Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Journal of Discrete Algorithms ••• (••••) •••–•••
JOURNAL OF DISCRETE ALGORITHMS
www.elsevier.com/locate/jda

Approximating the k-traveling repairman problem with repairtimes [☆]

Raja Jothi [a,*,1], Balaji Raghavachari [b]

[a] National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD 20894, USA
[b] Department of Computer Science, University of Texas at Dallas, Richardson, TX 75080, USA

Accepted 30 March 2006

Abstract

Given an undirected graph $G = (V, E)$ and a source vertex $s \in V$, the k-traveling repairman (KTR) problem, also known as the minimum latency problem, asks for k tours, each starting at s and together covering all the vertices (customers) such that the sum of the latencies experienced by the customers is minimum. *Latency* of a customer p is defined to be the distance traveled (time elapsed) before visiting p for the first time. Previous literature on the KTR problem has considered the version of the problem in which the repairtime of a customer is assumed to be zero for latency calculations. We consider a generalization of the problem in which each customer has an associated repairtime. For a fixed k, we present a $(\beta + 2)$-approximation algorithm for this problem, where β is the best achievable approximation ratio for the KTR problem with zero repairtimes (currently $\beta = 6$). For arbitrary k, we obtain a $(\frac{3}{2}\beta + \frac{1}{2})$-approximation ratio. When the repairtimes of all the customers are the same, we present an approximation algorithm with better ratio.[2] We also introduce the *bounded-latency problem*, a complementary version of the KTR problem, in which we are given a latency bound L and are asked to find the minimum number of repairmen required to service all the customers such that the latency of no customer is more than L. For this problem, we present a simple bicriteria approximation algorithm that finds a solution with at most $2/\rho$ times the number of repairmen required by an optimal solution, with the latency of no customer exceeding $(1 + \rho)L$, $\rho > 0$.

Keywords: Approximation algorithms; Combinatorial optimization

---

1. Introduction

Given a finite metric on a set of vertices V and a source vertex $s \in V$, the k-traveling repairman (KTR) problem, a generalization of the metric traveling repairman problem (also known as the *minimum latency problem*, the *delivery man problem* [6,11], and the *school bus-driver problem* [14]), asks for k tours, each starting at s (depot) and together covering all the vertices (customers) such that the sum of the latencies experienced by the customers is minimum.

[☆] Preliminary version of this paper appeared in Proc. LATIN 2004. Research supported in part by the National Science Foundation under grant CCR-9820902.

[*] Corresponding author.

E-mail addresses: jothi@ncbi.nlm.nih.gov (R. Jothi), rbk@utdallas.edu (B. Raghavachari).

[1] This work was done when this author was a graduate student at the University of Texas at Dallas.

[3] The ratio is 7.1604, based on the current best β value, which is 6.

1570-8667/$ – see front matter © 2006 Published by Elsevier B.V.
doi:10.1016/j.jda.2006.03.023

Latency of a customer p is defined to be the distance traveled (time elapsed) before visiting p for the first time. The KTR problem is NP-hard [12], even for $k = 1$. The problem remains NP-hard even for weighted trees [13].

The KTR problem with $k = 1$ is known as the minimum latency problem (MLP) in the literature. The first constant factor approximation for MLP was given by Blum et al. [2]. Goemans and Kleinberg [8] improved the ratio for MLP to 3.59α, where discussion, let α is the best achievable approximation ratio for the i-MST problem. Given an undirected graph with non-negative edge costs and an integer i, the well-known i-MST problem is that of finding a minimum cost spanning tree spanning i nodes. The i-MST problem is NP-hard. The current best approximation ratio for the i-MST problem is 2, due to Garg [7]. Archer, Levin and Williamson [1] presented faster algorithms for MLP with a slightly better approximation ratio of 7.18. Recently, Chaudhuri et al. [3] have reduced the ratio by a factor of 2, to 3.59. They build on Archer, Levin and Williamson's techniques with the key improvement being that they bound the cost of their i-trees by the cost of a minimum cost path visiting i nodes, rather than twice the cost of a minimum cost tree spanning i nodes.

For the KTR problem, Fakcharoenphol, Harrelson, and Rao [5] presented a 8.497α-approximation algorithm. Their ratio was recently improved to $2(2 + \alpha)$ by Chekuri and Kumar [4]. For a multidepot variant of the KTR problem, in which k repairmen start from k different starting locations, Chekuri and Kumar [4] presented a 6α-approximation algorithm. Recently, Chaudhuri et al. [3] have reduced the ratio to 6 for both the KTR problem and its multidepot variant.

1.1. Problem statement

1.1.1. The generalized KTR problem

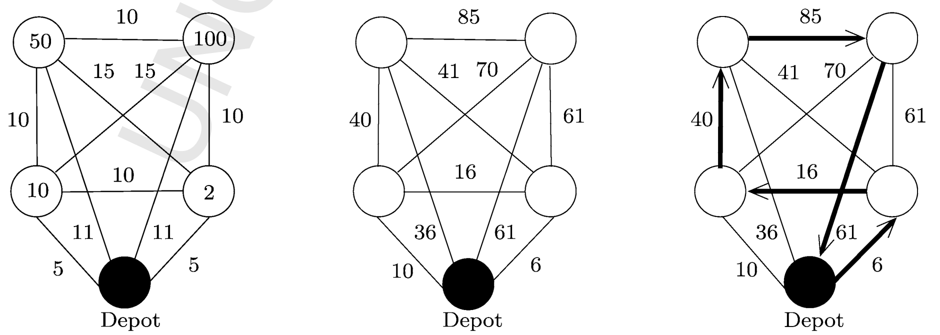
Literature on the KTR problem shows that all the results thus far are based on the assumption that the repairtime of a customer is zero for latency calculations. In this paper, we consider a generalization of the KTR problem (GKTR), the problem definition of which may be formalized as follows.

