

The 1-Center and 1-Highway problem

J. M. Díaz-Báñez¹, M. Korman², P. Pérez-Lantero¹, I. Ventura¹

¹ Departamento de Matemática Aplicada II, Universidad de Sevilla, Spain. Partially supported by project MEC MTM2009-08652.
`{dbanez, pplantero, iventura}@us.es`

² Université libre de Bruxelles (ULB).
`mkormanc@ulb.ac.be`

Abstract. We study a variation of the 1-center problem in which, in addition to a single supply facility, we are allowed to locate a highway. This highway increases the transportation speed between any demand point and the facility. We show that we can find the optimal location of both the facility and the highway in $O(n^2)$ or $O(n \log n)$ time, depending on whether or not the highway's length is fixed.

1 Introduction

Geometric optimization related to urban transportation systems is an important topic in computational geometry. Although the metric given by a real urban transportation system is often quite complicated, simplified mathematical models have been widely studied in order to investigate basic geometric properties of urban transportation systems. Abellanas *et. al.* [1] considered a geometric modeling of this environment: represent highways as polygonal chains consisting of line segments in the plane, giving each line segment an associated speed. Then, the travel time between two points gives a metric called the *time distance*.

Recently, there has been an interest in problems derived from urban modeling. In many cases we are interested in locating a highway that optimizes some given function that depends on the distance between elements of a given pointset. Espejo and Chía [4] introduced a variant of the problem in which we are given a set of clients (represented by a set of points S) located in a city. One is interested in locating a service facility and a highway simultaneously in such a way that the average supply time between the clients and the supply point is minimized. In this paper we study a variation of this problem in which we want to minimize the largest travel time between the clients and the facility. The case in which the objective function to minimize is the average travel time is done in a companion paper [6].

2 Definitions and notation

Let S be the set of n client points, f be the service facility point, h be the highway, ℓ be the length of h , t and t' be the endpoints of h , and $v > 1$ be the speed. We assume that the highway to locate can have any orientation. Given a point u of the plane, let $x(u)$ and $y(u)$ denote respectively the x and y coordinates of u .

Fixed the location of the facility and the endpoints of the highway, the distance from a demand point $p \in S$ to f is defined as $d_h(p, f) = \min\{\|p - f\|_1, \|p - t\|_1 + \frac{\ell}{v} + \|t' - f\|_1, \|p - t'\|_1 + \frac{\ell}{v} + \|t - f\|_1\}$. That is, the minimum time between either walking, using the highway in one direction or using it in the reverse direction, see Figure 1. Whenever

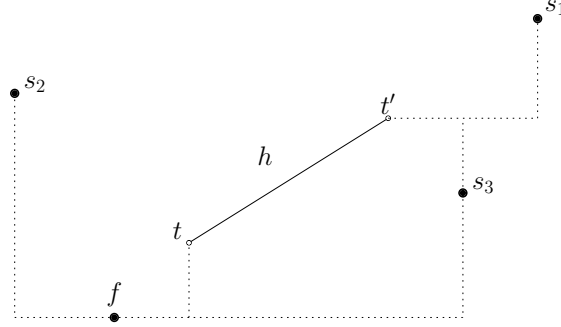


FIGURE 1. The distance model; in the example s_1 uses the highway from t' to t in order to reach f faster. The highway does not speed up transportation between s_2 and f , hence is not used by s_2 . Demand point s_3 however, can either walk or use the highway to reach f , and will need the same time in both cases. Observe that, since we are interested in paths that reach f , the highway will only be used in one direction.

40 $d_h(p, f) < \|f - p\|_1$, we say that p uses the highway to reach f . Otherwise, we say that
 41 p walks (or does not use h) to reach the facility.

42 Given S and $v > 1$, we are interested in locating the facility point f and the highway
 43 h that minimize the expression $\max_{p \in S} d_h(p, f)$, where d_h is the time distance between p
 44 and f . This problem is called the *1-center 1-highway location problem* (1C1H for short).
 45 In this paper we consider two variations of the problem: in the first problem, we consider
 46 the case in which the highway's length ℓ is fixed. In the second variation we are given
 47 more freedom and can locate a highway of any length. We call the former problem the
 48 *fixed length* (or FL-1C1H for short) problem. The latter one will be called the *variable*
 49 *length* (or VL-1C1H) problem.

