_v \geq 1/n^2\}$ denotes all other vertices.
3: $p \leftarrow \frac{1}{n} \min_{v\in X} p_v$
4: Construct instance $\mathcal{J}$ with vertex set $\widehat{V}$ that contains for each $v \in X$, a set S_v of $t_v = \lceil \frac{p_v}{p} \rceil$ copies of v. The distance between any two vertices of S_v is zero for all $v \in X$. The distance between any vertex of S_u and any vertex of S_v is $d(u, v)$. All vertices in $\widehat{V}$ have an independent uniform activation probability p.
5: Run any approximation algorithm for *uniform a priori* TRP on $\mathcal{J}$ to obtain master tour $\widehat{\pi}$.
6: Run procedure MakeConsecutive($\widehat{\pi}$) to ensure that $\widehat{\pi}$ visits each group S_v consecutively.
7: Obtain tour π by visiting vertices of X in the same order that $\widehat{\pi}$ visits S_v for all $v \in V$.
8: Extend π by visiting vertices $w \in Y$ in *non-decreasing* order of $d(r, w)$, to obtain tour $\bar{\pi}$.
9: **return** $\bar{\pi}$.

Algorithm 1 describes the reduction formally. In Step 6, Algorithm 1 relies on a procedure MakeConsecutive that modifies tour $\widehat{\pi}$ such that it visits all copies of the same node consecutively. We will prove Theorem 1.2 by analyzing this algorithm.

2.1. Overview of analysis

We first assume that the master tour $\widehat{\pi}$ on instance $\mathcal{J}$ already visits copies of each vertex consecutively: so there is no need for Step 6. We split this proof into two parts corresponding to the X-vertices (normal probabilities) and Y-vertices (low probabilities). The analysis for X-vertices (Section 2.2) is the main part, where we show that the optimal values of $\mathcal{I}$ and $\mathcal{J}$ are within a constant factor of each other. In Lemma 2.3 we show that a constant-factor perturbation in probabilities of V will only change the cost of any solution (including the optimal) by a constant factor. Then we prove (in Lemma 2.4) that the optimal value of instance $\mathcal{J}$ is within a constant factor of the optimal value of $\mathcal{I}$: although $\mathcal{J}$ has many more vertices than $\mathcal{I}$, the proof exploits the fact that the expected number of active vertices is roughly the same as $\mathcal{I}$. Lemma 2.5 proves the other direction for the cost of our algorithm, i.e. the cost of Algorithm 1 for $\mathcal{I}$ is at most that of the consecutive master tour for $\mathcal{J}$. To handle the Y-vertices, we use a simple expected distance lower-bound to show (in Section 2.3) that visiting Y at the end of our tour only adds a small factor to the overall expected cost.

Above, we assumed above that the master tour $\widehat{\pi}$ visits copies of each vertex consecutively. However, this is not necessary in an approximation algorithm for the uniform *a priori* TRP (see Appendix A). So in Section 2.4, we provide a subroutine that ensures this consecutive property. Thus, our approach can be combined with *any* algorithm for uniform *a priori* TRP.

2.2. Analysis for vertices in X

Here we analyze the steps of the algorithm that deal with vertices in X, i.e. with probability at least $\frac{1}{n^2}$. In order to reduce notation, we will assume here that $X = V$ which is the entire vertex set. Recall that $p = \frac{1}{n} \cdot \min_{v\in V} p_v$. Note that q_v is equal to the probability of having at least one active vertex in S_v, for each $v \in V$. Also define $\bar{p}_v = \min\left\{(1+\frac{1}{n})p_v, 1\right\}$, $t_v = \lceil p_v/p \rceil$ and $q_v = 1-(1-p)^{t_v}$ for each $v \in V$. We will refer to the instances on metric (V, d) with probabilities $\{p_v\}_{v\in V}$, $\{q_v\}_{v\in V}$ and $\{\bar{p}_v\}_{v\in V}$ as $\mathcal{I}_p$, $\mathcal{I}_q$ and $\mathcal{I}_{\bar{p}}$ respectively. Note that the original instance is $\mathcal{I} = \mathcal{I}_p$. For simplicity we use $\mathbf{p}$, $\mathbf{q}$ and $\bar{\mathbf{p}}$ to refer to the vector of probabilities for each corresponding distribution.

Lemma 2.1. *For any $v \in V$, we have $p_v(1-\frac{1}{e}) \le q_v \le \bar{p}_v \le p_v(1+\frac{1}{n})$.*

Proof. Note that for every real number x we have $1+x \le e^x$. Using $x = -p$ and raising both sides to the power of t_v we obtain $(1-p)^{t_v} \le e^{-pt_v}$. Now we have:

$$q_v = 1-(1-p)^{t_v} \ge 1-e^{-pt_v} \ge 1-e^{-p\cdot\frac{p_v}{p}} = 1-e^{-p_v} \ge (1-\frac{1}{e})p_v .$$

The second inequality uses $t_v = \lceil p_v/p \rceil$ and the last one uses $1-e^{-x} \ge (1-1/e)x$ for any $x \in [0, 1]$ with $x = p_v$. Now, to prove the other inequality we use union bound to obtain:

$$q_v = 1-(1-p)^{t_v} \le 1-(1-pt_v) = pt_v \le p(\frac{p_v}{p}+1)$$
$$\le p_v + \frac{p_v}{n} = p_v(1+\frac{1}{n}) .$$

Combined with the fact that $q_v \le 1$, we obtain $q_v \le \bar{p}_v$. □

Lemma 2.2. *Let π be any master tour on (V, d). Consider two probability distributions given by $\{q_v\}_{v\in V}$ and $\{\bar{p}_v\}_{v\in V}$ such that $0 \le q_v \le \bar{p}_v \le 1$ for each $v \in V$. Then the expected latency of π under $\{q_v\}_{v\in V}$ is at most that under $\{\bar{p}_v\}_{v\in V}$.*

Proof. Let function $f(p_1, \ldots, p_n)$ denote the expected latency of π as a function of vertex probabilities $\{p_v\}$. We will show that all partial derivatives of f are non-negative. This would imply the lemma. We can express f as a multilinear polynomial

$$f(\mathbf{p}) = \sum_{A\subseteq V}\left(\prod_{u\in A} p_u \prod_{w\in V\setminus A}(1-p_w)\right)\cdot \mathsf{LAT}_\pi^A .$$

Recall that LAT_π^A is the total latency of vertices in active set A in the shortcut tour π_A. So the vth partial derivative is:

$$\frac{\partial f}{\partial p_v} = \sum_{A\subseteq V\setminus v}\left(\prod_{u\in A} p_u \prod_{w\in (V\setminus A)\setminus v}(1-p_w)\right)\left(\mathsf{LAT}_\pi^{A\cup v} - \mathsf{LAT}_\pi^A\right) .$$

