The Heavy Luggage Metric

Manuel Abellanas
U. Politécnica de Madrid
Spain

Ferran Hurtado
U. Politècnica de Catalunya
Spain

Belén Palop
U. Valladolid
Spain

Abstract

In this paper we consider a new time metric called the Heavy Luggage Metric. This metric models the behavior of a traveller in a city wishing to walk as little as possible, maybe carrying some very heavy luggage. A transportation network allows the traveller to move between any pair of stations at cost zero, while walking has a cost proportional to the length of the walked path. We propose efficient algorithms for computing the closest and farthest Voronoi Diagrams for a set of points with respect to this metric.

Keywords: Voronoi Diagram, transportation network, time metric.

1. Introduction

Suppose you are coming back home from a trip and that you have bought lots of presents. Minimizing the time to reach your destination is no longer your highest priority. You just care about the distance you travel carrying this very heavy luggage. You wish you could enter the subway as soon as possible, sit, and wait until you reach the closest station to your house.

In order to model the previous situation, we consider a transportation network in the plane and assume that users can enter or leave the network only at some points called *stations* and that all stations are connected to each other through the network. Our main concern is the distance (or the time) users spend walking out of the network, since they want to minimize it.

Let S be the set of stations. Users will choose between two different paths either using or not using the network when they travel from p to q. If a straight path does not use the network, we assign it a cost equal to the Euclidean distance between the two points, $d(p,q)$. Otherwise, the path will *enter* and *exit* the network at stations $s_p \in S$ and $s_q \in S$. Note that, if drawn, the latter path from p to q is not a polygonal line but two disjoint segments, since we are assuming all stations are connected and at distance 0 from each other. The cost of such a path is the length of the portions outside the network, that is, $d(p,s_p)+d(s_q,q)$. We define a *heavy luggage shortest path* between two points p and q, $sp_h(p,q)$, as a path from p to q minimizing its cost.

We say that a set of points is in general position if no two of them are at the same distance to their closest station, no point is at the same distance to two stations, and no point lies in a station. From now on, we will always refer to points in general position.

Given a point $p \in \mathbb{R}^2 \setminus S$, we call s_p the station in S minimizing $d(p,s)$, and we use d_p^s for this distance. We trivially have:

Lemma 1. *Given two points $p,q \in \mathbb{R}^2 \setminus S$, the cost of a shortest path between p and q, $sp_h(p,q)$, is the minimum between $d(p,q)$ and $d_p^s + d_q^s$.*

Proof. If $sp_h(p,q)$ does not use the network, its cost equals the Euclidean distance between the points. Otherwise, suppose $sp_h(p,q)$ enters the network at some station $s \neq s_p$ (analogous reasoning can be done for s_q, and for both s_p and s_q). The cost of this path is $d(p,s)+d(s_q,q)$. But $d(p,s) > d(p,s_p)$, which contradicts the fact that $sp_h(p,q)$ is a heavy luggage shortest path. □

Definition 1. *Given two points $p,q \in \mathbb{R}^2 \setminus S$, the* heavy luggage distance *from p to q, $d_h(p,q)$ is the cost of a heavy luggage shortest path from p to q, that is,*

$$d_h(p,q) = \min\{d(p,q), d_p^s + d_q^s\}$$

2. Properties

Let us first see that the Heavy Luggage Metric fits in the general framework of *Time Metrics* (see [1]). Suppose we assign speed 1 to movements outside of the network and speed ∞ to movements inside the network. Any portion of a heavy luggage path between two stations is travelled in time 0 while, outside of the network, the time equals the Euclidean distance travelled. A shortest time path under this assumptions coincides with a heavy luggage shortest path.

For any given point p in the plane, we can find a division of the plane depending on whether a traveller in p will use the network or not when travelling from p. Let d_p^s be the distance from p to its nearest station s_p. If we assign to the point p the weight d_p^s and to each station weight 0, the Additively Weighted Voronoi Diagram of $S \cup \{p\}$ separates the plane into regions associated either to the point or to some station (see Figure 2). Each region is associated to the station where a traveller departing from p will leave the network in order to reach its destination, if it were in that region. The region associated to p is the locus of the target points where a traveller starting at p will decide not to use the network. We call this region the *Walking region of* p and denote it with $W(p)$.

3. Closest Voronoi Diagram

We give in this section an algorithm to compute the Voronoi diagram for a set of n points P and a set of k stations S with respect to the Heavy Luggage Metric. Notice that in many practical situations k will simply be a constant.

