

## MOTION PLANNING OF LEGGED ROBOTS: THE SPIDER ROBOT PROBLEM

JEAN-DANIEL BOISSONNAT\*†

OLIVIER DEVILLERS\*‡

LEONBATTISTA DONATI\*§

*INRIA, BP 93, 06902 Sophia-Antipolis cedex (France)*

FRANCO P. PREPARATA¶

*Department of Computer Science, Brown University, Providence, RI 02912, (USA)*

Received 5 August 1992

Revised 20 December 1993

Communicated by F. Aurenhammer

### ABSTRACT

We consider the problem of planning motions of a simple legged robot called the spider robot. The robot is modelled as a point where all its legs are attached, and the footholds where the robot can securely place its feet consist of a set of  $n$  points in the plane. We show that the space  $\mathcal{F}$  of admissible and stable placements of such robots has size  $\Theta(n^2)$  and can be constructed in  $O(n^2 \log n)$  time and  $O(n^2)$  space. Once  $\mathcal{F}$  has been constructed, we can efficiently solve several problems related to motion planning.

*Keywords:* Spider robot, Legged robot, Motion planning

### 1. Introduction

Although legged robots already exist, until now researchers have been more interested in their dynamics and their control (see general references<sup>1,2</sup>) than in motion planning problems : the literature on the topic is almost nonexistent and we are aware of only very few papers that consider the problem of planning the movement of the legs of a legged robot moving amidst obstacles and dangerous areas<sup>3,4</sup>.

---

\* *Work of these authors has been supported in part by the ESPRIT Basic Research Actions Nr. 3075 (ALCOM), 7141 (ALCOMII) and 6546 (PROMotion).*

† *Jean-Daniel.Boissonnat@sophia.inria.fr.*

‡ *Olivier.Devillers@sophia.inria.fr.*

§ *donati@math.unice.fr. Currently at the University of Nice (France).*

¶ *franco@cs.brown.edu.*

*Work of this author has been supported in part by NSF Grant CCR-91-96176 and by ONR Contract N00014-91-J4052, ARPA order 8225.*

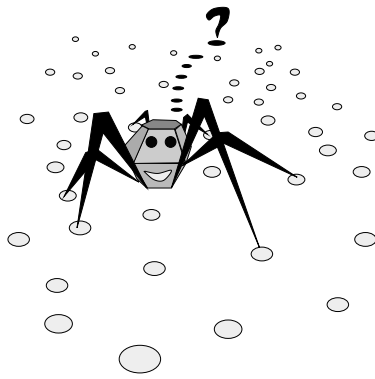


Fig. 1. The spider robot problem

The robot we consider here is a simple legged robot, called a spider robot. The body is a single point; a number of segments (the legs), whose lengths may vary within bounds, are attached to the body (see Figure 1). This model has been inspired by the Ambler,<sup>5</sup> a legged robot developed at The constraints on the robot motion are of two types; first, the robot feet have to rest on reachable footholds (feasibility constraint) and, second, each position of the robot must be stable (stability constraint). In this paper, the footholds are points in the plane. This simple model is valid for very constrained environments where the robot can only put its feet in a few places. It can also be of interest for less constrained cases after sampling the footholds area. We show that the space of valid configurations, that is the set of all the positions of the robot body for which there exists a feasible and stable placement of its feet, has size  $\Theta(n^2)$  where  $n$  is the number of predefined footholds. Next we give an algorithm that computes this subset of the plane in time  $O(|\mathcal{A}| \log n)$  and  $O(|\mathcal{A}|)$  space where  $|\mathcal{A}|$  is the size of a certain arrangement of  $n$  circles, which in the worst-case is  $\Omega(n^2)$ , but in most realistic situations will be  $O(n)$ . This set of stable configurations can readily be used to decide whether or not there exists a feasible and stable path between two given placements. It can also be used in combination with  $\mathcal{A}$  to compute a sequence of stable placements for the feet along that path.

The paper is organized as follows: in the next section, we introduce the necessary definitions for the robot and we make precise the problems to be solved. In Section 3 we study the space of feasible and stable configurations and give a tight bound on its complexity. Section 4 presents the algorithm that computes the space of stable configurations. Applications of the present work to solve several motion planning problems and some concluding remarks are presented in the last two sections.

## 2. Definitions

The body of a spider robot is a single point and will be denoted  $G$ ; each foot of the robot can reach the points of the plane inside the disk of radius  $R$  centered at  $G$ : this disk is called the *range of action* of the robot. If  $q$  is a point in the plane;  $D(q)$  denotes the closed disk of radius  $R$  centered at  $q$ .

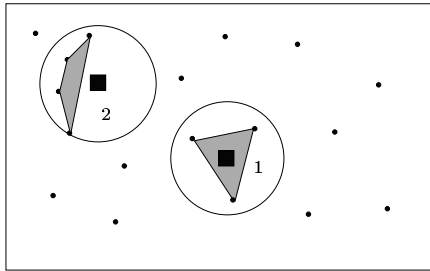


Fig. 2. Two configurations

The footholds where the robot feet can stay safely consist of a set  $\mathcal{M}$  of  $n$  points (the *sites*) of the Euclidean plane.

As the problem we consider is essentially planar, all figures, except Figure 1, will be drawn in the plane and we will assume that  $G$  is also a point of the plane (we identify  $G$  and its vertical projection onto the plane).

### 2.1. Configurations and Placements

We call *configuration* of the robot a position of its body  $G$ . For a given configuration, we call *placement* of the robot a set of pairings between some of the robot feet and some points of  $\mathcal{M}$ . A placement is defined by  $\mathcal{I}$ , the set of resting legs and by  $p_i \in \mathcal{M}, i \in \mathcal{I}$ , the position of the feet. We will say that a placement is a  *$l$ -leg placement* if  $|\mathcal{I}| = l$ .

For a given configuration, a placement is said  *$l$ -feasible* if there is a set of indices  $\mathcal{I}$  such that

$$\forall i \in \mathcal{I}, d(G, p_i) \leq R \text{ and } |\mathcal{I}| \geq l \quad (1)$$

where  $d(A, B)$  is the Euclidean distance between two points  $A$  and  $B$  of the plane. Eq. (1) says that there must be at least  $l$  sites in the range of action of the robot.

A configuration is said to be  *$l$ -feasible* if there is an  $l$ -feasible placement for this configuration. In Figure 2, configuration 1 is 3-feasible and configuration 2 is 4-feasible (the position of  $G$  is given by the black square).