50 It is easy to see that in the 1C1H problem, the highway will only be used in one
 51 direction. Moreover, there always exists an optimal location in which one of the rapid
 52 transit line endpoints coincides with the facility (this result was also observed in [4],
 53 Lemma 2.1). Therefore, we assume throughout the paper that $f = t'$ thus the distance
 54 from a demand point $p \in S$ to f is now $d_h(p, f) = \min\{\|p - f\|_1, \|p - t\|_1 + \frac{\ell}{v}\}$.

55 3 Solving the 1C1H Problem

56 In this section we will give a general algorithm for solving the 1C1H problem. Unless
 57 otherwise stated, all results of this section hold for both variants of the problem. Due to
 58 space constraints, some of the proofs of this paper have been moved to the Appendix.
 59 Using the standard transformation from L_1 to L_∞ , we solve the problem using L_∞
 60 instead. Let f^* and h^* be an optimal solution of a given problem instance and let
 61 $R^* = \max_{p \in S} d_{h^*}(p, f^*)$.

62 Let $B(u, r)$ denote the axis-parallel square of radius r centered at u , and consider
 63 the balls $B(f^*, R^*)$ and $B(t^*, R^* - \ell/v)$. By definition of R^* , all points of S must be
 64 included in the union of both balls. We partition the pointset S into two sets W^* and
 65 H^* as follows: set W^* contains the points whose L_∞ distance to f is at most R^* . The
 66 set $H^* = S \setminus W^*$ contains the points that must use the highway to reach f in R^* or less
 67 units of time.

68 Observe that we cannot have $W^* = \emptyset$, since by reversing the positions of f^* and t^*
 69 we would obtain a better solution. By definition, the set H^* is empty all point of S can
 70 walk to f^* in R^* or less units of time. This situation can be easily detected, because f^*
 71 is also the solution of the rectilinear 1-center problem. Since the solution to this problem
 72 can be found in linear time, from now on we assume that neither W^* nor H^* is empty.

73 We consider the next problem called the *basic problem*: Given a partition $\{W, H\}$ of
 74 S , find the smallest value R (called the *diameter* of the partition) and the coordinates
 75 of f and t such that $W \subseteq B(f, R)$ and $H \subseteq B(t, R - \ell/v)$. When we consid the fixed-
 76 length variation of the problem, we also add the constraint that f and t must satisfy
 77 $\|f - t\|_2 = \ell$. Since f^* and t^* are optimal, it is easy to see that they are the solution
 78 of the basic problem for the partition $\{W^*, R^*\}$. Moreover, the diameter of any other
 79 partition of S will have equal or higher diameter than R^* .

80 Our algorithm works as follows: we consider different partitions of S and solve the
 81 basic problem associated to each partition. We identify $\{W^*, R^*\}$ as the partition whose
 82 diameter is the smallest. A naive method would be to guess the partition $\{W^*, R^*\}$
 83 among the 2^n candidates. In the following we reduce the search space to one of polynomial
 84 size:

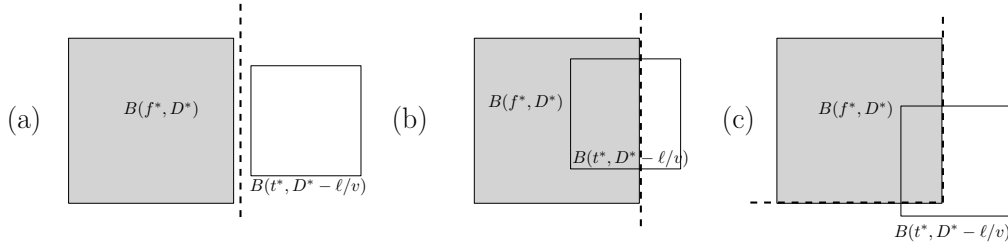


FIGURE 2. Relative positions of the balls $B(f^*, R^*)$ and $B(t^*, R^* - \ell/v)$. For each of the cases the sets W^* (marked in grey) can be split from H^* with either an axis-aligned line or an upper-left quadrant.