> **GKTR:** Given a metric defined on a set of vertices, V, a source vertex $s \in V$ and a positive number k. Also given is a non-negative number for each vertex $v \in \{V - s\}$, denoting the repairtime at v. The objective is to find k tours, each starting at s, together covering all the vertices such that the sum of the latencies of all the vertices is minimum.

Note that the definition of "latency" for the GKTR problem is the same as that of the KTR problem. In particular, the latency of a customer p does not include p's repairtime. By definition, latency is the amount of time a customer waits before being served.

It is easy to see that the GKTR problem resembles most real-life situations, one of which is that the repairmen have to spend some time at each customer's location, say, for the repair or installation of equipment. This applies even for a deliveryman who spends some time delivering goods. Hence, it is natural to formulate the repairman problem with repairtimes.

At first, even though it looks like that the GKTR problem can be reduced to the KTR problem in a straight forward manner, taking a deeper look into the problem reveals that such a reduction might not be possible without a compro-



(a) (b) (c)

Fig. 1. (a) Original graph G. (b) Transformed graph G^*. (c) Optimal tour for G^*.

mise in the approximation ratio. An immediate idea would be to make the graph directed, by adding the repairtime of a vertex to the outgoing edges and making the vertex weights to be zero. Unfortunately, a solution to directed latency problem with asymmetric edge weights is not known.

Another idea would be to incorporate the repairtimes associated with vertices into edge weights (where the weight of an edge represents the time to traverse that edge), which can be done by boosting the edge weights as follows: for every edge e incident on vertices i and j in the given graph G, increase the weight (or distance) of e by the sum of $r_i/2$ and $r_j/2$, where r_i and r_j are the repairtimes of i and j respectively. Fig. 1 depicts such a transformation for a sample instance with $k = 1$. The resultant graph G^* after such a transformation will still obey triangle inequality, which allows us to use any of the KTR algorithms, say, with approximation guarantee β. The solution obtained would be a β-approximation for the modified graph G^*. However, the obtained solution will not be a β-approximation for the original problem G. This is due to the reason that the lower bounds for the problems defined as G and G^* are different, as can be seen from the fact that the latency of a customer v in an optimal solution to G^* comprises half of v's repairtime, while this is not the case with an optimal solution to G. At first, even though it looks like an optimal solution to G will be off by just a small amount when compared to an optimal solution to G^*, in reality, it could be arbitrarily large.

Let us consider the following instance to understand this better. Let there be a customer whose repairtime is much larger than the repairtimes of all other customers and the edge weights in the graph. An optimal solution for such an instance will serve this customer last in a repairman's route and therefore its latency will not include the repairtime of this customer. If we use the above described strategy to transform the given graph into a new graph, then the optimal solution for the new graph will be much larger than the optimal solution to the original graph, since the latency of every vertex in such a solution will be more than half its repairtime. The above discussion demonstrates the difficulty involved in the reduction of the GKTR problem to the KTR problem. As a result, we conclude that the algorithms presented for the KTR problem do not solve the GKTR problem with the same approximation guarantee.

In this paper, we present algorithms that surmount these difficulties. For fixed k, we present a $(\beta+2)$-approximation algorithm[3] for the GKTR problem, where β is the best achievable approximation ratio (currently 6) for the KTR problem. For arbitrary k, we obtain a $(\frac{3}{2}\beta + \frac{1}{2})$-approximation ratio. When the repairtimes of all the customers are the same, we present an approximation algorithm with a better ratio. Based on the current best β value, the ratio is 7.1604.

1.1.2. *The bounded-latency problem*

This problem is a complementary version of the KTR problem, in which we are given a latency bound L and are asked to find the minimum number of repairmen required to service all the customers such that the latency of no customer is more than L. More formally, we can define the bounded-latency problem (BLP) as follows:

> **BLP:** Given a metric defined on a set of vertices, V, a source vertex $s \in V$ and a positive number L. The objective is to find a minimum number of tours, each starting at s, together covering all the vertices, such that the latency of no customer is more than L.

The bounded-latency problem is very common in real-life as most service providers work only during the day, generally an 8-hour work day. Under these circumstances, the service provider naturally wants to provide service to all its outstanding customers within the work day, by using the least number of repairmen. Using a simple reduction to the Hamiltonian cycle problem, we show that the BLP is strongly NP-hard, and present a simple bicriteria approximation algorithm that finds a solution with at most $2/\rho$ times the number of repairmen required by an optimal solution, with the latency of no customer exceeding $(1+\rho)L$, $\rho > 0$.

2. The GKTR problem

Our algorithms for the GKTR problem uses the best available approximation algorithm for the KTR algorithm as a black-box. The current best approximation ratio for the KTR problem is 6, due to Chaudhuri et al. [3].

---

[3] In the preliminary version of this paper [10], we claimed a ratio of 3β for arbitrary k. Gubbala and Pursnani [9] improved the analysis to obtain a ratio of $\frac{3}{2}\beta + \frac{3}{2}$.