Lastly, for a given configuration, a placement is said to be *stable* if the following condition holds:

$$G \in \text{CH}(\{p_i, i \in \mathcal{I}\}), \quad (2)$$

where  $\text{CH}(S)$  denotes the convex hull of  $S$ . As above, a robot configuration is *stable* if there is a stable placement for the configuration (only configuration 1 is stable in Figure 2).

### 2.2. Paths and Motions

We call  *$l$ -path* of the spider robot, a continuous trajectory  $\mathcal{C}$  of  $l$ -feasible configurations; that is, for each position of  $G$  along  $\mathcal{C}$  there must be an  $l$ -feasible placement. An  *$l$ -motion* consists of an  $l$ -path  $\mathcal{C}$  of  $G$  together with a sequence of possible placements.

A *stable path* or *motion* is a path or a motion in which every position of the robot body satisfies the stability condition 2.

It should be added that along an  $l$ -motion there could be configurations where the robot has to change its placement: that is, without changing the position of its body, it has to change the resting places of its legs. This is possible if the robot has an  $l + 1$ th leg; with this additional leg, it can change its resting sites while always keeping  $l$  legs on the ground.

### 3. Feasible and Stable Configurations

#### 3.1. Relaxing One Constraint

In this section, we show that the problem can be easily solved if only one constraint, the feasibility or the stability constraint, is considered.

Clearly, if we relax the feasibility constraint, the set of stable configurations is the convex hull of the sites, which can be computed in  $O(n \log n)$  time.<sup>6</sup>

On the other hand, if we relax the stability constraint, the set of  $l$ -feasible (not necessarily stable) configurations can be deduced from the order- $l$  Voronoi diagram. Indeed, for a given configuration  $G$  of the robot, there exists an  $l$ -feasible placement as soon as the placement defined by setting  $l$  feet on the  $l$  nearest footholds is  $l$ -feasible.

More precisely, we compute the superposition of the order- $l$  and order- $(l - 1)$  Voronoi diagrams. For any point of a given cell  $V_i$  of the resulting map  $\mathcal{D}$ , the  $l$  nearest footholds are the same, say  $p_{i_1}, p_{i_2}, \dots, p_{i_l}$  and the furthest among these ones is a unique site  $p_{i_l}$ . Thus  $W_i = V_i \cap D(p_{i_l})$  is the portion of  $V_i$  consisting of  $l$ -feasible configurations (all footholds can be reached as soon as  $p_{i_l}$  is reachable since it remains the furthest foothold inside  $V_i$ ). The union  $W$  of the  $W_i$  for all the cells  $V_i$  of  $\mathcal{D}$  is the whole set of  $l$ -feasible configurations. The sizes of the order- $(l - 1)$  and order- $l$  Voronoi diagrams, as well as the size of their superposition<sup>a</sup> are  $O(nl)$  and they can be computed in  $O(n \log n + nl^2)$  time.<sup>7,8</sup> Constructing  $W$  can be done within the same time bound.

Unfortunately, Voronoi diagrams do not help solving the problem of the stability of the robot. It may happen that the nearest footholds placement is not stable while there exists another placement which is both feasible and stable.

The rest of the section and of the paper will be concerned with both constraints.

#### 3.2. Preliminary Results

The set  $\{D(p), p \in \mathcal{M}\}$  defines an arrangement  $\mathcal{A}$  of circles.

The number of edges of  $\mathcal{A}$  will be called the size of  $\mathcal{A}$  and will be denoted by  $|\mathcal{A}|$ . By slightly adapting standard techniques for line segments,  $\mathcal{A}$  can be computed in  $O(|\mathcal{A}| \log n)$  time.<sup>6</sup>

---

<sup>a</sup>As noticed by Lee<sup>7</sup> the edges of the order- $(l - 1)$  and order- $l$  Voronoi diagrams intersect only in their end-points, thus the size of the superposition is the same as the size of these two diagrams.

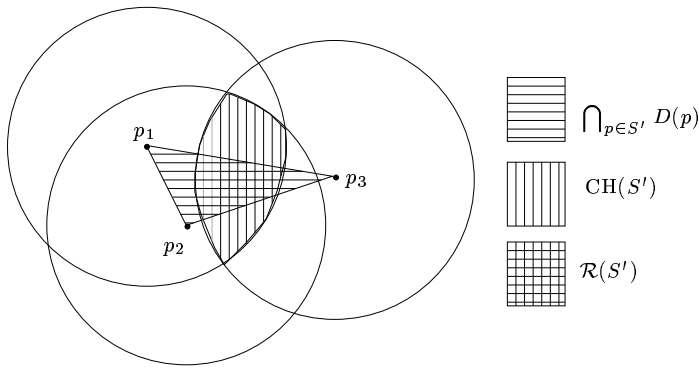


Fig. 3. Stability region defined by  $S = \{p_1, p_2, p_3\}$

For any  $S \subseteq \mathcal{M}$ ,  $|S| \geq 3$ , we call *stability region* defined by  $S$  the following subset of the Euclidean plane:

$$\mathcal{R}(S) = \text{CH}(S) \cap \left( \bigcap_{p \in S} D(p) \right). \quad (3)$$

Figure 3 shows the stability region defined by a subset of three sites.

**Definition 1**  $\mathcal{F}$  denotes the union of the stability regions associated with all the subsets of  $\mathcal{M}$  of cardinality 3:

$$\mathcal{F} = \bigcup_{\substack{T \subseteq \mathcal{M} \\ |T|=3}} \mathcal{R}(T). \quad (4)$$

$\mathcal{F}$  is the set of stable 3-feasible configurations.