For any $A \subseteq V \setminus v$, it follows by triangle inequality that $\mathsf{LAT}_\pi^{A\cup v} \ge \mathsf{LAT}_\pi^A$. This shows that each term in the above summation is non-negative and so $\frac{\partial f}{\partial p_v} \ge 0$. □

Lemma 2.3. *Let π be any master tour on (V, d). Consider two probability distributions given by $\{q_v\}_{v\in V}$ and $\{\bar{p}_v\}_{v\in V}$ and some constant $\beta \le 1$ such that $\beta\bar{p}_v \le q_v \le \bar{p}_v$ for each $v \in V$. Then the expected latency of π under $\{q_v\}_{v\in V}$ is at least β^3 times that under $\{\bar{p}_v\}_{v\in V}$.*

Proof. Let function $f(p_1, \ldots, p_n)$ denote the expected latency of π under probabilities $\{p_v\}_{v\in V}$. For $\mathbf{q}$ and $\bar{\mathbf{p}}$ as in the lemma, we will show $f(\mathbf{q}) \ge \beta^3 \cdot f(\bar{\mathbf{p}})$. To this end, we now view f as the expected sum of terms corresponding to all possible edges used in the shortcut tour π_A (where A is the active set). Renumber the vertices as $1, 2, \ldots, n$ in the order of appearance in π; so the root r is numbered 1. For any $i, j \in [n]$ let I_{ij} denote the indicator random variable for (ordered) edge (i, j) being used in the shortcut tour π_A. For any $j \in [n]$, let N_j denote the number of active vertices among $\{j, j+1, \ldots, n\}$. Then, the total latency of tour π_A is

$$\sum_{1\le i<j\le n} d(i, j)\cdot I_{ij}\cdot N_j .$$

Under probabilities $\mathbf{q}$, for any $i < j$ we have $\mathbb{E}[I_{ij}] = q_i \cdot q_j \cdot \prod_{k=i+1}^{j-1}(1-q_k)$ which corresponds to the event that i and j are active but all vertices between i and j are inactive. Moreover, $\mathbb{E}[N_j|I_{ij} = 1] = 1+\sum_{\ell=j+1}^{n} q_\ell$ using the independence across vertices. So we can write:

$$f(\mathbf{q}) = \sum_{1\le i<j\le n} d(i, j)\cdot \mathbb{E}[I_{ij}]\cdot \mathbb{E}[N_j|I_{ij} = 1]$$
$$= \sum_{1\le i<j\le n} d(i, j)\cdot q_i\cdot q_j\cdot \prod_{k=i+1}^{j-1}(1-q_k)\left(1+\sum_{\ell=j+1}^{n} q_\ell\right) .$$

Note that for any $i < j$, using the fact that $\beta\cdot\bar{\mathbf{p}} \le \mathbf{q} \le \bar{\mathbf{p}}$ we have:

$$q_i\cdot q_j\cdot \prod_{k=i+1}^{j-1}(1-q_k)\left(1+\sum_{\ell=j+1}^{n} q_\ell\right) \ge \beta^3\cdot \bar{p}_i\cdot \bar{p}_j$$
$$\cdot \prod_{k=i+1}^{j-1}(1-\bar{p}_k)\left(1+\sum_{\ell=j+1}^{n} \bar{p}_\ell\right) .$$

This implies $f(\mathbf{q}) \ge \beta^3 \cdot f(\bar{\mathbf{p}})$ as desired. □

Lemma 2.4. *Instances $\mathcal{I}$ and $\mathcal{J}$ in Algorithm 1 satisfy*

$$\mathsf{OPT}(\mathcal{J}) \le \left(\frac{e}{e-1}\right)\left(1+\frac{1}{n}\right)^4 \cdot \mathsf{OPT}(\mathcal{I}) .$$

Proof. Recall the three instances $\mathcal{I} = \mathcal{I}_p$, $\mathcal{I}_q$ and $\mathcal{I}_{\bar{p}}$ on the metric (V, d). Using $\mathbf{q} \le \bar{\mathbf{p}}$ (Lemma 2.1) and Lemma 2.2 we have $\mathsf{OPT}(\mathcal{I}_q) \le \mathsf{OPT}(\mathcal{I}_{\bar{p}})$. Further, using $\mathbf{p} \le \bar{\mathbf{p}} \le (1+1/n)\mathbf{p}$ and Lemma 2.3 we have $\mathsf{OPT}(\mathcal{I}_{\bar{p}}) \le (1+1/n)^3\mathsf{OPT}(\mathcal{I}_p)$. So we obtain $\mathsf{OPT}(\mathcal{I}_q) \le (1+1/n)^3 \cdot \mathsf{OPT}(\mathcal{I})$.

For $\alpha = \frac{e}{e-1}(1+\frac{1}{n})$, we will show that $\mathsf{OPT}(\mathcal{J}) \le \alpha \cdot \mathsf{OPT}(\mathcal{I}_q)$ which would prove the lemma. Recall that instance $\mathcal{J}$ is defined on the "scaled" vertex set $\widehat{V} = \cup_{v\in V} S_v$. Let π be an optimal master tour for instance $\mathcal{I}_q$ and $\widehat{\pi}$ be its corresponding master tour for $\mathcal{J}$: i.e. $\widehat{\pi}$ visits each group S_v consecutively at the point when π visits v. It suffices to show that the expected latency $\mathsf{ELAT}_{\widehat{\pi}}$ of tour $\widehat{\pi}$ for $\mathcal{J}$ is at most $\alpha \cdot \mathsf{ELAT}_\pi$, where ELAT_π is the expected latency of tour π for $\mathcal{I}_q$.

Let $A \subseteq V$ and $\widehat{A} \subseteq \widehat{V}$ denote the random active subsets in the instances $\mathcal{I}_q$ and $\mathcal{J}$ respectively. For any $v \in V$, let $\mathcal{E}_v$ denote the event that $S_v \cap \widehat{A} \neq \emptyset$; note that these events are independent. Moreover, for any $v \in V$, $\Pr_{\widehat{A}}[\mathcal{E}_v] = \Pr_{\widehat{A}}[S_v \cap \widehat{A} \neq \emptyset] = q_v = \Pr_A[v \in A]$. Let $\mathsf{ELAT}_{\widehat{\pi}}(w) = \mathbb{E}_A\left[\sum_{v\in S_w} \mathsf{LAT}^A_{\widehat{\pi}}(v)\right]$ denote the total expected latency of vertices of S_w in tour $\widehat{\pi}$. Fix any vertex $w \in V$: we will show that $\mathsf{ELAT}_{\widehat{\pi}}(w)$ is at most $\alpha \cdot \mathsf{ELAT}_\pi(w)$, where $\mathsf{ELAT}_\pi(w)$ is the expected latency of vertex w in π. Summing over $w \in V$, this would imply $\mathsf{ELAT}_{\widehat{\pi}} \leq \alpha \cdot \mathsf{ELAT}_\pi$, and hence $\mathsf{OPT}(\mathcal{J}) \leq \alpha \cdot \mathsf{OPT}(\mathcal{I}_q)$.