Throughout this paper, the terms vertex and customer will be used interchangeably. Let s denote the depot or the starting vertex. Let G be the edge-weighted complete graph induced by the vertex set V. Let r_i denote the repairtime of customer i (repairtime of s is zero). Let $l(v)$ denote v's latency. Let $|ab|$ denote the weight of the edge connecting vertices a and b, which is the metric distance between a and b.

2.1. Non-uniform repairtimes

Let $G = (V, E)$ be the given graph for which a solution is sought. Let $M \subset V$ be a set of vertices with k largest repairtimes. Let G' be the graph induced by $V \backslash M$. Construct a new graph G^* from G' such that for every edge e' incident on vertices i and j in G', introduce an edge e^* connecting i and j in G^* with weight $|ij| + \frac{r_i}{2} + \frac{r_j}{2}$. Make the repairtimes of all the vertices in G^* to be zero. It can be easily seen that the edges in G^* obey triangle inequality, and that G^* is an ordinary KTR instance while G an G' are not. Let *opt*, *opt*$'$ and *opt** denote the total latencies of all the customers in an optimum solution for G, G' and G^*, respectively. Let *apx* and *apx*$'$ denote the total latencies of all the customers in our solution for G and G', respectively. Let *APX*$'$ and *APX** denote the respective approximate solutions for G' and G^*. Before we proceed to the algorithm and its analysis, we present the following lemmas.

Lemma 2.1. $opt \geqslant opt'$.

Proof. Construct an approximate solution *APX*$'$ for G' from an optimal solution of G by visiting only the vertices in $V \backslash M$ (using short-cutting). The fact that G' is a subgraph of G and that its edges obey triangle inequality proves that $opt \geqslant apx'$, which in turn proves the lemma. □

Lemma 2.2. *Let $V = \{x_1, \ldots, x_n\}$ be the set of vertices in G. Let r_i denote the repairtime of x_i and let R_k denote the sum of the k largest repairtimes among the repairtimes of all vertices. Then,*

$$opt \geqslant \left[\sum_{i=1}^{n} \bigl(|sx_i| + r_i\bigr)\right] - R_k.$$

Proof. The fact that the latency of every vertex in an optimal solution is at least $|sx_i|$ and that such a solution has to include at least all, but the k largest, repairtimes proves the lemma. □

The following lemma shows that solutions to the GKTR problem G' (which is G without the nodes with the k largest repairtimes) and the KTR problem G^* (which is G' in which the edge weights have been modified to handle repairtimes) are different by a fixed amount, independent of the tour chosen. Therefore an optimal tour in G' is an optimal tour in G^*, and vice versa.

Lemma 2.3. $opt' = opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2}$.

Proof. We prove the lemma by showing that

$$opt' \geqslant opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2} \quad \text{and} \quad opt' \leqslant opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2}.$$

- $opt' \geqslant opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2}$. Let *OPT*$'$ be an optimal solution to G'. Suppose $opt' < opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2}$. Then, we can construct the same set of k tours for G^* as in *OPT*$'$ such that the ith tour in G^* visits the same set of vertices as visited by the ith tour in *OPT*$'$, and in the same order. The sum of the latencies of all the customers in such a solution for G^* will be $opt' + \sum_{i \in V \backslash M} \frac{r_i}{2}$, which contradicts the fact that *opt** is the optimal sum of latencies for G^*.
- $opt' \leqslant opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2}$. Let *OPT** be an optimal solution to G^*. Suppose $opt' > opt^* - \sum_{i \in V \backslash M} \frac{r_i}{2}$. Then, we can construct the same set of k tours for G' as in *OPT** such that the ith tour in G' visits the same set of vertices as visited by the ith tour in *OPT**, and in the same order. The sum of the latencies of all the customers in such a

solution for G' will be $opt^* - \sum_{i\in V\setminus M}\frac{r_i}{2}$, which contradicts the fact that opt' is the optimal sum of latencies for G'. $\square$

We now describe our algorithm for the GKTR problem. We obtain an approximate solution APX' to G' as follows. We first obtain a β-approximate solution APX^* to G^*. Let $t_1, t_2, \ldots, t_k$ be the set of k tours in APX^*. We then construct the same set of k tours for G' as in APX^* such that the ith tour in G' visits the same set of vertices as visited by the ith tour in APX^*, and in the same order. It can be seen that the sum of the latencies of all the customers in G' is

$$\begin{aligned} apx' &\leqslant \beta opt^* - \sum_{j\in V\setminus M}\frac{r_j}{2} \\ &= \beta\left(opt' + \sum_{i\in V\setminus M}\frac{r_i}{2}\right) - \sum_{i\in V\setminus M}\frac{r_i}{2} \quad \text{(by Lemma 2.3)}. \end{aligned} \tag{1}$$