Let us call $\mathcal{V}_h^S(P)$ the Voronoi Diagram induced by the heavy luggage metric for a set of points P and a set of stations S. As usual, we call *Voronoi region* of a point $p \in P$, and denote it by $\mathcal{VR}_h^S(p, P)$, the locus of the points that are closer to p than to any other point in P with respect to the heavy luggage metric.

Let p_0 be the point in P which is closer to its nearest station among all points in P and let s_0 be its closest station in S, that is,

$$d_{p_0}^s = d(p_0, s_0) = \min_{\forall q \in P} \{d(q, s_q)\}.$$

Lemma 2. *The Heavy Luggage Voronoi Region of* p_0*,* $\mathcal{VR}_h^S(p_0, P)$*, contains all stations in* S*.*

Proof. The distance $d_{p_0}^s$ equals the heavy luggage distance $d_h(p_0, s)$ for all $s \in S$. For all $q \in P, q \neq p_0$ we have that $d_h(q, s_q) = d(q, s_q) > d_{p_0}^s$, which implies that $d_h(q, s_q) > d_h(p_0, s_0)$. □

Proposition 1. *The diagram* $\mathcal{V}_h^S(P)$ *can be constructed by merging some regions from the Additively Weighted Voronoi Diagram of the union of points* $p \in P$ *with weight* $d_{p_0}^s$ *and stations in* S *with weight* 0.

Proof. (A more general construction for time metrics with stations can be found in [3]) Call $AWVD(P, S)$ the above mentioned Voronoi Diagram. Lemma 2 shows that, by the time a traveller in p_0 reaches the station s_0 (and therefore any station in S), any traveller at a different point $q \in P$ has only been able to reach points inside the circle centered at q with radius $d_{p_0}^s$. Since points in P are assigned the same weight, points in the region of $p \in P$ in $AWVD(P, S)$ are closer to p (with respect to the Euclidean distance) than to any other point in P. Moreover, their distance to p is smaller than their distance to any station plus the distance from p_0 to its closest station. Hence, they are closer to q than to any other point in P with respect to the heavy luggage distance. On the other hand, any point t in the region of a station $s \in S$ satisfies obviously that $d(t, s) < d(t, s')$ for all $s' \in S, s' \neq s$ and that, for all $p \in P$, $d(t, p) - d_{p_0}^S > d(t, s)$. Therefore, $d(t, p) > d(t, s) + d_{p_0}^S = d_t^S + d_{p_0}^S = d_h(t, p_0)$ holds. This way, the regions corresponding to stations in $AWVD(P, S)$ together with p_0's own region are assigned to p_0 in $\mathcal{V}_h^S(P)$, while regions corresponding to points in P other than p_0 in $AWVD(P, S)$ are assigned to the same point in $\mathcal{V}_h^S(P)$ (Figure 3 shows an example this assignment). □

The preceding results lead to the following algorithm for constructing the closest heavy luggage Voronoi diagram:

1. Compute the Euclidean Voronoi diagram of the set of stations S in time $O(k \log k)$.

2. Find the closest point-station pair, (p_0, s_0), by locating each point in P in the previous Voronoi diagram in time $O(logk)$ and computing its distance to its corresponding station. The overall time of this step is $O(n \log k)$.