Consider now a fixed $w \in V$. Note that the probability distribution of the vertices in $V \setminus \{w\}$ whose groups (in $\widehat{V}$) have at least one vertex in $\widehat{A}$ is *identical* to that of $A \setminus \{w\}$. In other words, the random subset $\{v \in V \setminus \{w\} : \mathcal{E}_v \text{ occurs}\}$ has the same distribution as random subset $A\setminus\{w\}$. Below, we couple these two distributions: We *condition* on the events $\mathcal{E}_v$ for all $v \in V\setminus\{w\}$ (for tour $\widehat{\pi}$) which corresponds to conditioning on $A\setminus\{w\}$ being active (for tour π). Under this conditioning (denoted $\mathcal{E}$), the latency of any active S_w vertex in $\widehat{\pi}$ is deterministic and equal to the latency of w (if it is active) in π; let $L(\pi, w \mid \mathcal{E})$ denote this deterministic value. So the conditional expected latency of w is $L(\pi, w \mid \mathcal{E}) \cdot \Pr[w \in A] = L(\pi, w|\mathcal{E}) \cdot q_w$ where we used the independence of $A \setminus \{w\}$ and the event $w \in A$. Similarly, the total conditional expected latency of S_w in $\widehat{\pi}$ is

$$L(\pi, w|\mathcal{E}) \cdot \mathbb{E}[|\widehat{A} \cap S_w|] = L(\pi, w|\mathcal{E}) \cdot (pt_w) \leq L(\pi, w|\mathcal{E}) \cdot (p_w + p) .$$

The equality above uses the independence of $\{\mathcal{E}_v : v \in V \setminus \{w\}\}$ and $\widehat{A} \cap S_w$, and the inequality uses $t_w = \lceil p_w/p \rceil$. Thus, the total conditional expected latency of S_w in $\widehat{\pi}$ is at most $\frac{p_w+p}{q_w}$ times the conditional expected latency of w in π. Deconditioning, we obtain $\mathsf{ELAT}_{\widehat{\pi}}(w) \leq \frac{p_w+p}{q_w} \cdot \mathsf{ELAT}_\pi(w)$. Using Lemma 2.1, $\frac{p_w+p}{q_w} \leq \frac{e}{e-1}(1 + p/p_w) \leq \frac{e}{e-1}(1 + 1/n) = \alpha$. So $\mathsf{LAT}_{\widehat{\pi}}(w) \leq \alpha \cdot \mathsf{LAT}_\pi(w)$ as needed. □

Lemma 2.5. *Consider any consecutive master tour $\widehat{\pi}$ on instance $\mathcal{J}$ with expected latency $\mathsf{ALG}(\mathcal{J})$. Then the expected latency of the resulting master tour π on instance $\mathcal{I}$ is*

$$\mathsf{ALG}(\mathcal{I}) \leq \left(\frac{e}{e-1}\right)^3 \left(1 + \frac{1}{n}\right)^3 \cdot \mathsf{ALG}(\mathcal{J}) .$$

Proof. Let $\mathsf{ALG}(\mathcal{I}_p)$, $\mathsf{ALG}(\mathcal{I}_q)$ and $\mathsf{ALG}(\mathcal{I}_{\bar{p}})$ denote the expected latency of master tour π under probabilities $\mathbf{p}$, $\mathbf{q}$ and $\bar{\mathbf{p}}$ respectively. Below we use $\alpha = \frac{e}{e-1}(1 + \frac{1}{n})$. Using $\mathbf{p} \leq \bar{\mathbf{p}}$ and Lemma 2.2 we have $\mathsf{ALG}(\mathcal{I}_p) \leq \mathsf{ALG}(\mathcal{I}_{\bar{p}})$. Using $\frac{1}{\alpha} \cdot \bar{\mathbf{p}} \leq \mathbf{q} \leq \bar{\mathbf{p}}$ (Lemma 2.1) and Lemma 2.3, we have $\mathsf{ALG}(\mathcal{I}_{\bar{p}}) \leq \alpha^3 \cdot \mathsf{ALG}(\mathcal{I}_q)$. Combining these bounds, we have $\mathsf{ALG}(\mathcal{I}) \leq \alpha^3 \cdot \mathsf{ALG}(\mathcal{I}_q)$. Finally, it is easy to see that $\mathsf{ALG}(\mathcal{I}_q) \leq \mathsf{ALG}(\mathcal{J})$ as the probability of having at least one active vertex in group S_v (for any $v \in V$) in $\mathcal{J}$ is exactly equal the probability (q_v) of visiting v in $\mathcal{I}_q$. □

2.3. Overall analysis including vertices in Y

Now we have the tools to finish the proof of Theorem 1.2 assuming the tour $\widehat{\pi}$ in $\mathcal{J}$ is consecutive. Recall that π is the tour corresponding to $\widehat{\pi}$ on vertices X and $\bar{\pi}$ is the extended tour that also visits the vertices Y.

First, the analysis for the vertices X (Lemmas 2.4 and 2.5) yields:

Corollary 2.5.1. *The tour π on vertices X satisfies*

$$\mathbb{E}_A\left[\sum_{v\in A\cap X} \mathsf{LAT}^A_\pi(v)\right] \leq (1 + o(1)) \left(\frac{e}{e-1}\right)^4 \rho \cdot \mathsf{OPT}_X ,$$

where ρ is the approximation ratio for the uniform a priori *TRP used in Algorithm 1 and OPT_X is the optimal value of the instance restricted to vertices X.*

After extending tour π to $\bar{\pi}$, we can write the final expected latency as

$$\begin{aligned}\mathsf{ALG}(\mathcal{I}) &= \mathbb{E}_A\left[\sum_{v\in A\cap X} \mathsf{LAT}^A_{\bar{\pi}}(v) + \sum_{v\in A\cap Y} \mathsf{LAT}^A_{\bar{\pi}}(v)\right] \\ &= \mathbb{E}_A\left[\sum_{v\in A\cap X} \mathsf{LAT}^A_\pi(v)\right] + \mathbb{E}_A\left[\sum_{v\in A\cap Y} \mathsf{LAT}^A_{\bar{\pi}}(v)\right] \qquad (1)\end{aligned}$$

where $A \subseteq V$ is the active subset. The last equality uses the fact that $\bar{\pi}$ visits all vertices of X (along π) before Y. The first term above can be bounded by Corollary 2.5.1. We now focus on the second term involving vertices Y.