Let $M = \{v_1, v_2, \ldots, v_k\}$ be the set of vertices, with k largest repairtimes in G. We now extend tour t_i in G' to include v_i as its last vertex, for all i, resulting in a feasible set of k tours for G, with apx denoting the sum of the latencies of all the customers in G. The latency of vertex v_i, $l(v_i)$, added to the ith tour will be at most the sum of the latency of its predecessor vertex p_i (vertex visited by the ith tour just before visiting v_i), p_i's repairtime r_{p_i}, and $|p_i v_i|$. Let p_i be the jth vertex visited in the ith tour and let $\{u_{(1,i)}, \ldots, u_{(j-1,i)}\}$ be the other vertices visited by the ith tour before visiting p_i. Since $|sp_i| + |sv_i| \geqslant |p_i v_i|$, where s is the central depot, the latency of v_i is given by

$$l(v_i) \leqslant l(p_i) + r_{p_i} + |sp_i| + |sv_i|.$$

The sum of the latencies of all v_i's, where $i = 1\ldots k$, is given by

$$\begin{aligned} \sum_{i=1}^{k} l(v_i) &\leqslant \sum_{i=1}^{k}\bigl[l(p_i) + r_{p_i} + |sp_i| + |sv_i|\bigr] \\ &\leqslant \sum_{i=1}^{k}\left[|sp_i| + \sum_{g=1}^{j-1} 2|su_{(g,i)}| + \sum_{g=1}^{j-1} r_{u_{(g,i)}}\right] + \sum_{i=1}^{k}\bigl[r_{p_i} + |sp_i| + |sv_i|\bigr]. \end{aligned} \tag{2}$$

2.1.1. $(\frac{3}{2}\beta + \frac{1}{2})$-*approximation analysis for arbitrary* k

We can rewrite (2) as follows:

$$\begin{aligned} \sum_{i=1}^{k} l(v_i) &\leqslant \sum_{i=1}^{k}\left[|sp_i| + \sum_{g=1}^{j-1}|su_{(g,i)}|\right] + \sum_{i=1}^{k}\left[\left(\sum_{g=1}^{j-1} r_{u_{(g,i)}}\right) + r_{p_i} + \left(\sum_{g=1}^{j-1}|su_{(g,i)}|\right) + |sp_i| + |sv_i|\right] \\ &\leqslant \sum_{i=1}^{k}\left[|sp_i| + \sum_{g=1}^{j-1}|su_{(g,i)}|\right] + opt \quad \text{(by Lemma 2.2)}. \end{aligned} \tag{3}$$

The sum of the latencies of all the customers in G is given by

$$apx = apx' + \sum_{i=1}^{k} l(v_i).$$

By substituting (1) and (3), we get

$$\begin{aligned} apx &\leqslant \beta opt' + opt + \frac{\beta - 1}{2}\sum_{i\in V\setminus M} r_i + \sum_{i=1}^{k}\left[|sp_i| + \sum_{g=1}^{j-1}|su_{(g,i)}|\right] \\ &\leqslant (\beta + 1)opt + \frac{\beta - 1}{2}\sum_{i\in V\setminus M} r_i + \sum_{i=1}^{k}\left[|sp_i| + \sum_{g=1}^{j-1}|su_{(g,i)}|\right] \quad \text{(by Lemma 2.1)} \end{aligned}$$

$$
\begin{aligned}
&= (\beta+1)opt + \frac{\beta-3}{2}\sum_{i\in V\setminus M} r_i + \sum_{i=1}^{k}\left[|sp_i| + \sum_{g=1}^{j-1}|su_{(g,i)}|\right] + \sum_{i\in V\setminus M} r_i \\
&\leqslant (\beta+1)opt + \frac{\beta-3}{2}\sum_{i\in V\setminus M} r_i + opt \quad \text{(by Lemma 2.2)} \\
&= (\beta+2)opt + \frac{\beta-3}{2}\sum_{i\in V\setminus M} r_i \\
&\leqslant (\beta+2)opt + \frac{\beta-3}{2}opt \quad \text{(by Lemma 2.2)} \\
&= \left(\frac{3}{2}\beta + \frac{1}{2}\right)opt.
\end{aligned}
$$

Theorem 2.1. *For the GKTR problem, there exists a polynomial time algorithm with* $(\frac{3}{2}\beta + \frac{1}{2})$*-approximation ratio, where* β *is the best achievable approximation ratio for the KTR problem.*

2.1.2. $(\beta+2)$-approximation analysis for fixed k

Let Q be the set of vertices that are visited last in each of the k tours in an optimal solution to $G=(V,E)$. Since k is fixed, we can assume that we know Q, as it takes polynomial time in the order of $|V|^k$ to try out all possible sets for Q. Knowing set Q strengthens Lemma 2.1 as shown in the following lemma. Since $O(|V|^k)$ is polynomial but prohibitive, this algorithm is only of theoretical interest.

Lemma 2.4.

$$opt \geqslant opt' + \sum_{j\in V\setminus Q} r_j + \sum_{i\in Q}|sv_i|.$$

Proof. The lemma follows immediately from the following fact: every vertex $v\in Q$ that is visited last by a tour in an optimal solution to G must account for a latency of at least the sum of the repairtimes of all the vertices that were visited by that tour prior to visiting v, and the cost of the path along that tour. □

For fixed k, we can rewrite (2) as follows:

$$\sum_{i=1}^{k} l(v_i) \leqslant \sum_{i=1}^{k}\left[\sum_{g=1}^{j}2|su_g| + |sv_i| + \sum_{g=1}^{j} r_{u_g}\right] = \sum_{i\in V\setminus Q}2|si| + \sum_{i\in Q}|si| + \sum_{i\in V\setminus Q} r_i. \tag{4}$$