**Lemma 2** For any  $S \subseteq \mathcal{M}$ , with  $|S| \geq 3$ , we have

$$\mathcal{R}(S) \subseteq \mathcal{F}. \quad (5)$$

**Proof.** As is well known,

$$\text{CH}(S) = \bigcup_{\substack{T \subseteq S \\ |T|=3}} \text{CH}(T).$$

Thus

$$\begin{aligned} \mathcal{R}(S) &= \left( \bigcup_{T \subseteq S, |T|=3} \text{CH}(T) \right) \cap \left( \bigcap_{p \in S} D(p) \right) \\ &= \bigcup_{T \subseteq S, |T|=3} \left( \mathcal{R}(T) \cap \bigcap_{p \in S \setminus T} D(p) \right) \\ &\subseteq \bigcup_{T \subseteq S, |T|=3} \mathcal{R}(T) = \mathcal{F} \end{aligned}$$

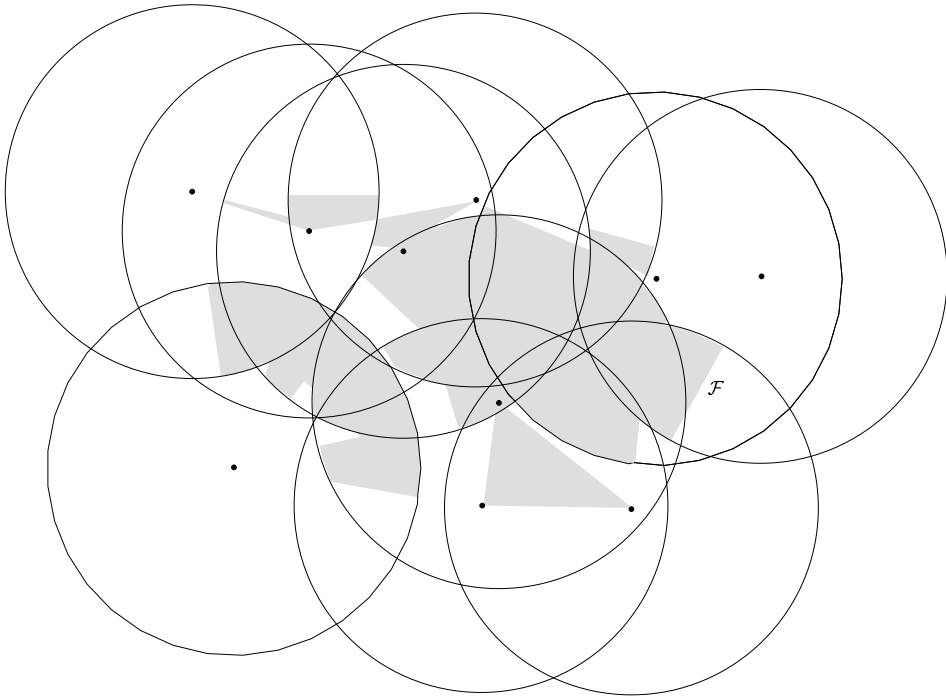


Fig. 4. An example of free space of a spider robot

□

The lemma shows that the complete information about stable regions is contained in  $\mathcal{F}$ . In particular, if there is no stable 3-leg placement, there is no stable  $l$ -leg placement with  $l > 3$ .

This is the reason why we shall focus our attention on the set  $\mathcal{F}$  and take  $l = 3$  in the sequel. However, this is not crucial and our analysis could be extended in a straightforward way to the analysis and the construction of the set of stable  $l$ -feasible configurations ( $l > 3$ ).

$$\mathcal{F}_l = \bigcup_{T \subseteq S, |T|=l} \mathcal{R}(T). \quad (6)$$

In particular, Lemma 2 can be generalized and we can prove that for any  $S \subseteq \mathcal{M}$ , with  $|S| \geq l$ , we have  $\mathcal{R}(S) \subseteq \mathcal{F}_l$ .

The boundary of  $\mathcal{F}$ , denoted by  $\delta(\mathcal{F})$ , consists of a disjoint union of cycles whose edges are arcs of circles  $C_i = \delta(D(p_i))$  and straight-line segments belonging to lines passing through two sites (see Figure 4). Analogously, we can define  $\delta(\mathcal{F}_l)$ ,  $l > 3$ .

We shall classify the vertices of  $\delta(\mathcal{F})$  on the basis of their internal angle (*convex* if the internal angle is less than  $\pi$ , *concave* otherwise) and on the basis of their incident edges (*SS* for segment-segment, *CC* for circular arc-circular arc, and *SC* for segment-circular arc).

Clearly, for any point  $x$  of the interior of a cell  $\Gamma$  of  $\mathcal{A}$ , the set  $\mathcal{M} \cap D(x)$  of sites inside  $D(x)$  is invariant. This set will be denoted  $S_\Gamma$ .  $\mathcal{R}(S_\Gamma)$  is the stability

regions of the sites reachable from  $\Gamma$ . We denote by  $\mathcal{A}_l$  the set of all the cells  $\Gamma$  of  $\mathcal{A}$  satisfying  $|S_\Gamma| \geq l$ .

**Lemma 3** *Let  $\Gamma$  be a cell of  $\mathcal{A}_3$ . We have*

$$\mathcal{F} \cap \Gamma = \mathcal{R}(S_\Gamma) \cap \Gamma.$$

**Proof.** From Lemma 2, we have

$$\mathcal{R}(S_\Gamma) \cap \Gamma \subseteq \mathcal{F} \cap \Gamma.$$

Conversely, assume that  $x \in \mathcal{F} \cap \Gamma$ . Then, by definition, there exist three sites  $p_1, p_2$ , and  $p_3$ , such that

$$x \in \mathcal{R}(\{p_1, p_2, p_3\}) \cap \Gamma$$

As the disks  $D(p_i)$  ( $i = 1, 2, 3$ ) cannot intersect the boundary of  $\Gamma$  transversally,

$$\mathcal{R}(\{p_1, p_2, p_3\}) \cap \Gamma \subseteq \mathcal{R}(S_\Gamma),$$

which proves the lemma.  $\square$

**Corollary 4** *Each  $SS$  vertex of  $\delta(\mathcal{F})$  is a site  $p \in \mathcal{M}$ .*

**Proof.** Let  $u$  be an  $SS$  vertex of  $\delta(\mathcal{F})$ , intersection of two straight-line edges. If  $u$  belongs to a cell  $\Gamma$  of  $\mathcal{A}$ , then, by Lemma 3,  $u$  is a vertex of  $\mathcal{R}(S_\Gamma)$  and by Eq. (3),  $u$  must be a vertex of  $\text{CH}(S_\Gamma)$  that is a site in  $\Gamma$ . We can also remark that every  $SS$  vertex is convex.  $\square$

**Corollary 5** *The vertices of  $\delta(\mathcal{F})$  either are sites or belong to the boundary of a cell of the arrangement  $\mathcal{A}$ .*

**Proof.** We already know that  $SS$  vertices are sites. On the other end, if a vertex is not an  $SS$  vertex, then it is defined by at least one arc of circle and thus belongs to the boundary of a cell of  $\mathcal{A}$ .  $\square$

### 3.3. Complexity of $\delta(\mathcal{F})$

**Lemma 6** *Any edge of the arrangement  $\mathcal{A}$  contains at most four  $SC$  vertices.*

**Proof.** Let  $C$  be a circular edge of  $\mathcal{A}$  contained in  $m \geq 3$  disks and let  $q$  be the center of the circle containing  $C$ . There is a unique maximal subset  $S \subseteq \mathcal{M}$  such that

$$C \subset \bigcap_{p \in S} D(p). \tag{7}$$

Let  $m = |S|$ ,  $P = \text{CH}(S)$  and  $P' = \text{CH}(S')$  with  $S' = S - \{q\}$ . We denote by  $p_1$  and  $p_2$  the two vertices of  $P'$  belonging to the two lines passing through  $q$  and tangent to  $P'$ , and by  $T$  the triangle  $q, p_1, p_2$ .