Let z denote the last X-vertex visited in tour $\bar{\pi}_A$ and let L denote the length of tour $\bar{\pi}_A$ until vertex z. Note that z and L are random variables. Clearly $\mathbb{E}_A[L]$ is at most the expected total latency of the X-vertices. Consider any $v \in Y$, and let N_v denote the number of active Y-vertices appearing before v. By the ordering of Y-vertices in master tour $\bar{\pi}$ and triangle inequality,

$$\mathsf{LAT}^A_{\bar{\pi}}(v) \leq L + d(z, v) + 2N_v \cdot d(r, v) \leq (2L + (2N_v + 1) \cdot d(r, v)) \cdot \mathbf{1}_{v\in A} ,$$

where the second inequality uses $d(z, v) \leq L + d(r, v)$ which follows from symmetry and triangle inequality. Taking expectations,

$$\begin{aligned}\mathbb{E}_A\left[\mathsf{LAT}^A_{\bar{\pi}}(v)\right] &\leq p_v \cdot \mathbb{E}_A[2L] + p_v \cdot d(r, v) \cdot (2\mathbb{E}_A[N_v] + 1) \\ &\leq p_v \cdot \mathbb{E}_A[2L] + p_v \cdot d(r, v) \cdot (2n \cdot \frac{1}{n^2} + 1) \\ &= p_v \cdot \mathbb{E}_A[2L] + p_v \cdot d(r, v) \cdot (1 + o(1)),\end{aligned}$$

The first inequality uses the fact that L, N_v and $\mathbf{1}_{v\in A}$ are independent. The second inequality uses that N_v is the sum of at most n Bernoulli random variables each with probability at most $\frac{1}{n^2}$.

Summing over all $v \in Y$, we obtain

$$\begin{aligned}\mathbb{E}_A\left[\sum_{v\in A\cap Y} \mathsf{LAT}^A_{\bar{\pi}}(v)\right] &\leq \left(\sum_{v\in Y} p_v\right) \cdot \mathbb{E}_A[2L] + (1 + o(1)) \sum_{v\in Y} p_v \cdot d(r, v) \\ &\leq \frac{2}{n} \cdot \mathbb{E}_A[L] + (1 + o(1)) \sum_{v\in Y} p_v \cdot d(r, v) ,\end{aligned}$$

where the last inequality uses $p_v \leq 1/n^2$ for all $v \in Y$.

Let E_X denote the expected latency of the X-vertices: this is the first term in the right-hand-side of (1). Recall that $\mathbb{E}_A[L] \leq E_X$. Using the above bound on the latency of Y-vertices,

$$\begin{aligned}\mathsf{ALG}(\mathcal{I}) &\leq E_X + \frac{2}{n} \cdot E_X + (1 + o(1)) \sum_{v\in Y} p_v \cdot d(r, v) \\ &= (1 + o(1)) \left(E_X + \sum_{v\in Y} p_v \cdot d(r, v)\right) \\ &\leq (1 + o(1)) \left(\frac{e}{e-1}\right)^4 \rho \cdot \left(\mathsf{OPT}_X + \sum_{v\in Y} p_v \cdot d(r, v)\right) \qquad (2) \\ &\leq (1 + o(1)) \left(\frac{e}{e-1}\right)^4 \rho \cdot \mathsf{OPT}. \qquad (3)\end{aligned}$$

Above, inequality (2) uses Corollary 2.5.1. Inequality (3) uses the fact that the latency contribution of Y-vertices in *any* master tour is at least $\sum_{v\in Y} p_v \cdot d(r, v)$ and the latency of X-vertices is clearly at least OPT_X. This completes the proof of Theorem 1.2 assuming that $\widehat{\pi}$ visits each group S_v consecutively. The next subsection shows that this consecutive property can always be ensured.

2.4. Ensuring the consecutive property

The main result here is:

Theorem 2.6. *Consider any instance $\mathcal{J}$ of uniform* a priori *TRP on vertices $\cup_{v\in X} S_v$ where the vertices in S_v are co-located for all $v \in X$. There is a polynomial time algorithm that given any master tour τ, modifies it into a consecutive tour having expected latency at most that of τ.*

Note that an optimal TRP solution can be fairly complicated even on simple metrics: for example, the optimum may cross itself several times on a line-metric [1] and the problem is NP-hard even on tree-metrics [20]. In Appendix A we provide some examples that motivate why we need a non-trivial algorithm to ensure the consecutive property.

Algorithm 2 describes the procedure used to establish Theorem 2.6. We use Π to denote the distribution of active vertices, where each vertex has independent probability p. It is obvious that each iteration of the while-loop decreases the number k of parts of S_z: so the number of iterations is polynomial. Moreover, the expected latency of any tour can be calculated exactly (using the expression in Lemma 2.3). So all the steps run in polynomial time. The key part of the proof is in showing that the expected latency does not increase, which is done in Lemma 2.7.

Algorithm 2 Algorithm to obtain a consecutive master tour.

```
Procedure MAKECONSECUTIVE(τ):
for z ∈ V do
    Let C_z^1, C_z^2, ..., C_z^k be the minimal partition of S_z, where for
    every i ∈ [k], the vertices in C_z^i appear consecutively in tour τ.
    while there exist C_z^i and C_z^j with i ≠ j do
        Construct tour τ_i from τ by relocating vertices C_z^j immediately after C_z^i
        Construct tour τ_j from τ by relocating vertices C_z^i immediately before C_z^j
        τ ← argmin_{τ∈{τ_i,τ_j}} E_A[LAT_τ^A]
        Update k ← k − 1 and the new partition of S_z.
    end while
end for
```

Lemma 2.7. *Let C_z^i and C_z^j be two parts of S_z with respect to the current tour τ in procedure* MAKECONSECUTIVE. *Then we have:*

$$\mathbb{E}_A\left[\mathsf{LAT}_\tau^A\right] \geq \min(\mathbb{E}_A\left[\mathsf{LAT}_{\tau_i}^A\right], \mathbb{E}_A\left[\mathsf{LAT}_{\tau_j}^A\right]) .$$

Proof. Let $|C_z^i| = k_i$ and $|C_z^j| = k_j$. Without loss of generality we assume that τ visits C_z^i before C_z^j. To reduce notation we use V to denote the vertex set of instance $\mathcal{J}$ and let $U = C_z^i \cup C_z^j$. Recall that $\mathsf{LAT}_\pi^A(w)$ is the latency of vertex w in tour π when the subset A of vertices is active; also $\mathsf{LAT}_\pi^A = \sum_{w\in A} \mathsf{LAT}_\pi^A(w)$. For any $R \subseteq V$ and $S \subseteq R$ we use the notation $p(S, R) = p^{|S|} \cdot (1-p)^{|R\setminus S|}$ for the probability that S is the set of active vertices amongst R.