As before, the sum of the latencies of all the customers in G is given by

$$apx = apx' + \sum_{i=1}^{k} l(v_i).$$

By substituting (1) and (4), we get

$$
\begin{aligned}
apx &= \beta\left(opt' + \sum_{i\in V\setminus Q}\frac{r_i}{2}\right) - \sum_{i\in V\setminus Q}\frac{r_i}{2} + \sum_{i\in V\setminus Q}2|si| + \sum_{i\in Q}|si| + \sum_{i\in V\setminus Q} r_i \\
&= \beta opt' + \frac{\beta+1}{2}\sum_{i\in V\setminus Q} r_i + \sum_{i\in Q}|si| + \sum_{i\in V\setminus Q}2|si| \\
&\leqslant \beta opt + \sum_{i\in V\setminus Q}2|si| \quad \text{(by Lemma 2.4)} \\
&\leqslant \beta opt + 2opt \quad \text{(by Lemma 2.2)}.
\end{aligned}
$$

Theorem 2.2. *For the GKTR problem with fixed k, there exists a polynomial time algorithm with* $(\beta + 2)$*-approximation ratio, where* β *is the best achievable approximation ratio for the KTR problem.*

2.2. Uniform repairtimes

In this variant of the problem, it is assumed that the repairtimes at customer locations are all the same. It appears that one could convert the GKTR instance with uniform repairtimes into a KTR instance by incorporating the repairtimes to the edge lengths, but such a transformation will violate the triangle inequality property. We show that the approximation guarantee for the GKTR problem can be improved when the repairtimes are all the same. To achieve a smaller approximation ratio, we present two algorithms that work at tandem. Our first algorithm produces an approximation ratio that decreases with increasing k/n (n is the number of customers), whereas our second algorithm produces a ratio that increases with k/n. As before, let β be the best achievable approximation ratio for the KTR problem.

Let $G = (V, E)$ be the given graph for which a solution is sought. Let r denote the repairtime of the customer i.e., $\forall_{i \neq s} r_i = r$ (repairtime of s is zero). Construct a new graph G^* from G such that for every edge e incident on vertices i and j in G, introduce an edge e^* connecting on i and j in G^* with weight $|ij| + \frac{r_i}{2} + \frac{r_j}{2} = |ij| + r$. Make the repairtimes of all the vertices in G^* to be zero. It can be easily seen that the edges in G^* obey triangle inequality and that G^* is a KTR instance. Let *opt* and *opt** denote the total latencies of all the customers in an optimum solution for G and G^*, respectively. Let *apx* denote the total latency of all the customers in our solution for G. Let n denote the number of vertices in the graph.

Fact 2.1. *Let* C *be a positive constant. Let* x *and* y *be integer variables such that* $x + y = C$*. Then,* $x(x-1) + y(y-1)$ *is minimum when* $x = y$ *(if* $x + y$ *is even) or* $|x - y| = 1$ *(if* $x + y$ *is odd).*

Lemma 2.5. *Let* $opt = opt_t + opt_r$ *be the total latency of an optimal solution, where* opt_t *and* opt_r *are the latency contributions due to travel and repairtime, respectively. Then,*

$$opt_r \geqslant \frac{rn(\frac{n}{k} - 1)}{2}.$$

Proof. If, in an optimal solution OPT, there exists two repairmen who visit different number of customers, say y and z, then, we can construct an alternate solution ALT from OPT by making those two repairmen visit $\frac{y+z}{2}$ customers each if $y + z$ is even or $\lceil \frac{y+z}{2} \rceil$ and $\lfloor \frac{y+z}{2} \rfloor$ customers, respectively, if $y + z$ is odd. If the total latency of ALT is greater than that of OPT, by Fact 2.1, the contribution to the sum of latencies, in ALT, due to repairtimes alone will be less than opt_r. We can continue to find a feasible solution in this manner, until the difference in the number of customers visited by any two repairmen is at most one. That is, each repairman will visit at least $\lceil \frac{n}{k} \rceil$ and at most $\lfloor \frac{n}{k} \rfloor$ customers. That brings us to the following equation, which proves the lemma.

$$opt_r \geqslant rk \frac{\frac{n}{k}(\frac{n}{k} - 1)}{2} = \frac{rn(\frac{n}{k} - 1)}{2}. \qquad \square$$

2.2.1. Algorithm 1

This algorithm proceeds on a case-by-case basis, based on the value of k with respect to n. Let the customers be sorted in non-decreasing order with respect to their distances to the depot. Let $A = \{c_1, \ldots, c_n\}$ be the sorted set of n customers, i.e., $|sc_1| \leqslant |sc_2| \leqslant \cdots \leqslant |sc_n|$.

Case 1. $k \geqslant \frac{n}{2}$. The ith repairman visits customer c_i first, $\forall i \leqslant k$. In addition, repairmen 1 to $n - k$ are assigned to visit one additional customer each, from the remaining pool of $n - k$ unassigned customers, as their second customer.

Let t_1 be one of k such tours constructed in this manner. Let c_1, and maybe c_I, be the customers visited by tour t_1, in that order. The latency of c_1 would just be $|sc_1|$ and the latency of c_I would be at most $|sc_1| + r + |c_1c_I| \leqslant |sc_1| + r + |sc_1| + |sc_I|$ (refer Fig. 2(a)). The sum of the latencies of customers c_1 and c_I is at most $|sc_1| + |sc_I| + r + 2|sc_1|$. The sum of the latencies of all the customers visited by k tours is then at most

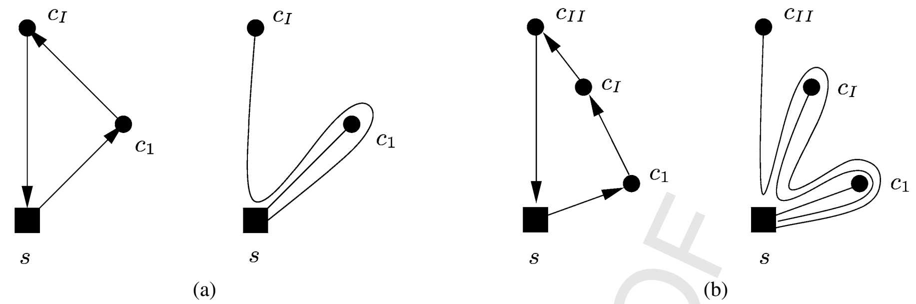


Fig. 2. (a) $k \leqslant \frac{n}{2}$; (b) $\frac{n}{3} \leqslant k < \frac{n}{2}$.