The  $SC$  vertices belonging to  $C$  are intersection points between  $C$  and  $P = P' \cup T$ . It follows that an  $SC$  vertex on  $C$  is either an intersection between  $C$  and  $P'$  or an intersection between  $C$  and  $T$  (specifically, an intersection between  $C$  and either of the supporting segments joining  $q$  to  $P'$ ).

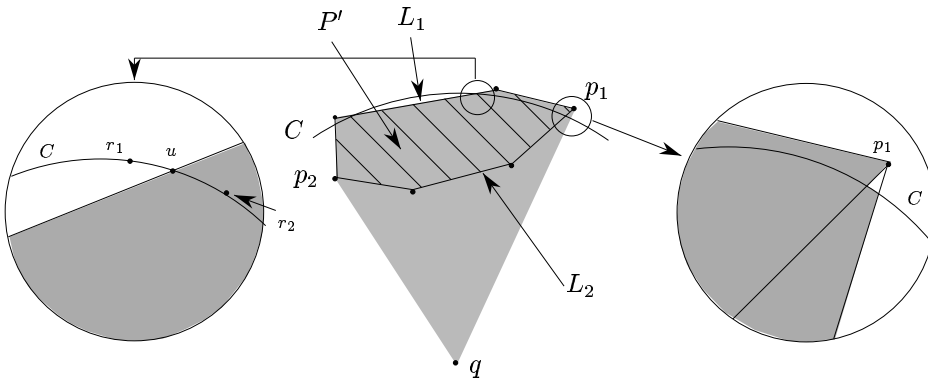


Fig. 5. Mixed vertices on  $C$

We consider first the case of an intersection between  $C$  and  $P'$ . Let  $L_1$  be the polygonal chain consisting of all the common edges of  $P$  and  $P'$  ( $p_1$  and  $p_2$  being its two endpoints). We claim that, if  $|S'| \geq 3$ , none of the intersections of  $L_1$  and  $C$  can be an  $SC$  vertex of  $\delta(\mathcal{F})$ . Indeed, let  $u$  be one of those intersections and consider a point  $r$  on  $C$  sufficiently close to  $u$  :

- if  $r \notin P$  ( $r_1$  in Figure 5), it cannot correspond to a stable placement of the robot since the only sites of  $\mathcal{M}$  which are in  $D(r)$  are contained in  $S$  and  $r$  is external to the convex hull of  $S$ , thus  $r \notin \mathcal{F}$ ;
- if  $r \in P$  ( $r_2$  in Figure 5), then  $r$  is a stable configuration and there exists a placement that does not involve  $q$  since  $P'$  has at least three vertices. Thus  $r$  is in the interior of  $\mathcal{F}$  and  $C$  cannot be an edge of  $\mathcal{F}$  in the neighborhood of  $u$ .

Let us now consider the remaining polygonal chain of  $P'$ , called  $L_2$  in Figure 5. Its intersections with  $C$  can actually be  $SC$  vertices but simple considerations of convexity show that there are at most two such vertices. As for the supporting segments  $qp_1$  and  $qp_2$ , each can contribute at most one  $SC$  vertex, there are at most four  $SC$  vertices of  $\delta(\mathcal{F})$  on  $C$ .

In the case where  $|S| = 3$  ( $|S'| = 2$ ) the portions of  $C$  inside  $T$  belong to  $\delta(\mathcal{F})$ . As the intersection of  $T$  with  $C$  consists of at most two circular arcs,  $C$  contributes at most four  $SC$  vertices.

In each case, we can conclude that  $C$  contains at most four  $SC$  vertices.  $\square$

Now that we have bounded the number of mixed vertices lying on each arc of  $\mathcal{A}$ , we can state the following theorem, where  $|\mathcal{A}|$  denotes the size of  $\mathcal{A}$ .

**Theorem 7** *The geometric complexity of  $\delta(\mathcal{F})$  is  $\Theta(|\mathcal{A}|)$ .*

**Proof.** The vertices of  $\delta(\mathcal{F})$  are sites ( $O(n)$ ), or vertices of  $\mathcal{A}$  ( $O(|\mathcal{A}|)$ ), or  $SC$  vertices ( $O(|\mathcal{A}|)$  from Lemma 6). Since,  $\delta(\mathcal{F})$  contains as many edges as vertices, the same bound  $O(|\mathcal{A}|)$  holds for the number of edges of  $\delta(\mathcal{F})$ .  $\square$

This bound is tight as is shown by the following example.

### 3.4. Worst-case Size Example

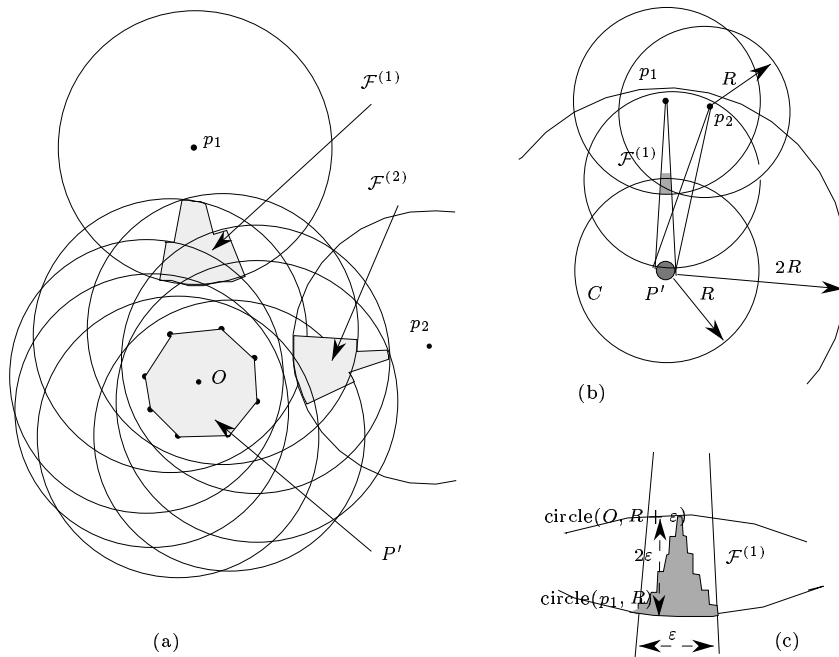


Fig. 6. Construction of  $O(n^2)$  configuration space

In the worst case, the complexity of  $\mathcal{A}$  is  $\Omega(n^2)$ . We exhibit in this section an example of a configuration set  $\mathcal{F}$  whose boundary has size  $\Omega(n^2)$ .