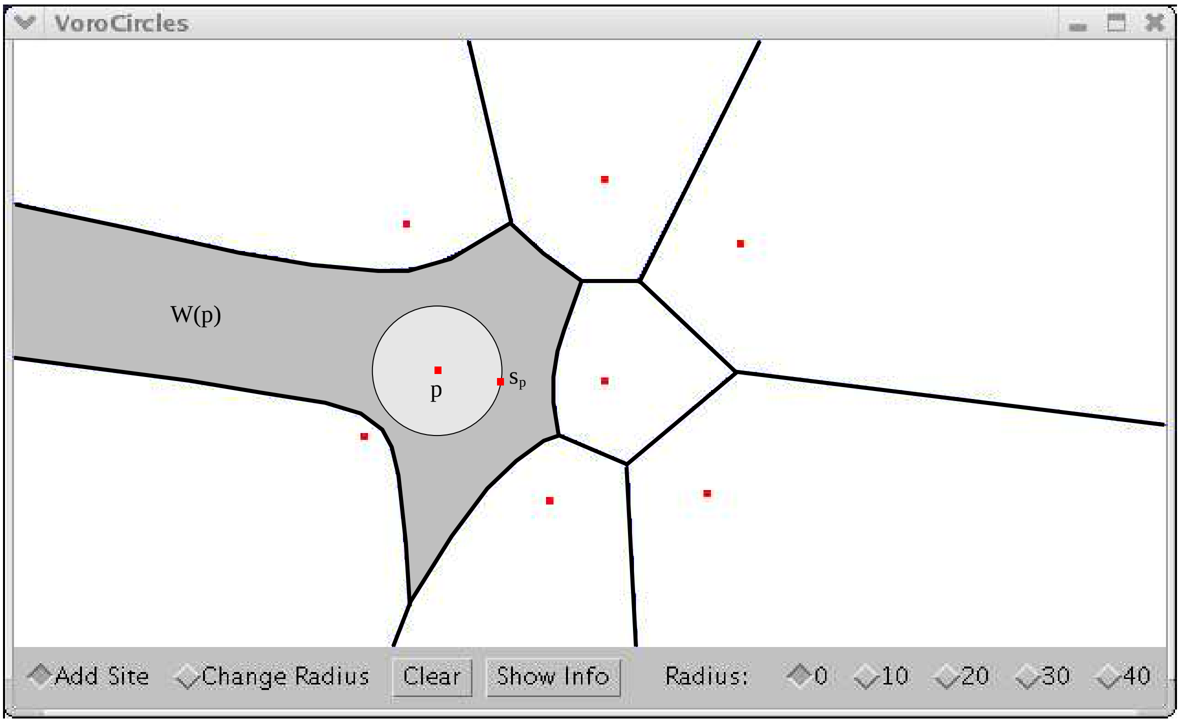

Figure 1. Exit station and walking decision for a traveller in p depending on his destination. (Picture made with aid of Voroadd [5])

3. Assign all points weight $d(p_0, s_0)$ and all stations weight 0 and compute the Additively Weighted Voronoi Diagram of this set in time $O((n+k)\log(n+k))$.

4. Merge the regions associated to stations in the previous Voronoi diagram and assign them to $\mathcal{V}R_h^S(p_0, P)$.

Theorem 1. *If the number of stations is constant, $\mathcal{V}_h^S(P)$ can be computed with the above algorithm in time $O(n \log n)$.*

4. Single-Exit Many-Entrances

In order to efficiently compute the farthest Voronoi diagram under the heavy luggage metric, an intermediate model is introduced in this section: The *SM-heavy luggage metric.*

Suppose now that travellers are allowed to enter the network through any station but that they can only exit at some fixed station $s_f \in S$. An *SM-heavy luggage shortest path* between two points p and q in the plane (with $d(q, s_f) < d(p, s_f)$) is, either the segment joining p and q, or the two segments $\overline{p\,s_p}$ and $\overline{s_f\,q}$. The *SM-heavy luggage distance* $d_{SM-h}(p, q)$ is the minimum between $d(p, q)$ and $d_p^S + d(q, s_f)$.

Let us now do a 3D construction in order to see how this distance behaves. For a point $p \in \mathbb{R}^2$ we construct a nappe C_p of a cone with axis in the positive z direction, apex in p and right opening angle; the rays with apex p contained in the cone form an angle of 45 degrees with the xy-plane. The Euclidean distance from any point q in the plane to the point p can also be measured as the vertical distance from q to the nappe C_p. For the station s_f, we lift vertically the nappe C_{s_f} by d_p^S; let us call $\widehat{C}_{s_f}$ this translated half-cone. The vertical distance from any point t in the plane to $\widehat{C}_{s_f}$ coincides with $d_p^S + d(t, s_f)$. The lower envelope of C_p and $\widehat{C}_{s_f}$ gives the minimum between the two distances, that is, the SM-heavy luggage distance from any point in the plane to point p.