It suffices to show $\mathbb{E}_A\left[\mathsf{LAT}_\tau^A\right]$ is at least a convex combination of $\mathbb{E}_A\left[\mathsf{LAT}_{\tau_i}^A\right]$ and $\mathbb{E}_A\left[\mathsf{LAT}_{\tau_j}^A\right]$. More specifically we show that:

$$\mathbb{E}_A\left[\mathsf{LAT}_\tau^A\right] \geq \lambda \cdot \mathbb{E}_A\left[\mathsf{LAT}_{\tau_i}^A\right] + (1-\lambda) \cdot \mathbb{E}_A\left[\mathsf{LAT}_{\tau_j}^A\right] .$$

where $\lambda \in [0, 1]$ is a value that will be set later. The above inequality is equivalent to proving the following:

$$\sum_{A\subseteq V} p(A, V)\mathsf{LAT}_\tau^A \geq \sum_{A\subseteq V} p(A, V)\left(\lambda \cdot \mathsf{LAT}_{\tau_i}^A + (1-\lambda)\cdot \mathsf{LAT}_{\tau_j}^A\right) .$$

Let us define $B = A \setminus U$ and $C = A \cap U$. Basically C is the subset of active vertices among $U = C_z^i \cup C_z^j$, and B is the subset of active vertices among the rest of V. Then by independence of probabilities we can re-write the above inequality as follows:

$$\sum_{B\subseteq V\setminus U} p(B, V\setminus U) \sum_{C\subseteq U} p(C, U)\mathsf{LAT}_\tau^{B\cup C}$$
$$\geq \sum_{B\subseteq V\setminus U} p(B, V\setminus U) \sum_{C\subseteq U} p(C, U)\left(\lambda \cdot \mathsf{LAT}_{\tau_i}^{B\cup C} + (1-\lambda)\cdot \mathsf{LAT}_{\tau_j}^{B\cup C}\right) .$$

Therefore, it is enough to prove

$$\sum_{C\subseteq U} p(C, U)\mathsf{LAT}_\tau^{B\cup C} \geq \sum_{C\subseteq U} p(C, U)\left(\lambda \cdot \mathsf{LAT}_{\tau_i}^{B\cup C} + \times (1-\lambda)\cdot \mathsf{LAT}_{\tau_j}^{B\cup C}\right), \quad \forall B \subseteq V\setminus U. \tag{4}$$

In the rest of this proof we fix a subset $B \subseteq V \setminus U$. This can be viewed as conditioning on the event "B is the active set of vertices within $V \setminus U$"; we denote this event by $\mathcal{E}_B$. Let the order of visited vertices of $B \cup U$ in τ be $B_1, C_z^i, B_2, C_z^j, B_3$ where B_1, B_2, B_3 are ordered sets of vertices that form a partition of B. Therefore, together with C_z^i and C_z^j they form a partition of $B \cup U$. See Fig. 1 for an example.

If $B_2 = \emptyset$ then all three tours τ, τ_i and τ_j become identical when restricted to $B \cup C$ for any $C \subseteq U$. So (4) is satisfied with an equality in this case. Below we assume $B_2 \neq \emptyset$. We will prove the inequality (4) by considering the latency contributions of vertices in each of the 5 different parts $B_1, B_2, B_3, C_z^i, C_z^j$.

We define $l_w := \mathsf{LAT}_\tau^{B\cup\{w\}}(w)$ for all $w \in V$ and

$$T_i := \mathsf{LAT}_\tau^{B\cup C_z^i}(w) \quad \forall w \in C_z^i, \quad \text{and} \quad T_j := \mathsf{LAT}_\tau^{B\cup C_z^j}(w) \quad \forall w \in C_z^j. \tag{5}$$

Basically T_i (resp. T_j) is the length of the path in τ from the root to any vertex in C_z^i (resp. C_z^j) when the active vertices are $B \cup C_z^i$ (resp. $B \cup C_z^j$). Note that $T_j \geq T_i$ by triangle inequality. Also, let $L_\pi^B(w)$ be the expected latency of any vertex w for any tour $\pi \in \{\tau, \tau_i, \tau_j\}$ conditioned on the event $\mathcal{E}_B$. More formally:

$$L_\pi^B(w) = \sum_{C\subseteq U} p(C, U)\mathsf{LAT}_\pi^{B\cup C}(w), \qquad \forall w \in V .$$

Finally, defining the following terms will help us simplify our notation:

$$\Delta_i := \mathsf{LAT}_\tau^{B\cup C_z^i}(w) - \mathsf{LAT}_\tau^B(w) = \mathsf{LAT}_\tau^{B\cup C_z^i}(w) - l_w \quad \forall w \in B_2 \cup B_3. \tag{6}$$

$$\Delta_j := \mathsf{LAT}_\tau^{B\cup C_z^j}(w) - \mathsf{LAT}_\tau^B(w) = \mathsf{LAT}_\tau^{B\cup C_z^j}(w) - l_w \qquad \forall w \in B_3. \tag{7}$$

Note that Δ_i (resp. Δ_j) corresponds to the increase in latency (conditioned on $\mathcal{E}_B$) of any B-vertex appearing after C_z^i (resp. C_z^j) if at least one vertex in C_z^i (resp. C_z^j) is active. Note that the right hand side in (6) is the same for any w in the given set and as a result independent of w; the same observation is true for (7). Moreover, by triangle inequality, having a superset of active vertices can only increase the latency of any vertex: so Δ_i and Δ_j are non-negative.

Table 1 lists the expected latency of vertices in each of the five different parts, conditioned on $\mathcal{E}_B$. We use $\alpha_i = 1 - (1-p)^{k_i}$ and

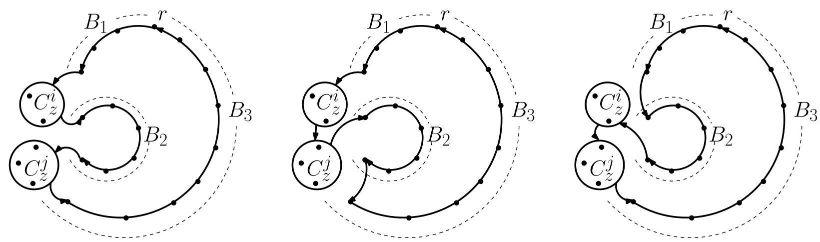

Fig. 1. From left to right: Tours τ, τ_i and τ_j.