Put  $\frac{n}{2}$  sites at the vertices of a regular polygon  $P'$  and  $\frac{n}{2}$  other sites  $p_1, \dots, p_{\frac{n}{2}}$  at the vertices of a larger regular polygon  $P$  with the same center  $O$  as the first one (Figure 6a). Let  $\mathcal{F}^{(i)}$  denote the portion of  $\mathcal{F}$  contained in  $D(p_i)$ . It is possible to choose the diameters of the polygons such that

- the size of boundary of  $\mathcal{F}^{(i)}$  is  $\Omega(n)$ , for  $i = 1, \dots, \frac{n}{2}$ ;
- the  $\mathcal{F}^{(i)}$ 's are disjoint.

This is ensured by the following construction (refer to Figure 6b). Let  $\varepsilon$  be the radius of  $P'$ . We take the radius of  $P$  to be  $2R - \varepsilon$ . For a sufficiently small  $\varepsilon$   $|\mathcal{F}^{(1)}| = \Omega(n)$  (see Figure 6a). Moreover,  $\mathcal{F}^{(1)}$  is contained in the intersection of the disk with center  $O$  and radius  $R + \varepsilon$ , the disk  $D(p_1)$ , and the wedge with apex  $p_1$  tangent to  $P'$ . It follows that the diameter of  $\mathcal{F}^{(1)}$  is smaller than  $\sqrt{5}\varepsilon$  and that  $\mathcal{F}^{(1)}$  intersects  $C$ . If the length of the edges of  $P'$  is larger than  $2\sqrt{5}\varepsilon$  the  $\mathcal{F}^{(i)}$ 's are disjoint. Thus the set of stable configurations has boundary of size  $\Omega(n^2)$ .

In this example,  $\mathcal{F}$  has  $\Omega(n)$  connected components, each with boundary of size  $\Omega(n)$  but it is easy to slightly deform the figure so as to connect all the  $\mathcal{F}^{(i)}$ 's and obtain a single connected component of  $\mathcal{F}$  whose boundary has size  $\Omega(n^2)$ .

This example can also be adapted to reach the bound  $O(|\mathcal{A}|)$  of Theorem 7 for  $|\mathcal{A}| = o(n^2)$ . For example, by choosing  $|P| = O(1 + \frac{|\mathcal{A}|}{n})$  and  $|P'| = O(n)$ .

However, these worst case examples will not be encountered in most practical situations and much better bounds are to be expected. In order to illustrate this point, let  $k$  be the maximum number of disks of  $\{D(p), p \in \mathcal{M}\}$  that can cover a point of the plane.  $k$  will be called the *density* of the footholds. Clearly  $k$  is not larger than  $n$  and in case of *sparse footholds*, it is a constant that does not depend on  $n$ . It can be shown that  $|\mathcal{A}| = O(kn)$ <sup>9</sup>. Thus, in case of sparse footholds, the sizes of  $\mathcal{A}$  and  $\delta(\mathcal{F})$  are linearly related to the number of footholds.

#### 4. Computation of $\delta(\mathcal{F})$

Our algorithm is based on Lemma 3.

First, we construct the arrangement  $\mathcal{A}$  of the circles slightly adapting any standard  $O(|\mathcal{A}| \log n)$  time algorithm for line segments, for example using the plane sweep technique.<sup>6</sup>

We then have to intersect each cell with its *stability hull*, i.e. the convex hull of the centers of the disks covering the cell. Note that we do not need to recompute the entire hull for every cell since the sets of covering disks of two adjacent cells differ only by one element.

To exploit this property, we have to traverse all the cells of  $\mathcal{A}$  in an appropriate order so that two consecutive cells along the path in  $\mathcal{A}$  are incident in the arrangement. Notice that we do not need a Hamiltonian path, a cell may be visited several times provided that the size of the path is bounded by  $O(|\mathcal{A}|)$  (a depth first traversal of the adjacency graph of  $\mathcal{A}$  provides such a path). When a new cell is encountered along this path, we update the set of disks covering the cell by inserting or removing a disk. The path, and thus the whole sequence of insertions and deletions to be applied to the stability hull is computed in a first step. This allows to use the off-line maintenance of convex hulls of Hershberger and Suri<sup>10</sup> for the computation of the stability hulls. Here the length of the sequences is  $O(|\mathcal{A}|)$ , thus Hershberger and Suri algorithm needs  $O(|\mathcal{A}| \log n)$  time and  $O(|\mathcal{A}|)$  space. We obtain the stability hulls of all the cells of  $\mathcal{A}$  and a data structure which allows to perform binary search on the boundary on any hull. Queries can be answered in logarithmic time, if the queries are known in advance : they are processed during the traversal of  $\mathcal{A}$ , when the relevant cell is visited. If the queries are not known in advance, they can be processed afterwards, using  $O(|\mathcal{A}| \log n)$  memory space.

Suppose we are in a cell  $A_1$  of the arrangement and that we know, from the convex hull maintenance algorithm, the stability hull  $P_1$  of  $A_1$ . Let  $A_2$  be a cell adjacent to  $A_1$ , so that their common arc  $C$  turns its concavity towards  $A_2$  (see Figure 7) and let  $q$  be the center of the circle containing  $C$ . The stability hull  $P_2$  of  $A_2$ , which is  $CH(P_1 \cup \{q\})$ , is computed in  $O(\log n)$  time using the convex hull maintenance algorithm which also provides the following data:

- the polygonal chain  $L_2$  whose edges are edges of  $P_1$  and not of  $P_2$ ,
- the polygonal chain  $L_1$  whose edges belong to both  $P_1$  and  $P_2$ ,
- the two supporting lines  $s_1$  and  $s_2$  from  $q$  to  $P_1$ .