Given two points p and q, we call *SM-heavy luggage bisector*, $B_{SM-h}(p, q)$, the locus of points at the same SM-heavy luggage distance from p and from q, that is,

$$B_{SM-h}(p, q) = \{x \in \mathbb{R}^2 : d_{SM-h}(x, p) = d_{SM-h}(x, q)\}.$$

Let us now perform the previous construction for two points p and q. The lower envelope of the four nappes allows us to decide which point, p or q, is

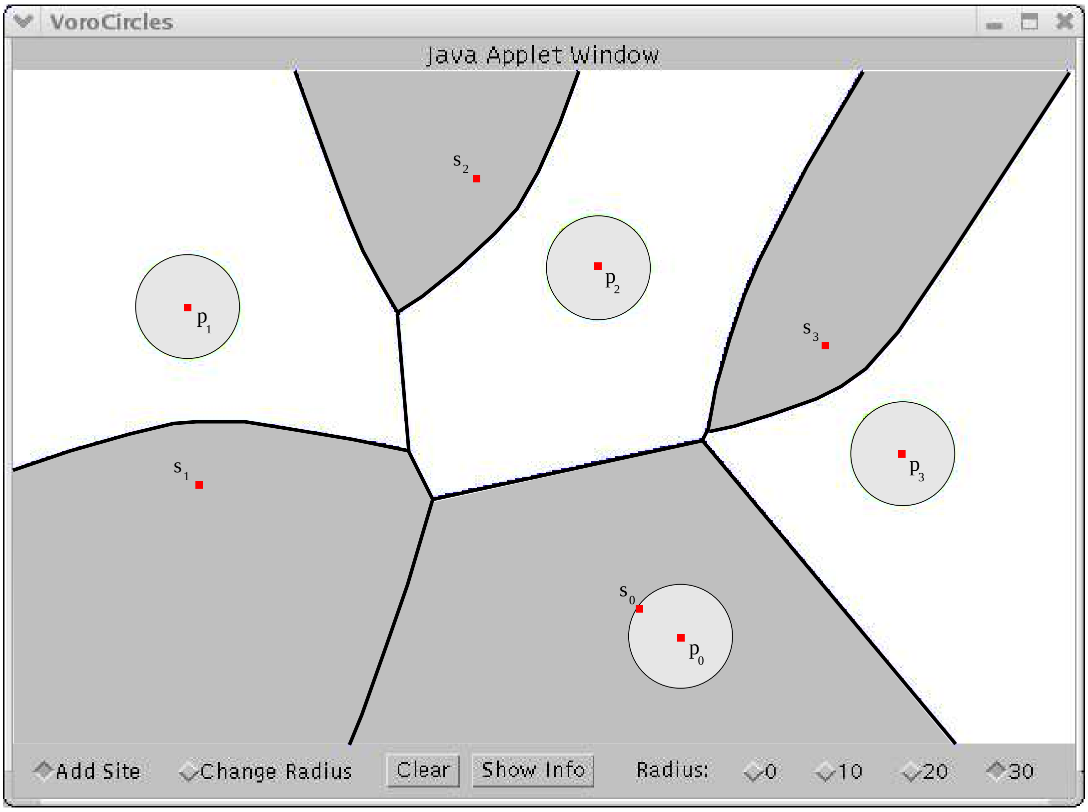

Figure 2. Example of a closest heavy luggage Voronoi diagram for four points and four stations. Shaded regions compose $\mathcal{V}R_h^S(p_0, P)$. (Picture made with aid of Voroadd [5])

closer to any given point in the plane, and its vertical projection onto the xy-plane is the SM-heavy luggage bisector of p and q.

From this construction, it is easy to see that SM-Bisectors behave *nicely* (in the sense of *nice metrics*, introduced in [2]), and that they are only composed of line segments and arcs of hyperbolae. The plane splits into two regions containing one site each; we use $D_{SM-h}(p,q)$ for the region where site p lies and $D_{SM-h}(q,p)$ for the region where q lies.

Lemma 3. *Given two points p and q, $B_{SM-h}(p,q)$ is homeomorphic to the open interval (0,1).*

Proof. Suppose $B_{SM-h}(p,q)$ is a closed curve and, w.l.o.g., that it contains p in its interior. Since the intersection of nappes with apex in p and q projects on a straight line, the dominance region $D_{SM-h}(p,q)$ has to be closed due to the nappes in the vertical line through s_f. But since their apex is above the half-cone starting in p, the arc of hyperbola made in the intersection is strictly contained in the Euclidean Voronoi region of s_f with respect to p and, therefore, in the halfplane containing s_f. As a consequence, the bisecting curve between p and q cannot enclose any of the two points. Moreover, since the nappes with apex in p and q start at a lower z-coordinate than those above s_f, at points far enough from s_f, only these two nappes appear in the lower envelope and two halflines from the Euclidean bisector of p and q go to infinity. $\square$

In fact, the set of the $O(n^2)$ bisectors given by pairs of points in a set P of n points, form an admissible dominance system. As Mehlhorn et al. proved in [4] the following theorem holds for Voronoi diagrams induced by these kind of systems:

Theorem 2. [4] *The farthest site abstract Voronoi diagram of a set of n sites can be computed by a randomized algorithm in expected time $O(n \log n)$ and expected space $O(n)$.*

We recall here the properties that a family of dominance regions must satisfy in order to be called admissible, since the framework of abstract Voronoi diagrams will be crucial in our construction.