$\alpha_j = 1-(1-p)^{k_j}$ as the probabilities of having at least one active vertex in parts C_z^i and C_z^j respectively.

We first prove the lemma assuming the entries stated in the table. Then we explain why each of these table entries is correct, which would complete the proof.

2.4.1. Completing proof of Lemma 2.7 using Table 1

We now prove (4) for a suitable choice of $\lambda \in [0, 1]$. The value λ will not depend on the subset B: so (as discussed before) we can take an expectation over B to complete the proof of the lemma.

Choosing any λ such that $\lambda \le \frac{\alpha_i}{\alpha_i+\alpha_j-\alpha_i\alpha_j}$ and $1-\lambda \le \frac{\alpha_j}{\alpha_i+\alpha_j-\alpha_i\alpha_j}$, it follows from the first three columns of Table 1 (for B_1, B_2 and B_3) that:

$$L^B_\tau(w) \ge \lambda \cdot L^B_{\tau_i}(w) + (1-\lambda) \cdot L^B_{\tau_j}(w), \qquad \forall w \in B. \tag{8}$$

Next we show that the *total* latency contribution from U satisfies a similar inequality:

$$\sum_{w\in U} L^B_\tau(w) \ge \lambda \cdot \sum_{w\in U} L^B_{\tau_i}(w) + (1-\lambda) \cdot \sum_{w\in U} L^B_{\tau_j}(w). \tag{9}$$

To see this, note from the last two columns of the table that

$$\sum_{w\in U} L^B_\tau(w) \ge k_i \cdot T_i p + k_j \cdot T_j p, \qquad \sum_{w\in U} L^B_{\tau_i}(w) = (k_i+k_j)T_i p,$$
$$\sum_{w\in U} L^B_{\tau_j}(w) = (k_i+k_j)T_j p\ .$$

So, to prove (9) it suffices to show $k_iT_ip + k_jT_jp \ge (k_i+k_j)(\lambda T_i + (1-\lambda)T_j)p$. Using the fact that $T_i \le T_j$, it suffices to show $k_j \ge (k_i+k_j)(1-\lambda)$. In other words, choosing λ such that $1-\lambda \le \frac{k_j}{k_i+k_j}$, we would obtain (9).

Finally, adding the inequalities (8) and (9) (which account for the latency contribution from all active vertices) we would obtain (4). We only need to ensure that there is some choice for λ satisfying the conditions we assumed, namely:

$$\lambda \le \frac{\alpha_i}{\alpha_i+\alpha_j-\alpha_i\alpha_j}, \quad 1-\lambda \le \frac{\alpha_j}{\alpha_i+\alpha_j-\alpha_i\alpha_j}, \quad \text{and}$$
$$1-\lambda \le \frac{k_j}{k_i+k_j}\ .$$

It can be verified directly that $\lambda = \frac{1-(1-p)^{k_i}}{1-(1-p)^{k_i+k_j}}$ satisfies these conditions (see Appendix B).

2.4.2. Obtaining the entries in Table 1

Below we consider each vertex-type separately.

Vertices $w \in B_1$. By construction of τ_i and τ_j it is obvious that τ, τ_i and τ_j are identical until visiting any $w \in B_1$. So for any $C \subseteq U$ and $\pi \in \{\tau, \tau_i, \tau_j\}$ we have $\mathsf{LAT}^{B\cup C}_\pi(w) = \mathsf{LAT}^B_\tau(w) = \mathsf{LAT}^{B\cup\{w\}}_\tau(w) = l_w$. This means that $L^B_\pi(w) = l_w$ for all $\pi \in \{\tau, \tau_i, \tau_j\}$.

Vertices $w \in B_2$. Consider first tour τ. Note that if there is at least one active vertex in C_z^i (which happens with probability α_i) then the latency of any $w \in B_2$ will be $\mathsf{LAT}^{B\cup C_z^i}_\tau(w)$. However, if all vertices in C_z^i are inactive (which happens with probability $1-\alpha_i$) then the latency of w would be $\mathsf{LAT}^B_\tau(w)$. Now using (6) we have:

$$\begin{aligned} L^B_\tau(w) &= \mathsf{LAT}^{B\cup C_z^i}_\tau(w)\cdot\alpha_i + l_w\cdot(1-\alpha_i) \\ &= (l_w + \Delta_i)\cdot\alpha_i + l_w\cdot(1-\alpha_i) = l_w + \Delta_i\alpha_i\ . \end{aligned}$$

Now, we can use a similar logic for τ_i. Here, if there is any active vertex in $U = C_z^i \cup C_z^j$ (with probability $\alpha_i+\alpha_j-\alpha_i\alpha_j$) the latency of w is $\mathsf{LAT}^{B\cup U}_{\tau_i}(w)$, and if all of U is inactive the latency is l_w. Note that by definition of τ and τ_i and the fact that all vertices in C_z^i appear consecutively on both tours, $\mathsf{LAT}^{B\cup U}_{\tau_i}(w) = \mathsf{LAT}^{B\cup C_z^i}_{\tau_i}(w) = \mathsf{LAT}^{B\cup C_z^i}_{\tau}(w)$. So we have $L^B_{\tau_i} = l_w + \Delta_i(\alpha_i+\alpha_j-\alpha_i\alpha_j)$.

Finally, by definition of τ_j we have $\mathsf{LAT}^{B\cup C}_{\tau_j}(w) = \mathsf{LAT}^B_\tau(w) = l_w$ for any $C \subseteq U$. So $L^B_{\tau_j}(w) = l_w$.

Vertices $w \in B_3$. Consider first tour τ. The latency of such a vertex w is:

- l_w if all of $U = C_z^i \cup C_z^j$ is inactive,
- $\mathsf{LAT}^{B\cup C_z^i}_\tau(w)$ if some vertex in C_z^i is active and all of C_z^j is inactive,
- $\mathsf{LAT}^{B\cup C_z^j}_\tau(w)$ if some vertex in C_z^j is active and all of C_z^i is inactive, and
- $\mathsf{LAT}^{B\cup C_z^i\cup C_z^j}_\tau(w)$ if some vertex in C_z^i and some vertex in C_z^j are active.