85 **Lemma 3.1** *For any set S , the partition $\{W^*, R^*\}$ can be found among $O(n^2)$ candidates*

86 *Proof.* Without loss of generality we can assume that f^* is above and to the left of t^* .
 87 It is then easy to see that there are three possible relative positions of the two balls (see
 88 Figure 2). In each of these cases the two sets can be split by either an axis-aligned line
 89 or an upper left quadrant (see dashed lines in Figure 2). Observe that both cases can
 90 be characterized by the number of points of S to the right and above it, hence $\{W^*, R^*\}$
 91 must be among at most $O(n^2)$ candidates. \square

92 Given a set T of points, let $X(T) \subseteq T$ be the set containing the points with highest
 93 and lowest x and y coordinate of T (this set is called the set of *extreme points* of T).
 94 We define $\delta(T)$ as the largest L_∞ distance between any two points of T . Observe that
 95 $|X(T)| \leq 4$ and that $\delta(T) = \delta(X(T))$. For any real number $r > 0$, let $Center(T, r)$
 96 the locus of the centers of the axis-parallel squares of radius r that cover S . A similar
 97 definition can be seen in [5].

98 **Lemma 3.2** *Let $\{W, H\}$ be a partition of S . If we are given the extreme sets $X(W)$ and*
 99 *$X(H)$, then the basic problem can be solved in constant time (for both fixed-length and*
 100 *variable-length cases).*

By combining the above results we obtain a method to solve both problems:

Theorem 3.3 *Both variants of the 1C1H problem can be solved in $O(n^2)$ time and $O(n)$ space.*

4 Locating a highway of variable length

Using Theorem 3.3 we have an algorithm that runs in $O(n^2)$ time for both the fixed-length and variable-length variants of the problem. The bottleneck of the algorithm is case (c) of Lemma 3.1. In the following we show how to treat this case more efficiently.

Lemma 4.1 *If in every optimal solution of the VL-1C1H problem each ball contains a corner of the other one, there exists an optimal solution (f^*, t^*) of diameter R^* of the VL-1C1H problem in which the extreme points $X(S)$ are in the boundary of $B(f^*, R^*) \cup B(t^*, R^* - \ell/v)$.*

With this observation we can speed up the algorithm for the variable length variant of the problem:

Theorem 4.2 *The VL-1C1H problem can be solved in $O(n \log n)$ time.*

5 Concluding Remarks

In this paper we have considered a variant of the facility location problem introduced in [4] in which we are interested in locating a supply point and a highway simultaneously. In our model we only allow entering and leaving the highway at its endpoints (in other literature this kind of highway is called *walkway* [3] or *turnpike* [2]). We note that there exists another model of highway (called *freeway* [2] or simply *highway* [1]) in which one is allowed to enter and leave at any point. A natural extension of the problems studied in this paper is considering the location of a freeway instead.

Observe that the faster algorithm for the variable length variant of the problem is based on Lemma 4.1. Unfortunately we have examples in which this result does not hold whenever the highway's length is fixed. Thus it remains open to show whether or not the fixed-length problem can also be solved in $O(n \log n)$ time (or showing that it is 3-SUM hard).

References

- [1] M. Abellanas, F. Hurtado, C. Icking, R. Klein, E. Langetepe, L. Ma, B. Palop, and V. Sacristán. Voronoi diagram for services neighboring a highway. *Information Processing Letters*, 86:283–288, 2003.
- [2] S. W. Bae, M. Korman, and T. Tokuyama. All farthest neighbors in the presence of highways and obstacles. In *Proc. of the 3rd International Workshop on Algorithms and Computation, WALCOM'09*, pages 71–82, Berlin, Heidelberg, 2009. Springer-Verlag.
- [3] J. Cardinal, S. Collette, F. Hurtado, S. Langerman, and B. Palop. Optimal location of transportation devices. *Comput. Geom. Theory Appl.*, 41:219–229, 2008.
- [4] I. Espejo and A. M. Rodríguez-Chía. Simultaneous location of a service facility and a rapid transit line. *Comput. Oper. Res.*, 38:525–538, 2011.
- [5] P-H. Huang, Y. T. Tsai, and C. Y. Tang. A fast algorithm for the alpha-connected two-center decision problem. *Inf. Process. Lett.*, 85:205–210, February 2003.
- [6] J. M. Díaz-Báñez, M. Korman², P. Pérez-Lantero¹, and I. Ventura¹. Locating a service facility and a rapid transit line. In *Submitted to the 14th Spanish Meeting on Computational Geometry*, 2011.