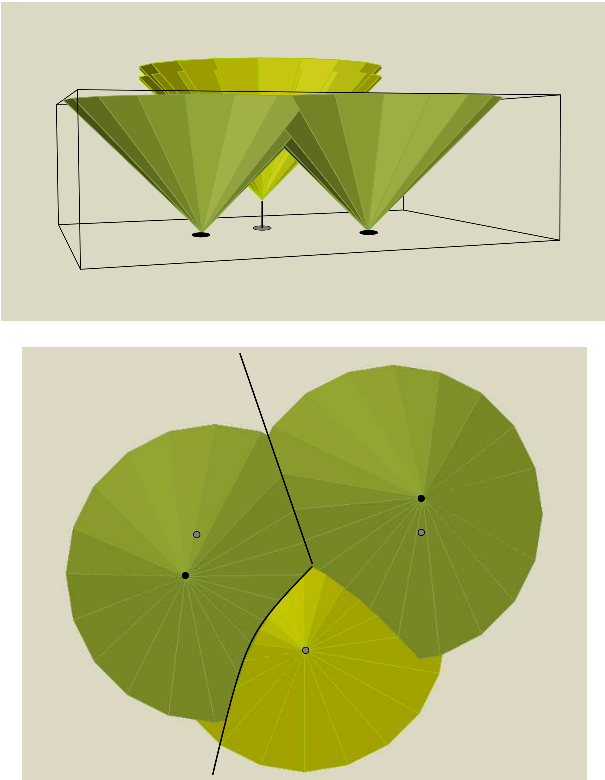

Figure 3. Finding the SM-Bisector of two points

We use $\mathcal{D}(p,q)$ for the *dominance region* of a site p with respect to a site q. In the metric case we are dealing with, we will refer to the dominance region as the set of points which are strictly closer to p than to q. It is important to note that, the non-degenerate setting of the points and stations guarantees that there are no 2-dimensional bisectors between points. In order to call a family of dominance regions a *dominance system*, the following properties must be satisfied:

1. $\mathcal{D}(p,q)$ is a non-empty open subset of the plane.
2. $\mathcal{D}(p,q) \cap \mathcal{D}(q,p) = \emptyset$ and $bd(\mathcal{D}(p,q)) = bd(\mathcal{D}(q,p))$.
3. $B(p,q) = bd\mathcal{D}(p,q)$ is homeomorphic to the open interval (0,1).

Moreover, a dominance system is called *admissible*, and hence defines a farthest site abstract Voronoi diagram, if two additional properties are satisfied. In our metric case, they can be stated as follows (see [4] for the exact definitions and conditions):

4. Any two bisecting curves intersect in a finite number of connected components.
5. Call the intersection of all dominance regions of a site with respect to the set of sites its *Voronoi region.*
   (a) Voronoi regions of all sites in the abstract Voronoi diagram of any subset of the set of sites are path-connected and have non-empty interior.
   (b) The union of all Voronoi regions and bisectors covers the plane.

Theorem 3. *The family of dominance regions given by $D_{SM-h}(p,q)$ for all pairs of sites p and q in P forms an admissible dominance system.*

Proof.

1. As follows from Lemma 3, $D_{SM-h}(p,q)$ is an open subset of the plane. Moreover, p is closer to p than to q with respect to the heavy luggage distance and, thus, $p \in D_{SM-h}(p,q)$ and the set is not empty.
2. Since the heavy luggage metric is well defined for all points which are not stations, dominance regions are disjoint and their boundary is exactly the curve $B_{SM-h}(p,q)$.
3. See Lemma 3.
4. Follows from the bisectors being formed by three segments of algebraic curves of degree one or two.
5. (a) Even though the condition of path-connectedness could be unfulfilled by the Voronoi region of some site $p \in P$, we can easily avoid this problem: We create an auxiliary site p' for every site p in P and assign it the region it might own due to the cone growing in s_f^x, s_f^y, d_p^S. After the abstract Voronoi diagram is computed, the regions associated to auxiliary sites, are assigned to the original points in the set of sites.
   (b) The complementary of $D_{SM-h}(p,q) \cup D_{SM-h}(p,q)$ is $B_{SM-h}(p,q)$. Hence, the Voronoi diagram covers the plane.