Therefore, we can write:

$$\begin{aligned} L^B_\tau(w) &= l_w(1-\alpha_i)(1-\alpha_j) + \mathsf{LAT}^{B\cup C_z^i}_\tau(w)\alpha_i(1-\alpha_j) \\ &\quad + \mathsf{LAT}^{B\cup C_z^j}_\tau(w)\alpha_j(1-\alpha_i) + \mathsf{LAT}^{B\cup U}_\tau(w)\alpha_i\alpha_j\ . \end{aligned}$$

From (6) and (7) we have $\mathsf{LAT}^{B\cup C_z^i}_\tau = l_w + \Delta_i$ and $\mathsf{LAT}^{B\cup C_z^j}_\tau = l_w + \Delta_j$. Also, since we assumed that $B_2 \ne \emptyset$, we have $\mathsf{LAT}^{B\cup U}_\tau = l_w + \Delta_j + \Delta_i$. Combined with the above equation,

$$L^B_\tau(w) = l_w + \Delta_i\alpha_i + \Delta_j\alpha_j\ .$$

Now for tour τ_i the latency would be equal to $\mathsf{LAT}^{B\cup C_z^i}_\tau(w) = l_w + \Delta_i$ if there is at least one active vertex among U which happens with probability $\alpha_i+\alpha_j-\alpha_i\alpha_j$. Otherwise it would be just l_w. So $L^B_{\tau_i} = l_w + \Delta_i(\alpha_i+\alpha_j-\alpha_i\alpha_j)$. Similarly, for tour τ_j we have $L^B_{\tau_j} = l_w + \Delta_j(\alpha_i+\alpha_j-\alpha_i\alpha_j)$.

Vertices $w \in C_z^i$. We start with tour τ. If $w \notin C$ then $\mathsf{LAT}^{B\cup C}_\tau(w) = 0$. Otherwise, w is active and using (5) we have $\mathsf{LAT}^{B\cup C}_\tau(w) = \mathsf{LAT}^{B\cup C_z^i}_\tau(w) = T_i$. So $L^B_\tau(w) = T_ip$.

As C_z^i appears in the same position in tours τ and τ_i, we also have $L^B_{\tau_i}(w) = T_ip$.

In tour τ_j, part C_z^i has moved to the position of part C_z^j in τ. Here, when $w \in C$ we have $\mathsf{LAT}^{B\cup C}_{\tau_j}(w) = T_j$. So $L^B_{\tau_j}(w) = T_jp$.

Vertices $w \in C_z^j$. As in the previous case, we have $L^B_{\tau_i}(w) = T_ip$ and $L^B_{\tau_j}(w) = T_jp$.

Table 1
The values of $L^B_\pi(w)$ for $w \in B_1 \cup B_2 \cup B_3 \cup C^i_z \cup C^j_z$, and $\pi \in \{\tau, \tau_i, \tau_j\}$.

Tour π	Type				
	B_1	B_2	B_3	C^i_z	C^j_z
τ	l_w	$l_w + \Delta_i\alpha_i$	$l_w + \Delta_i\alpha_i + \Delta_j\alpha_j$	T_ip	$T_jp + \Delta_i\alpha_i p$
τ_i	l_w	$l_w + \Delta_i(\alpha_i + \alpha_j - \alpha_i\alpha_j)$	$l_w + \Delta_i(\alpha_i + \alpha_j - \alpha_i\alpha_j)$	T_ip	T_ip
τ_j	l_w	l_w	$l_w + \Delta_j(\alpha_i + \alpha_j - \alpha_i\alpha_j)$	T_jp	T_jp

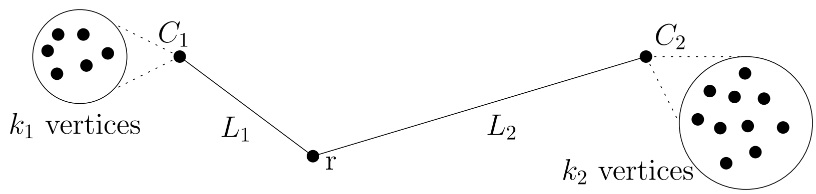

Fig. 2. Examples without consecutive property.

Now, consider tour τ. First note that if $w \notin C$, $\mathsf{LAT}^{B\cup C}_\tau(w) = 0$. Below we consider the cases that w is active, which happens with probability p. If there is at least one active vertex in C^i_z (which happens independently with probability α_i) we have $\mathsf{LAT}^{B\cup C}_\tau(w) = l_w + \Delta_i = T_j + \Delta_i$. And if there is no active vertex in C^i_z (with probability $1 - \alpha_i$), then we have $\mathsf{LAT}^{B\cup C}_\tau(w) = l_w = T_j$. So

$$L^B_\tau(w) = p\alpha_i \cdot (T_j + \Delta_i) + p(1 - \alpha_i) \cdot T_j = T_jp + \Delta_i\alpha_i p\ .$$

This completes the proof of all cases in Table 1, and hence Lemma 2.7. □

Acknowledgments

V. Nagarajan and F. Navidi were supported in part by NSF, USA grants CCF-1750127 and CMMI-1940766. I. L. Gørtz was supported in part by the Danish Research Council, Denmark grant DFF – 1323-00178.

Appendix A. Useful examples for the consecutive property

Here we provide two examples that motivate why we need an algorithm for ensuring the consecutive property. Both examples involve a metric consisting of two clusters, cluster C_1 and C_2 that have k_1 and k_2 vertices and are at L_1 and L_2 distance from root; see Fig. 2. They differ in the values of these parameters.

Example 1. The algorithm from [9] does not necessarily produce a consecutive tour. Recall that the algorithm for uniform probabilities [9] is based on concatenating a sequence of *a priori TSP approximators*. Given a bound L, a this involves finds a tour visiting the maximum number of vertices with expected length at most L. (Even a bicriteria approximation to this suffices.) We provide an instance where an optimal TSP approximator visits clusters partially. In Fig. 2, let $k_1 = 1$, $L_1 = 1$, $L_2 = \frac{1}{p}$, $k_2 = \frac{2}{p}$ and bound $L = 2(1 - 1/e) \cdot \frac{1}{p}$. The optimal TSP approximator visits $\frac{1}{p} = k_2/2$ vertices from C_2 and has expected length $2L_2\left(1 - (1 - p)^{1/p}\right) \to L$ as $p \to 0$. So, if this were found as an approximate solution then the resulting tour for *a priori* TRP is not consecutive.

Example 2. An obvious way to make a tour consecutive is to visit each cluster completely at the point when one of its vertices is first visited. We show that this can lead to a large increase in expected latency. Let $p = \frac{1}{\alpha}$, $k_2 = \alpha$, $k_1 = \alpha^2$, $L_1 = \alpha$ and $L_2 = \alpha^2$, where $\alpha \approx \sqrt{n}$. Consider the non-consecutive tour τ that visits one vertex from C_2, then visits all of C_1 and returns to visit the remaining vertices in C_2. Let τ' be the tour that is obtained by the algorithm that visits a cluster completely upon its first visit. So τ' visits all of C_2 and then all of C_1. One can check that $\mathsf{ELAT}_\tau = O(\alpha^2)$ and $\mathsf{ELAT}_{\tau'} = \Omega(\alpha^3)$, implying a multiplicative gap of $\Omega(\alpha) = \Omega(\sqrt{n})$.