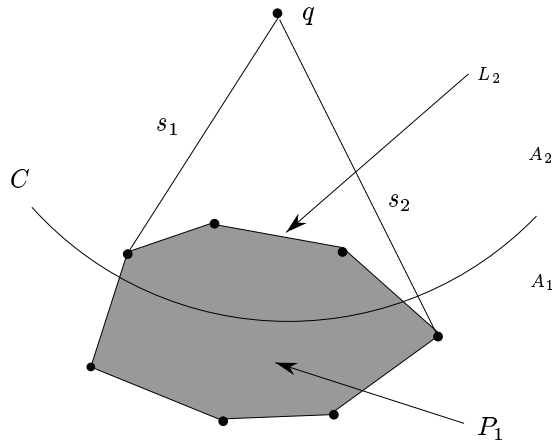


Fig. 7. Construction of  $\mathcal{F}$

The situation shown in Figure 7 is similar to the one in Figure 5: as before the  $SC$  vertices belonging to  $C$  can only lie on  $L_i$ ,  $s_1$ , and  $s_2$ . Here  $L_i$  stands for  $L_1$  if  $L_1$  is a line segment and for  $L_2$  otherwise. The search on  $s_1$  and  $s_2$  can be done in constant time; on the other hand, since  $C$  and  $L_i$  intersect at most twice, by binary search, we can locate the intersections between  $C$  and  $L_i$  in time  $O(\log n)$ .

Once we know, for a given cell, the  $SC$  vertices lying on its boundary and its stability hull, we can easily combine these data to get the portion of  $\delta(\mathcal{F})$  in the cell. Repeating this construction during the traversal of  $\mathcal{A}$  gives the entire  $\delta(\mathcal{F})$ .

**Theorem 8** *The boundary of  $\mathcal{F}$  can be computed in time  $O(|\mathcal{A}| \log n)$  and space  $O(|\mathcal{A}|)$ .*

**Proof.** For each cell of  $\mathcal{A}$  we need  $O(\log n)$  time to update the convex hull and  $O(\log n)$  time to compute the  $SC$  vertices on each arc. The construction of  $\mathcal{F}$  from these items can then be done in time proportional to the size of  $\delta(\mathcal{F})$ , which is  $O(|\mathcal{A}|)$ .  $\square$

Furthermore,  $\mathcal{F}$  can be decomposed into generalized trapezoids with no overhead. A trapezoidal map is obtained by extending from each vertex of  $\mathcal{F}$  a vertical line and keeping the portion that is included in  $\mathcal{F}$  and contains the vertex. Such a vertical segment is called a wall. The vertical walls subdivide  $\mathcal{F}$  into generalized trapezoids, each one consisting of two walls (possibly one in some cases) a straight or circular floor and a straight or circular ceiling. See Figure 8 for the trapezoidal decomposition of the example of Figure 4. In the sequel, we will assume that such a map is available. In fact many algorithms computing arrangements compute this map as a preliminary first step.

In the sequel, we shall also use a similar construction for the whole arrangement  $\mathcal{A}$ . The only difference with the one just described is that we also erase walls from the points of the edges of  $\mathcal{A}$  with a vertical tangent.

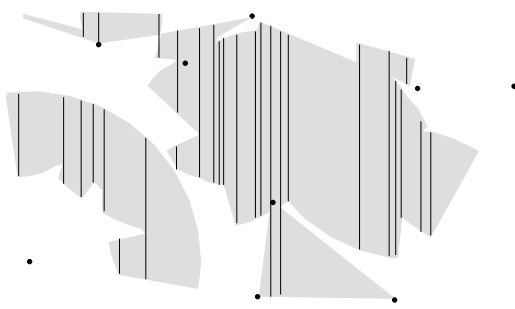


Fig. 8. The trapezoidal decomposition of  $\mathcal{F}$

## 5. Applications

### 5.1. Placements

**Question 1 :** *For a given configuration  $G$ , does there exist a feasible and stable placement ?*

To answer the query, we have to decide whether or not  $G$  lies in  $\mathcal{F}$ . The problem reduces to locating  $G$  in the trapezoidal map of  $\mathcal{F}$ . As a location query in a planar map of size  $m$  can be done in time  $O(\log m)$  after a preprocessing that requires  $O(m \log m)$  time and  $O(m)$  space,<sup>11</sup> we have

**Theorem 9** *There exists a data structure such that, for a given position of the body  $G$  of the robot, we can decide if there exists a stable placement in time  $O(\log n)$ . The data structure requires  $O(|\delta(\mathcal{F})|)$  space and can be constructed in  $O(|\delta(\mathcal{F})| \log n)$  additional time, once  $\mathcal{F}$  has been computed.*

**Question 2 :** *For a given feasible and stable configuration  $G$ , find a placement.*

We first determine the cell  $\Gamma$  of  $\mathcal{A}$  which contains  $G$ . Let  $S_\Gamma$  be the set of sites whose disks cover  $\Gamma$ ,  $CH(S_\Gamma)$  their convex hull and  $x$  any point of  $S_\Gamma$ . We determine the edge  $yz$  of  $CH(S_\Gamma)$  that is intersected by the ray issuing from  $x$  and passing through  $G$ :  $xyz$  is a feasible and stable placement.

In order to construct  $CH(S_\Gamma)$ , we use the data structure of Hershberger and Suri<sup>10</sup> already mentioned in Section 4. As the data structure allows us to perform binary search on  $CH(S_\Gamma)$  in  $O(\log n)$  time, we can find edge  $yz$  in logarithmic time.

**Theorem 10** *There exists a data structure such that, for a given feasible and stable configuration of the robot, we can compute a stable placement in time  $O(\log n)$ . The data structure requires  $O(|\mathcal{A}| \log n)$  space and can be constructed in  $O(|\mathcal{A}| \log n)$  time.*

### 5.2. Paths

**Question 3 :** *For two given stable configurations, does there exist a stable path joining them ?*

To answer the query, we have to decide whether or not the two configurations belong to the same connected component of  $\mathcal{F}$ . This can be done within the same

bounds as for Question 1, provided that each trapezoid is labelled by the connected component of  $\mathcal{F}$  it belongs to (this can be done in  $O(|\delta(\mathcal{F})|)$  time during the preprocessing step by a traversal of the trapezoidal map of  $\mathcal{F}$ ).

If one wants to construct the path, it suffices to look for a path inside  $\mathcal{F}$ , which can be done in time  $O(|\delta(\mathcal{F})|\log n)$  and space  $O(|\delta(\mathcal{F})|)$ .