□

Corollary 1. *The farthest Voronoi diagram of a set of n sites P and a set of stations S with a single exit station, can be computed with a randomized algorithm in expected time $O(n \log n)$ and expected space $O(n)$.*

5. Farthest Voronoi Diagram

Let us now construct the farthest heavy luggage Voronoi diagram of a set of n points P and a set of stations S. For each point p in P, we define its farthest heavy luggage Voronoi region $\mathcal{V}R^{-1}(p,P)$ as the set of points which are farther from p than from any other point in P. The most difficult problem for this construction is that we are dealing with a *maxmin*-problem, that is, we are looking for a point maximizing a certain distance, which is defined as the minimum between two Euclidean distances. It is not obvious how to solve this problem in a direct way, even with the 3D cone construction, since we will have to compute the upper envelope of a set of n lower envelopes, each one produced by $k+1$ nappes (one for each station and one for the point itself). We give next a simpler way of computing this diagram using the SM-heavy luggage distance introduced in the previous section.

Let $VR(s,S)$ be the Euclidean Voronoi region of a station $s \in S$ with respect to the set of stations S.

Lemma 4. *If $t \in VR(s,S)$ is in the farthest Voronoi region of point $p \in P$, either $sp_h(p,t)$ is the segment joining them, or $\overline{sp} \subset sp_h(p,t)$.*

Proof. The first situation is trivial, since if t is in the Walking region of p, then $\overline{st} = sp_h(p,t)$. Otherwise, a traveller in p uses the set of stations to reach t. Suppose $s' \neq s$ were the exit station and, hence $sp_h(p,t) = \overline{p\,s_p} \cup \overline{s'\,t}$. The length of this path is $d_p^S + d(s',t)$, but $d(s',t) > d(s,t)$, since $t \in VR(s,S)$, which contradicts the fact that $sp_h(p,t)$ is a shortest path. □

An algorithm for computing the Farthest Voronoi diagram with respect to the heavy luggage metric follows. The correctness of the algorithm follows from the results in this and the previous sections.

1. For each station $s \in S$, compute the family of dominance regions with respect to the SM-heavy luggage distance supposing s is the exit station, which is an admissible dominance system for a set of points P in general position.

2. For each admissible dominance system, compute the farthest abstract Voronoi diagram of the set of points P.

3. Compute the Euclidean Voronoi diagram of the set of stations S.

4. For each cell $VR(s,S)$ in the previous diagram, compute its intersection with the SM-heavy luggage diagram where s is the exit station and merge all cells.

Theorem 4. *The above algorithm computes the farthest Voronoi diagram of a set of n points P and a set S of stations in time $O(n^2)$ considering the number of stations to be constant. Otherwise, the algorithm is $O(kn^2)$.*

Proof. Step 1 can be carried out, for each station, in time $O(n^2)$, since we have to compute a quadratic number of bisecting curves. Thus, the complexity is $O(kn^2)$.
As follows from Theorem 2, Step 2 has expected cost $O(n \log n)$ (and worst case complexity $O(n^2)$) and has to be performed k times.
Step 3 is trivially $O(k \log k)$.
Finally, the intersection of one of the computed diagrams with a convex polygon can be performed in time $O(kn)$, since all diagrams have linear complexity. □

Figure 5 shows an example of a farthest Voronoi diagram with respect to the heavy luggage metric. Black dots represent stations and grey dots represent sites. It is interesting to see how some stations, like the one labeled $s5$, produce up to three arcs of hyperbola: one bisector for each site ($p1$,$p3$ and $p4$) different to the farthest one ($p2$). *Close* to the points, the Euclidean farthest Voronoi diagram appears, while close to some stations, like the above mectioned, Euclidean bisectors not belonging to that diagram may appear.

6. Conclusions

The Heavy Luggage Metric that we have studied in this paper can be considered as an extreme case in the Time Metrics framework: travellers do not care about the time they spend inside the network, only the external part is meaningful. In the opposite extreme we would have the models for long distance travelling, in which walking time to stations may be disregarded. There are obviously several trade-offs between these extremes that are interesting in some other real situations and are the topic of our current research.