Appendix: proofs omitted from the body

Proof of Lemma 3.2

Proof. We start by giving the algorithm for the case in which the distance between f and t is fixed. Observe that we will always have $R \geq \max\{\delta(W), \delta(H) + \ell/v\}$ (otherwise an extreme point of either W or H will not be able to reach either f in R units of time). If there exist two points $f \in \text{Center}(W, \max\{\delta(W), \delta(H) + \ell/v\})$ and $t \in \text{Center}(H, \max\{\delta(W), \delta(H) + \ell/v\})$ such that $\|f - t\|_2 = \ell$ we are done.

Unfortunately, this does not always happen. In general, we must find two values $\varepsilon_1, \varepsilon_2 \geq 0$ such that: (i) $\delta(W) + \varepsilon_1 = \delta(H) + \varepsilon_2 + \frac{\ell}{v}$, (ii) there are points $f \in \text{Center}(W, \delta(W) + \varepsilon_1)$ and $t \in \text{Center}(H, \delta(H) + \varepsilon_2)$ satisfying $\|f - t\|_2 = \ell$, and (iii) $\delta(W) + \varepsilon_1 = \delta(H) + \varepsilon_2 + \frac{\ell}{v}$ is minimized. The values ε_1 and ε_2 can be found in constant time as follows:

First, set $\varepsilon_1 = \max\{0, \delta(H) + \frac{\ell}{v} - \delta(W)\}$ and $\varepsilon_2 = \max\{0, \delta(W) - \delta(H) - \frac{\ell}{v}\}$. That is, we either increase the radius of W to $\delta(H) + \ell/v$ or increase the radius of H to $\delta(W)$. By increasing the smallest radius we are ensuring that condition (i) is satisfied. We now look for the smallest value of $x > 0$ such that there are points $f \in \text{Center}(W, \delta(W) + \varepsilon_1 + x)$ and $t \in \text{Center}(H, \delta(H) + \varepsilon_2 + x)$ satisfying $\|f - t\|_2 = \ell$. Observe that this problem is of constant size and can be computed in $O(1)$ time.

The variable-length variant of the problem is slightly simpler. For this case it suffices to find values $\varepsilon_1, \varepsilon_2 \geq 0$ such that condition (ii) is satisfied. The main difference is that ε_1 and ε_2 must now minimize the expression $\max\{\delta(W) + \varepsilon_1, \delta(H) + \varepsilon_2 + g(\varepsilon_1, \varepsilon_2)/v\}$, where $g(\varepsilon_1, \varepsilon_2)$ denotes the smallest Euclidean distance between points $f \in \text{Center}(W, \delta(W) + \varepsilon_1)$ and a point $t \in \text{Center}(H, \delta(H) + \varepsilon_2)$. As before, this problem has constant size and thus can be solved in $O(1)$ time. \square

Proof of Theorem 3.3

Proof. Recall that, by Lemma 3.1, we can split sets H^* and W^* by either a vertical line or an upper-left quadrant. We start by considering first the case in which there is a vertical splitting line. Sort the points of S in increasing value of x coordinates; let p_1, p_2, \dots, p_n be the obtained order. For any $1 \leq i < n$, let L_i and R_i be the smallest bounding axis-aligned rectangle containing points $\{p_1, \dots, p_i\}$ and $\{p_{i+1}, \dots, p_n\}$, respectively. By scanning from left to right, we can compute and store the extreme points of L_i for all $1 \leq i < n$ in $O(n)$ time. Analogously we sweep from right to left and compute $X(R_i)$ in linear time as well. Then we solve the basic problem for each pair (L_i, R_i) for all values $1 \leq i < n$ using Lemma 3.2. The computationally speaking most expensive part of the algorithm is computing the initial sorting of the points of S , which needs $O(n \log n)$ time.