$\sum_{j=1}^{n} |sc_j| + (n-k)r + 2\sum_{j=1}^{n-k} |sc_j|$. By Lemma 2.2, the sum of the latencies is at most $opt + 2\sum_{j=1}^{n-k} |sc_j|$. Since A is sorted in non-decreasing order, the approximation ratio is less than or equal to $1 + 2(\frac{n-k}{n}) = 1 + 2(1-x)$, where $x = \frac{k}{n}$. Since $k \geqslant \frac{n}{2}$, the ratio is at most 2.

Case 2. $\frac{n}{3} \leqslant k < \frac{n}{2}$. The ith repairman visits customer c_i first, $\forall i \leqslant k$. Each repairman picks one customer out of $\{c_{k+1}, \ldots, c_{2k}\}$ to be his next customer. In addition, repairmen 1 to $(n-2k)$ are assigned to visit one additional customer each, from the remaining pool of $n-2k$ unassigned customers, as their third customer.

Let t_1 be one of k such tours constructed in this manner. Let c_1, c_I, and maybe c_{II}, be the customers visited by tour t_1, in that order. The latency of c_1 would just be $|sc_1|$. The latencies of c_I and c_{II} would be at most $|sc_1| + r + |sc_1| + |sc_I|$ and $|sc_1| + r + |sc_1| + |sc_I| + r + |sc_I| + |sc_{II}|$, respectively (see Fig. 2(b)). The sum of the latencies of customers c_1, c_I and c_{II} is at most $|sc_1| + |sc_I| + |sc_{II}| + 4|sc_1| + 2|sc_I| + r + 2r$. The sum of the latencies of all the customers visited by k tours is given by

$$
\begin{aligned}
apx &\leqslant \sum_{j=1}^{n} |sc_j| + 2\sum_{j=1}^{k} |sc_j| + 2\sum_{j=1}^{n-2k} |sc_j| + 2\sum_{j=k+1}^{n-k} |sc_j| + kr + (n-2k)2r \\
&\leqslant \sum_{j=1}^{n} |sc_j| + \frac{2k}{n}\sum_{j=1}^{n} |sc_j| + \frac{2(n-2k)}{n}\sum_{j=1}^{n} |sc_j| + \frac{2(n-2k)}{n-k}\sum_{j=1}^{n} |sc_j| + 2(n-k)r - kr \\
&\leqslant \sum_{j=1}^{n} |sc_j| + \frac{2(n-k)}{n}\sum_{j=1}^{n} |sc_j| + \frac{2(n-2k)}{n-k}\sum_{j=1}^{n} |sc_j| + 2(n-k)r \\
&= 3\sum_{j=1}^{n} |sc_j| + 2(n-k)r - \frac{2k}{n}\sum_{j=1}^{n} |sc_j| + \frac{2(n-2k)}{n-k}\sum_{j=1}^{n} |sc_j| \\
&\leqslant 3opt - \frac{2k}{n}\sum_{j=1}^{n} |sc_j| + \frac{2(n-2k)}{n-k}\sum_{j=1}^{n} |sc_j| \quad \text{(by Lemma 2.2)} \\
&\leqslant 3opt - \frac{2k}{n}opt + \frac{2(n-2k)}{n-k}opt \quad \text{(by Lemma 2.2)} \\
&= \left[3 - \frac{2k}{n} + \frac{2(n-2k)}{n-k}\right]opt \\
&= \left[1 + 2(1-x) + \frac{2(1-2x)}{1-x}\right]opt
\end{aligned}
$$

where $x = \frac{k}{n}$. Since $\frac{n}{3} \leqslant k < \frac{n}{2}$, apx is bounded by $\frac{10}{3}opt$.

Case l (*in general*). $\frac{n}{l+1} \leqslant k < \frac{n}{l}$. The ith repairman visits customer c_i first, $\forall i \leqslant k$. Each repairman picks one customer out of $\{c_{k+1}, \ldots, c_{2k}\}$ to be his second customer, one customer out of $\{c_{2k+1}, \ldots, c_{3k}\}$ to be his third customer, $\ldots$, and one customer out of $\{c_{(l-1)k+1}, \ldots, c_{lk}\}$ to be his lth customer. In addition, repairmen 1 to $(n-lk)$ are assigned to visit one additional customer each, from the remaining pool of $n-lk$ unassigned customers, as their $(l+1)$th customer.