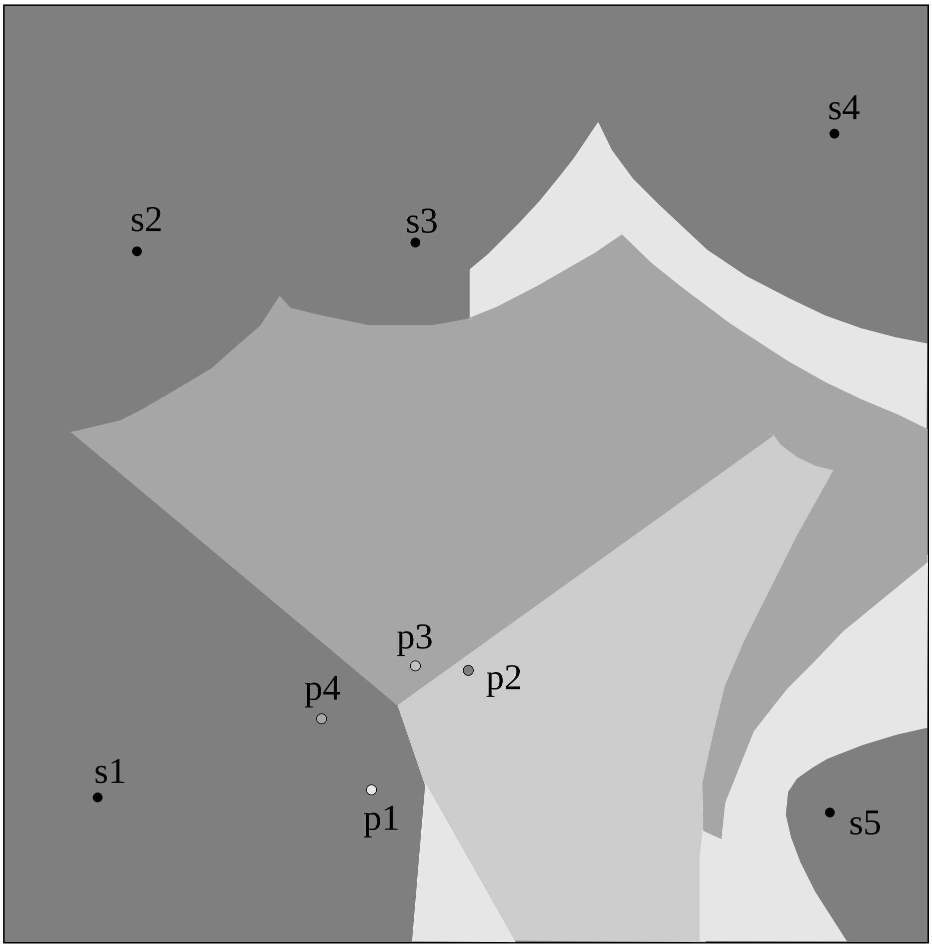

Figure 4. Farthest Voronoi diagram with respect to the heavy luggage metric for five stations and four points.

Acknowledgments

Work by Ferran Hurtado is partially supported by grants DURSI 2001SGR00224 and MCYT/MEC BFM2003-0368. Work by Manuel Abellanas and Belén Palop is partially supported by TIC2003-08933-C02-01.

References

[1] Abellanas, M., Hurtado, F., Icking, C., Klein, R., Langetepe, E., Ma, L., Palop, B., and Sacristán, V. Voronoi diagrams for services neighboring a highway. *Information Processing Letters 86* (2003), 283–288.

[2] Klein, R. *Concrete and Abstract Voronoi Diagrams*, vol. 400 of *Lecture Notes Comput. Sci.* Springer-Verlag, 1989.

[3] Ostrovsky-Berman, Y. Computing Transportation Voronoi Diagrams in Optimal Time. *21st European Workshop on Computational Geometry*, (2005), 159-162.

[4] Mehlhorn, K., Meiser, S., and Rasch, R. Furthest Site Abstract Voronoi Diagrams. *Int. J. Comput. Geometry Appl. 11(6)* (2001), 583-616.

[5] Münch,O. Available at http://www.pi6.fernuni-hagen.de/GeomLab/voroadd/ (5 September 2005).