More specifically, we search for a path in the adjacency graph of the trapezoidal map of  $\mathcal{F}$  using breadth-first search. This yields a sequence of adjacent trapezoids of  $\mathcal{F}$ , such that the first one contains  $S$ , the last one contains  $A$  and each trapezoid of  $\mathcal{F}$  appears at most once in the sequence and thus this sequence forms a simple polygon (whose edges are line segments or arcs of circle).

**Theorem 11** *There exists a data structure such that, given two configurations  $S$  and  $A$ , we can decide if there exists a stable path joining  $S$  and  $A$  in time  $O(\log n)$ . The data structure requires  $O(|\delta(\mathcal{F})|)$  space and can be constructed in  $O(|\delta(\mathcal{F})|\log n)$  additional time, once  $\mathcal{F}$  has been computed. Constructing such a path can be done within the same bounds as the preprocessing.*

A remaining problem is to find the sequence of placements along that path. This will be considered in the next section. Before that we give some additional useful results.

**Proposition 12** *There exists a stable path for the body of the robot intersecting the trapezoids of  $\mathcal{A}$   $O(k|\mathcal{A}|)$  times, if  $k$  is the density of the footholds.*

**Proof.** Each trapezoid of  $\mathcal{A}$  that is covered by  $k$  disks is decomposed into  $O(k)$  subcells by the convex hull of the  $k$  centers of the disks and thus into  $O(k)$  trapezoids of  $\mathcal{F}$ . Thus the path obtained from the trapezoidal decomposition of  $\mathcal{F}$ , as described above, intersects the trapezoidal decomposition of  $\mathcal{A}$  at most  $O(k|\mathcal{A}|)$  times.  $\square$

**Question 4 :** *For two given configurations belonging to the same connected component of  $\mathcal{F}$ , find the shortest stable path joining them.*

One may be interested in computing a shortest path for the body of the robot. We observe that a shortest path  $\mathcal{C}$  is a sequence of straight line-segments: indeed, the portions of  $\mathcal{C}$  inside  $\mathcal{F}$  are obviously line segments and, since each curved edge of  $\delta(\mathcal{F})$  is inner convex, only straight edges of  $\delta(\mathcal{F})$  can appear as parts of  $\mathcal{C}$ . As a consequence of this observation, we can compute a shortest path inside  $\mathcal{F}$  by means of any known algorithm for computing a shortest path inside a polygonal region, for example the ones based on the visibility graph. By the result of Welzl,<sup>12</sup>  $\mathcal{C}$  can thus be computed in  $O(|\delta(\mathcal{F})|^2)$  time.

**Theorem 13** *There exists a data structure such that, given two configurations  $S$  and  $A$  belonging to the same connected component of  $\mathcal{F}$ , we can compute the shortest stable path joining  $S$  and  $A$  in time  $O(|\delta(\mathcal{F})|^2)$ .*

Before we give the algorithm for the actual computation of the motion, we show that a shortest path  $\mathcal{C} \subset \mathcal{F}$ , between two stable and feasible configurations  $S$  and  $A$ , is a stable path with the property that the path intersects only  $O(|\mathcal{A}|)$  trapezoids of  $\mathcal{A}$  (this result is to be compared with Proposition 12).

**Proposition 14** *The intersection of a shortest path with any trapezoid of  $\mathcal{A}$  consists of at most three line segments.*

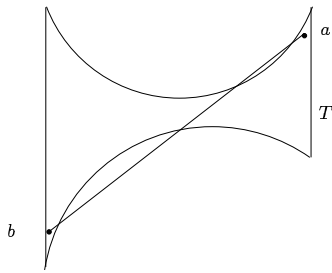


Fig. 9. Crossing of a trapezoid

**Proof.** Recall that each trapezoid of the trapezoidal decomposition of  $\mathcal{A}$  is bounded from above and from below by a circular arc and on the left and on the right by vertical walls (Figure 9).

Let  $\mathcal{C}$  be a shortest path and  $a$  and  $b$  be two points of  $\mathcal{C}$  belonging to the same trapezoid  $T$  of  $\mathcal{A}$ . We claim that the portion of  $\mathcal{C}$  between  $a$  and  $b$  is the line segment  $ab$ . Indeed, if  $P$  is the convex hull of the centers of the disks covering  $T$  and  $D$  is the intersection of the disks covering  $T$ ,  $ab \subset P$  and  $ab \subset D$  since  $P$  and  $D$  are convex. Thus  $ab \subset \mathcal{F}$  by convexity. The claim holds in particular when  $a$  is the point where  $\mathcal{C}$  enters for the first time  $T$  and  $b$  is the point where  $\mathcal{C}$  exits  $T$  for the last time. The claim then follows from the fact that the intersection of a segment and a trapezoid consists of at most three segments (see Figure 9).

□

**Proposition 15** *There exists a stable path for the body of the robot intersecting the trapezoids of  $\mathcal{A}$   $O(|\mathcal{A}|)$  times. This path can be computed in time  $O(k|\mathcal{A}|)$  if  $k$  is the density of the footholds.*

**Proof.** The path of Proposition 12 can be shortened. Each time a trapezoid of  $\mathcal{A}$  is visited for the second time, the short cut is taken. Thanks to Proposition 14, the shortened path is included in  $\mathcal{F}$ . The time needed is clearly proportional to the size of the first path that is  $O(k|\mathcal{A}|)$ . □

### 5.3. Motions

**Question 5 :** *Once a stable (possibly) shortest path  $\mathcal{C}$  has been found find the sequence of placements along  $\mathcal{C}$ .*

The motion planning problem is solved by computing the sequence of leg placements as the robot body  $G$  moves along  $\mathcal{C}$  in the trapezoidal map of  $\mathcal{A}$ . As four legs are always sufficient to find a stable motion if one exists, we assume in this paragraph that the robot has four legs. We will consider stable paths  $\mathcal{C}$  of the types described in Proposition 12 and 14.