Let t_1 be one of k such tours constructed in this manner. Let $c_1, c_I, c_{II}, \ldots$ be the customers visited by tour t_1, in that order. Latencies of $c_1, c_I, c_{II}, \ldots$ are calculated in the same manner as done in case 2. The analysis proceeds in the same manner as in case 2 and the sum of the latencies of all the customers visited by k tours is given by

$$apx \leqslant \left[1 + \sum_{j=0}^{l-1} \frac{2(1-(j+1)x)}{1-jx}\right] opt, \tag{5}$$

where $x = \frac{k}{n}$. Since $\frac{n}{l+1} < k \leqslant \frac{n}{l}$, for values of $l = 3, 4, 5, 6, 7, 8, 9, \ldots$, apx is bounded by $\frac{29}{6}opt$, $\frac{193}{30}opt$, $\frac{193}{30}opt$, $\frac{81}{10}opt$, $\frac{687}{70}opt$, $\frac{1619}{140}opt$, $\frac{16811}{1260}opt, \ldots$, respectively.

2.2.2. Algorithm 2

Just like in the non-uniform repairtime case, we find a β-approximate solution APX^* to G^*. Let $t_1, t_2, \ldots, t_k$ be the set of k tours in APX^*. Construct the same set of k tours in G as in APX^* such that the ith tour in G visits the same set of vertices as visited by the ith tour in APX^*, and in the same order. It can be seen that the sum of the latencies of all the customers in G is

$$apx = \beta opt^* - \sum_{i=1}^{n} \frac{r_i}{2} = \beta opt^* - \frac{nr}{2}. \tag{6}$$

By Lemma 2.3,

$$opt = opt^* - \sum_{i=1}^{n} \frac{r_i}{2} = opt^* - \frac{nr}{2}.$$

Substituting for opt^* in (6), we get

$$apx = \beta opt + \left(\frac{\beta - 1}{2}\right) nr.$$

The approximation ratio of Algorithm 2 can be calculated from the above equation as follows.

$$\begin{aligned} \frac{apx}{opt} &= \frac{\beta opt + (\frac{\beta-1}{2})nr}{opt} \\ &\leqslant \beta + \frac{(\frac{\beta-1}{2})nr}{\frac{rn(\frac{n}{k}-1)}{2}} \quad \text{(by Lemma 2.5)} \\ &\leqslant \beta + \left(\frac{\beta - 1}{\frac{n}{k} - 1}\right). \end{aligned} \tag{7}$$

For $\beta = 6$, plots for k versus Eqs. (5) and (7) reveal that the resulting curves intersect at $k = 0.188364n$ yielding the 7.1604 approximation ratio. In other words, for $\beta = 6$, Algorithm 1 guarantees an approximation ratio of 7.1604 for values of $k \geqslant 0.188364n$, and Algorithm 2 guarantees the same ratio for values of $k \leqslant 0.188364n$, leading us to the following theorem. A tight example for our analysis would be when vertices visited by a tour are on a line with the depot placed strategically in the middle. This would result in a tour criss-crossing the depot every time a new vertex is visited.

Theorem 2.3. *For the GKTR problem with uniform repairtimes, there exists a polynomial time algorithm with* 7.1604-*approximation ratio.*

Our algorithm scales nicely, and any future improvement of β will result in a better ratio for the uniform GKTR. For example, for $\beta = 5, 4, 3$, our algorithm would guarantee approximation ratios of 6.1, 5.0, 3.9, respectively.

3. The bounded-latency problem

The bounded-latency problem (BLP) is a complementary version of the KTR problem, in which we are given a latency bound L and are asked to find the minimum number of repairmen required to service all the customers such that the latency of no customer is more than L. This is unlike the GKTR problem, in which the sum of the latencies is minimized. It can easily be shown that the BLP is strongly NP-hard through a simple reduction to the Hamiltonian problem.

Theorem 3.1. *The BLP is strongly NP-hard.*

Proof. We prove the theorem by showing that a special case (with zero repairtimes) of the BLP is strongly NP-hard, even when there are only two different edge weights Let K_n be the complete graph induced by the vertices of the given graph $G = (V, E)$, with $n = |V|$. Each edge e_{ij} connecting vertices i and j in K_n is assigned a weight of 1 if there exists an edge between i and j in G, and 2 otherwise. The edges in K_n satisfy triangle inequality. Pick an arbitrary vertex to be the depot s, and set the repairtimes at all vertices to be zero. Set latency bound $L = n - 1$. Now, G is Hamiltonian if and only if a single repairman is sufficient to serve all the vertices in the weighted graph constructed, while satisfying the latency bound. □

Non-metric BLP is as non-approximable as the non-metric *traveling salesman problem* as edges not in E can be assigned an arbitrarily large weight instead of 2 (in the proof above).

For the BLP problem with zero repairtimes, we present a simple bicriteria approximation algorithm that finds a solution with at most $2/\rho$ times the number of repairmen required by an optimal solution, with the latency of no customer exceeding $(1+\rho)L$, $\rho > 0$.

For a BLP instance with latency bound L, let η be the minimum number of repairmen required to service all the customers with the latency of no customer exceeding L. Let ℓ denote the length of the tree obtained from a feasible BLP solution by removing the edges connecting the last customers in each of the tour to the depot.

Fact 3.1. *For any BLP instance I, ℓ is at least the length of an MST spanning the vertices in I.*

Fact 3.2. $\eta \geqslant \ell/L$.