Appendix B. Choice of λ in proof of Lemma 2.7

Here we show that $\lambda = \frac{1-(1-p)^{k_i}}{1-(1-p)^{k_i+k_j}}$ (where $0 \le p \le 1$) satisfies the following inequalities:

$$\lambda \le \frac{\alpha_i}{\alpha_i + \alpha_j - \alpha_i\alpha_j} \tag{10}$$

$$1 - \lambda \le \frac{\alpha_j}{\alpha_i + \alpha_j - \alpha_i\alpha_j} \tag{11}$$

$$1 - \lambda \le \frac{k_j}{k_i + k_j} \tag{12}$$

where $\alpha_i = 1 - (1 - p)^{k_i}$ and $\alpha_j = 1 - (1 - p)^{k_j}$.

We define function $f(k) = 1 - (1 - p)^k$. Then we can write:

$$\lambda = \frac{f(k_i)}{f(k_i + k_j)}, \quad \alpha_i = f(k_i), \quad \alpha_j = f(k_j)$$

Clearly,

$$f(k_i + k_j) = f(k_i) + f(k_j) - f(k_i)f(k_j) \tag{13}$$

Now, we can re-write inequality (10) as:

$$\frac{f(k_i)}{f(k_i + k_j)} \le \frac{f(k_i)}{f(k_i) + f(k_j) - f(k_i)f(k_j)}$$

which is true by Eq. (13).

For inequality (11), we rewrite it as:

$$1 - \frac{f(k_i)}{f(k_i + k_j)} \le \frac{f(k_j)}{f(k_i) + f(k_j) - f(k_i)f(k_j)} = \frac{f(k_j)}{f(k_i + k_j)}$$

$$\Leftrightarrow \quad f(k_i + k_j) \le f(k_i) + f(k_j)\ ,$$

which is true by (13) and the fact that $f(k) \ge 0$ for every k.

It remains to show the correctness of inequality (12) which can be written as:

$$\frac{f(k_i)}{f(k_i + k_j)} \ge \frac{k_i}{k_i + k_j} \quad \Leftrightarrow \quad \frac{f(k_i)}{k_i} \ge \frac{f(k_i + k_j)}{k_i + k_j}\ .$$

So it is enough to show that $g(k) = \frac{f(k)}{k}$ is decreasing, or equivalently $g'(k) \le 0$. We can write:

$$g'(k) = \frac{kf'(k) - f(k)}{k^2} = \frac{(1 - p)^k(1 - k\log(1 - p)) - 1}{k^2}$$
$$\le \frac{(1 - p)^k \cdot e^{-k\log(1-p)} - 1}{k^2} = 0\ .$$

Above we used the inequality $1 + x \le e^x$ for all real x.

References

[1] Foto N. Afrati, Stavros S. Cosmadakis, Christos H. Papadimitriou, George Papageorgiou, Nadia Papakostantinou, The complexity of the travelling repairman problem, Theor. Inform. Appl. 20 (1) (1986) 79–87.
[2] Sanjeev Arora, George Karakostas, Approximation schemes for minimum latency problems, SIAM J. Comput. 32 (5) (2003) 1317–1337.
[3] Dimitris Bertsimas, Probabilistic Combinatorial Optimization Problems (Ph.D. thesis), Massachusetts Institute of Technology, 1988.
[4] Dimitris J. Bertsimas, A vehicle routing problem with stochastic demand, Oper. Res. 40(3) (1992) 574–585.
[5] Dimitris J. Bertsimas, Patrick Jaillet, Amedeo R. Odoni, A priori optimization, Oper. Res. 38 (6) (1990) 1019–1033.
[6] Avrim Blum, Prasad Chalasani, Don Coppersmith, Bill Pulleyblank, Prabhakar Raghavan, Madhu Sudan, The minimum latency problem, in: Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, ACM, 1994, pp. 163–171.
[7] A.M. Campbell, B.W. Thomas, Probabilistic traveling salesman problem with deadlines, Transp. Sci. 42 (1) (2008) 1–21.
[8] Kamalika Chaudhuri, Brighten Godfrey, Satish Rao, Kunal Talwar, Paths, trees, and minimum latency tours, in: Proceedings of the 44th Symposium on Foundations of Computer Science, FOCS, 2003, pp. 36–45.
[9] Martijn van Ee, René Sitters, The a priori traveling repairman problem, Algorithmica 80 (10) (2018) 2818–2833.
[10] A.L. Erera, M.W.P. Savelsbergh, E. Uyar, Fixed routes with backup vehicles for stochastic vehicle routing problems with time constraints, Networks 54 (4) (2009) 270–283.
[11] Matteo Fischetti, Gilbert Laporte, Silvano Martello, The delivery man problem and cumulative matroids, Oper. Res. 41 (6) (1993) 1055–1064.
[12] Michel Goemans, Jon Kleinberg, An improved approximation ratio for the minimum latency problem, Math. Program. 82 (1–2) (1998) 111–124.
[13] Igor Gorodezky, Robert D Kleinberg, David B Shmoys, Gwen Spencer, Improved lower bounds for the universal and a priori tsp, in: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, Springer, 2010, pp. 178–191.
[14] Anupam Gupta, Viswanath Nagarajan, R. Ravi, Approximation algorithms for VRP with stochastic demands, Oper. Res. 60 (1) (2012) 123–127.
[15] P. Jaillet, Probabilistic Traveling Salesman Problems (Ph.D. thesis), Massachusetts Institute of Technology, 1985.
[16] Jean-Claude Picard, Maurice Queyranne, The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling, Oper. Res. 26 (1) (1978) 86–110.
[17] M.W.P. Savelsbergh, M. Goetschalkx, A comparison of the efficiency of fixed versus variable vehicle routes, J. Bus. Logist. 16 (1995) 163–187.
[18] Frans Schalekamp, David B. Shmoys, Algorithms for the universal and a priori TSP, Oper. Res. Lett. 36 (1) (2008) 1–3.
[19] David Shmoys, Kunal Talwar, A constant approximation algorithm for the a priori traveling salesman problem, in: International Conference on Integer Programming and Combinatorial Optimization, Springer, 2008, pp. 331–343.
[20] René Sitters, The minimum latency problem is NP-hard for weighted trees, in: International Conference on Integer Programming and Combinatorial Optimization, Springer, 2002, pp. 230–239.
[21] Anke Van Zuylen, Deterministic sampling algorithms for network design, Algorithmica 60 (1) (2011) 110–151.
[22] Bang Ye Wu, Polynomial time algorithms for some minimum latency problems, Inform. Process. Lett. 75 (5) (2000) 225–229.