We call *critical* a placement such that  $G$  cannot go further along  $\mathcal{C}$  with the same footholds. There are two types of critical placements. A critical placement of *Type 1* occurs when the stability condition does not hold any more because  $G$  crosses an edge of the current stability polygon (i.e. the convex hull of the footholds actually used by the robot). The placement of *Type 2* occurs when  $G$  comes out of a disk



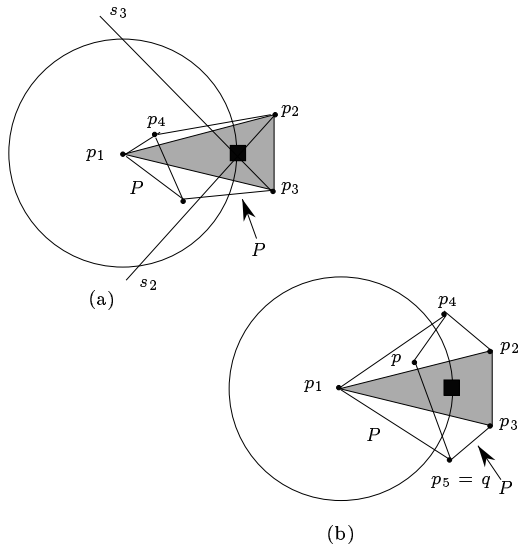


Fig. 11. Exit from a maximal extension placement

wedge  $W$  would contain  $q$ ),  $G$  must belong to the triangle  $pp_1p_3$ .

The robot moves as follows: first the free leg is moved to  $p$ , the leg in  $p_2$  is placed on  $q$ , and then the back leg (resting in  $p_1$ ) is lifted. The motion can then proceed until a new critical placement occurs.

We have shown how the robot can traverse a critical placement along the path  $\mathcal{C}$ : the complexity of the total traversal is given by the next two theorems.

**Theorem 16** *A robot motion joining two given placements can be computed in time proportional to  $|\mathcal{A}|(k + \log n)$  that is  $O(k^2n + kn \log n)$  in the worst case, if  $k$  is the density of the footholds. The motion consists of  $O(|\mathcal{A}| = kn)$  leg moves.*

**Theorem 17** *A robot motion along a shortest path joining two given placements can be computed in  $O(|\mathcal{F}|^2 + |\mathcal{A}| \log n)$  time. The motion consists of  $O(|\mathcal{A}|)$  leg moves.*

**Proofs.** For each critical placement we can find the new placement along  $\mathcal{C}$  in time  $O(\log n)$  using the off-line maintenance algorithm of convex hulls. In addition we have to compute dynamically the new convex hull at each intersection between  $\mathcal{C}$  and an arc of  $\mathcal{A}$ , which can be done in  $O(\log n)$  time for each intersection using an off-line dynamic algorithm.<sup>10</sup> Since, by Proposition 15 (resp. 14)  $\mathcal{C}$  crosses a given trapezoid of  $\mathcal{A}$  only a bounded number of times, the convex hull maintenance is invoked at most  $O(|\mathcal{A}|)$  times. Lastly, we have to know how many critical placements occur along  $\mathcal{C}$ . Critical placements of Type 2 occur at most each time the trajectory crosses an arc, that is  $O(|\mathcal{A}|)$  times. If we are at a critical placement of Type 1, we have shown above that after at most two new placements we have to cross an arc of  $\mathcal{A}$ : thus the same bound applies to this case.

Our placement strategy gives an upper bound on the number of leg moves: we have seen that for crossing each trapezoid we need at most two leg changes. Thus globally the robot performs at most  $O(|\mathcal{A}|)$  leg moves.  $\square$

## 6. Concluding Remarks

$|\mathcal{F}|$  is usually much smaller than  $|\mathcal{A}|$ , even when  $\mathcal{A}$  has quadratic size. It would be interesting to have an algorithm that constructs  $\mathcal{F}$  in time proportional to the actual size of  $\mathcal{F}$ , preferably to the size of  $\mathcal{A}$ .

A natural extension of the present work is to study other legged robots, for instance robots whose legs are attached at different points of the (no longer point like) body. For a given orientation of the robot, we can extend some of our results. We associate to each leg a color, and color the footholds that a given leg can reach with the color of the leg (a foothold may have several colors). The definition of the stability region (Eq. (3)) must be modified : the role of the convex hull  $CH(S)$  is now played by the union of the triangles whose vertices have distinct colors. It can be shown that such a union has linear size and can be computed in optimal  $\Theta(n \log n)$  time.<sup>13</sup> It follows that  $\mathcal{F}$  has size  $O(n|\mathcal{A}|)$  and can be computed in  $O(n|\mathcal{A}| \log n)$  time. We do not know if these bounds are tight.

Further extensions under consideration will allow these robots to rotate, thus adding a third degree of freedom.

It would also be interesting to consider a more general setting involving 3-D space, obstacles, and more complex footholds.

## Acknowledgement

The authors would like to thank Jean-Pierre Merlet for supplying them with his interactive drawing preparation system JDraw .

## References

1. Special issue on legged locomotion. *International Journal of Robotics Research*, 9(2), April 1990.
2. Special issue on legged locomotion. *International Journal of Robotics Research*, 3(2), 1984.
3. S. Hirose, M. Nose, H. Kikuchi, and Y. Umetani. Adaptive gait control of a quadruped walking vehicle. In *Int. Symp. on Robotics Research*, pages 253–277. MIT Press, 1984.
4. S. Hirose and O. Kunieda. Generalized standard foot trajectory for a quadruped walking vehicle. *The International Journal of Robotics Research*, 10(1), February 1991.
5. J. Bares, M. Hebert, T. Kanade, E. Krotkov, T. Mitchell, R. Simmons, and W. Whitaker. Ambler, an autonomous rover for planetary exploration. *Computer*, pages 18–26, June 1989.
6. F. P. Preparata and M. I. Shamos. *Computational Geometry: an Introduction*. Springer-Verlag, New York, NY, 1985.
7. D. T. Lee. On  $k$ -nearest neighbor Voronoi diagrams in the plane. *IEEE Trans. Comput.*, C-31:478–487, 1982.
8. A. Aggarwal, L. J. Guibas, J. Saxe, and P. Shor. A linear time algorithm for computing the Voronoi diagram of a convex polygon. In *Proc. 19th Annu. ACM Sympos. Theory Comput.*, pages 39–45, 1987.

9. M. Sharir. On  $k$ -sets in arrangements of curves and surfaces. *Discrete Comput. Geom.*, 6:593–613, 1991.
10. J. Hershberger and S. Suri. Offline maintenance of planar configurations. In *Proc. 2nd ACM-SIAM Sympos. Discrete Algorithms*, pages 32–41, 1991.
11. F. P. Preparata. Planar point location revisited. *Internat. J. Found. Comput. Sci.*, 1:71–86, 1990.
12. E. Welzl. Constructing the visibility graph for  $n$  line segments in  $O(n^2)$  time. *Inform. Process. Lett.*, 20:167–171, 1985.
13. J.-D. Boissonnat, O. Devillers, and F. Preparata. Computing the union of 3-colored triangles. *Internat. J. Comput. Geom. Appl.*, 1(2):187–196, 1991.