Given below is an algorithm which groups the customers, so that a repairman can be assigned to each of the groups. Let $\rho > 0$.

```
1. Construct a tour for the given set of vertices (depot and customers) using the best available approximation algorithm for TSP.
2. Remove the depot from the tour.
3. Set lengthTraveled = 0;
4. Starting from some vertex, traverse the tour.
5. While not all edges in the tour are traversed, traverse the next edge $e$ on the tour.
   (a) If lengthTraveled + length($e$) $\leqslant \rho L$, set lengthTraveled = lengthTraveled + length($e$).
   (b) Else remove $e$ from the tour, and set lengthTraveled = 0.
```

At the end of the above algorithm, we will be left with segments, each of length at most ρL, as shown in Fig. 3. For each segment, introduce two edges to connect its endpoints (vertices) to the depot. Since our tour is of length at most twice than that of an MST, by Fact 3.1, our solution will require at most $2\ell/\rho L \leqslant 2\eta/\rho$ repairmen (by Fact 3.2). Assuming that there exists a feasible solution for a given instance, the length of an edge connecting any vertex to the depot is at most L. Hence, regardless of which direction each tour in our solution is traversed, each customer will have a latency of at most $(1+\rho)L$.

Theorem 3.2. *For the bounded-latency problem, in which we are a given a latency bound L, there exists a bicriteria approximation algorithm that finds a solution with at most $2/\rho$ times the number of repairmen required by an optimal solution, with the latency of no customer exceeding $(1+\rho)L$, $\rho > 0$.*

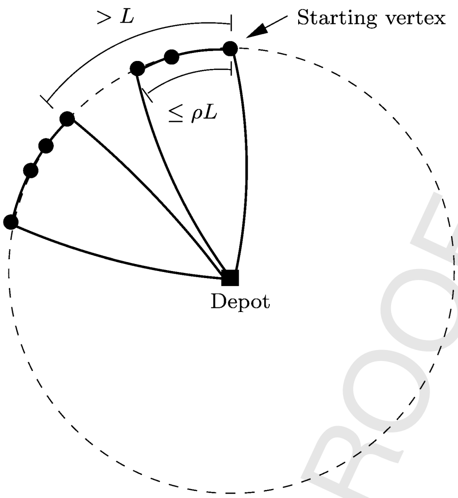


Fig. 3. Cutting the tour into segments of size at most ρL.

4. Conclusion

We presented approximation algorithms for the generalized version of the KTR problem, in which the time spent by a repairman at a customer's location is considered to be non-zero. For the case when the repairtimes are different for each customer, our algorithm guarantees a ratio of $\beta + 2$ for fixed k, $\frac{3}{2}\beta + \frac{1}{2}$ for arbitrary k. Here β is the best achievable approximation ratio for the original KTR problem with zero repairtimes. For the case when the repairtimes are all the same, we presented a 7.1604-approximation algorithm. We also introduced a complementary version of the KTR problem, the BLP, in which we are given a latency bound L and are asked to find minimum number of tours such that no customer experiences a latency more than L. For this problem, we presented a bicriteria approximation algorithm that finds a solution with at most $2/\rho$ times the number of repairmen required by an optimal solution, with the latency of no customer exceeding $(1+\rho)L$, $\rho > 0$.

It should be interesting to see whether one can come up with an algorithm that does not use the KTR algorithm as a black-box.

References

[1] A. Archer, A. Levin, D.P. Williamson, Faster approximation algorithms for the minimum latency problem, in: Proc. 14th Symposium on Discrete Algorithms (SODA), 2003, pp. 88–96.
[2] A. Blum, P. Chalasani, D. Coppersmith, B. Pulleyblank, P. Raghavan, M. Sudan, The minimum latency problem, in: Proc. 26th Symposium on Theory of Computing (STOC), 1994, pp. 163–171.
[3] K. Chaudhuri, B. Godfrey, S. Rao, K. Talwar, Paths, tours, and minimum latency tours, in: Proc. 44th IEEE Symposium on Foundations of Computer Science (FOCS), 2003, pp. 36–45.
[4] C. Chekuri, A. Kumar, Maximum coverage problem with group budget constraints, in: Proc. 7th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX), 2004, pp. 72–83.
[5] J. Fakcharoenphol, C. Harrelson, S. Rao, The k-traveling repairman problem, in: Proc. 14th Symposium on Discrete Algorithms (SODA), 2003, pp. 655–664.
[6] M. Fischetti, G. Laporte, S. Martello, The delivery man problem and cumulative matroids, Operations Research 41 (1993) 1064–1065.
[7] N. Garg, Saving an epsilon: A 2-approximation for the k-MST problem in graphs, in: Proc. 37th Symposium on Theory of Computing (STOC), 2005, pp. 396–402.
[8] M. Goemans, J. Kleinberg, An improved approximation ratio for the minimum latency problem, in: Proc. 7th Symposium on Discrete Algorithms (SODA), 1996, pp. 152–158.
[9] P. Gubbala, H. Pursnani, Personal communication, November 2003.
[10] R. Jothi, B. Raghavachari, Minimum latency tours and the k-traveling repairmen problem, in: Proc. Latin American Theoretical INformatics (LATIN), 2004, pp. 423–433.
[11] E. Minieka, The delivery man problem on a tree network, Annals of Operations Research 18 (1–4) (1989) 261–266.
[12] S. Sahni, T. Gonzales, P-complete approximation problems, JACM 23 (3) (1976) 555–565.
[13] R. Sitters, The minimum latency problem is NP-hard for weighted trees, in: Proc. 9th Conference on Integer Programming and Combinatorial Optimization (IPCO), 2002, pp. 230–239.
[14] T. Will, Extremal results and algorithms for degree sequences of graphs, Ph.D. Thesis, Univ. of Illinois at Urbana-Champaign, 1993.