To complete the proof it remains to show how case (c) can be solved in $O(n^2)$ time and $O(n)$ space. Let q_1, q_2, \dots, q_n be the points of S sorted in decreasing order of y coordinates. For any $1 \leq i, j \leq n$, let $UR_{i,j}$ and $DL_{i,j}$ be the smallest bounding rectangles of the sets $UR_{i,j} := \{u \in S \mid y(u) > y(q_i) \wedge x(u) < x(p_j)\}$ and $S \setminus UR_{i,j}$, respectively. For any fixed $1 \leq i \leq n$ we can proceed as in the previous case. That is, sweep twice the pointset (downwards and upwards) and compute the set of extreme points $X(UR_{i,j})$ for all $1 \leq j \leq n$. Once the bounding rectangles are known, we can solve the $O(n)$ basic problem instances in constant time each. We repeat this algorithm for all values of i and find the optimal partition in quadratic time. Observe that this

method never uses more than $O(n)$ memory, since once a column has been scanned we need only store the best partition found. \square

Proof of Lemma 4.1

Proof. Given S , let p_N and p_S be the points with highest and lowest y -coordinate, respectively. Analogously, p_E and p_W are defined with respect to the x -coordinates. Observe that if any of the balls contains both p_N and p_S (or p_W and p_E), the sets H^* and W^* can be split by either a horizontal or vertical line. Since we assumed that this is not possible, the ball that contains p_N cannot contain p_S (analogously, the ball that contains p_W cannot contain p_E). Without loss of generality, we assume that $p_N, p_W \in B(f^*, R^*)$ and $p_S, p_E \in B(t^*, R^* - \ell/v)$. Observe that this implies that $x(f^*) \leq x(t^*)$, and $y(f^*) \geq y(t^*)$ (that is, the highway and the abscissa form an angle between 0 and $-\pi/2$).

Assume that point p_N is not on the boundary of $B(f^*, R^*)$. In the following we will show how to do a local perturbation to the solution so that we obtain p_N on the boundary. We translate f^* downwards continuously while keeping t^* unchanged until p_N reaches the top boundary. Observe that, since initially f^* has higher coordinate than t^* , the translation will reduce the distance between f^* and t^* until both points share the same y coordinate. However, observe that this cannot happen, since otherwise we can split W^* and H^* with a vertical line. Moreover, no point of $B(f^*, R^*)$ can leave the ball before p_N reaches the top boundary, hence optimality is preserved through this translation operation. Analogously we can do the same operation on the other extreme points and obtain that either all extreme points are in the boundary of $B(f^*, R^*) \cup B(t^*, \delta(R^* - \ell/v))$ or find a way to split both f^* and t^* with a vertical or horizontal line. \square

Proof of Theorem 4.2

Proof. By Lemma 4.1 either there exists an axis-parallel line that splits the sets W^* and H^* or all extreme points are in the boundary of $B(f^*, R^*) \cup B(t^*, R^* - \ell/v)$. The first case can be treated in $O(n \log n)$ time using the same approach as in Theorem 3.3, hence we focus on the latter case.

Without loss of generality, we assume that $x(f^*) < x(t^*)$, and $y(f^*) \geq y(t^*)$ (that is, the angle that the highway forms with the abscissa ordinate is between 0 and $-\pi/2$). In particular, this implies that $p_N, p_W \in B(f^*, R^*)$ and $p_E, p_S \in B(t^*, R^* - \ell/v)$. Denote by $u = (x(p_W), y(p_N))$ the top-left corner of the smallest enclosing axis-aligned rectangle of S . By Lemma 4.1 this point must also be the top-left corner of $B(f^*, R^*)$. Let r_1, r_2, \dots, r_n be the elements of S sorted in increasing L_∞ distance to u . Then now apply the same approach as used in cases (a) and (b) using the new ordering instead. The result thus follows. \square