

On the Parameterization and the Geometry of the Configuration Space of a Single Planar Robot

Dror Atariah

Institut für Informatik
Freie Universität Berlin
Berlin, Germany

dror.atariah@fu-berlin.de

Sunayana Ghosh

Institut für Informatik
Freie Universität Berlin
Berlin, Germany

sghosh@inf.fu-berlin.de

Günter Rote

Institut für Informatik
Freie Universität Berlin
Berlin, Germany

rote@inf.fu-berlin.de

ABSTRACT

Translating and rotating planar polygonal robots are studied in the literature for decades. An integral part of this study is the configuration space which corresponds to the work space. In the context of motion planning problems, the boundary between the free and forbidden parts of the configuration space plays a major role. In this paper we find an explicit parameterization of the boundary of the forbidden space. Using this parameterization we detail several geometrical properties of the various elements which constitute this boundary. In addition, this parameterization enables us to visualize these elements.

Keywords

Robotics, Motion planning, Configuration Space, Parameterizations

1 INTRODUCTION

The *piano movers problem* is about four decades old [SS83, IKP73] and studied intensively ever since. A fundamental part of this study is the configuration space which is associated to the work space at hand. A work space which consists of a planar polygonal convex robot, which is free to rotate and translate and polygonal obstacles give rise to a *configuration space*. Each point in the configuration space corresponds to a unique placement or pose of the robot in its work space, and vice versa, that is, every pose of the robot in the work space corresponds to a unique configuration point. The presence of obstacles in the work space translates to the partition of the configuration space into two parts, namely the *free* and *forbidden* spaces. Configuration points in the forbidden part correspond to poses in which the interior of the robot intersects the interior of one or more obstacles.

Most studies set the solution of the *motion planning problem* as the primary goal and thus focus mainly on algorithmical aspects. Thus, the related configuration space was hardly studied from a *geometrical point of view*. In this paper we focus on the geometrical properties of the configuration space which is associated to

the piano movers problem. To that end, we derive in Section 3 an explicit parameterization of the boundary of the forbidden space. Better understanding of this boundary can contribute, for example, to the general study of the motion planning problem. In turn, using this parameterization we study in Section 4 the geometrical properties of this boundary.

In terms of visualization, most of the illustrations of the configuration space that can be found in the literature are rather simple. It is well known that for a robot which can *only* translate, the boundary of the forbidden space is polygonal and can be computed using *Minkowski sums*. Thus, most of the visualizations slice the configuration space with horizontal planes. Each slice corresponds to a fixed rotation of the robot and the boundary can be computed using Minkowski sums. Finally, stacking these slices yields a discrete visualization of the obstacles as they appear in the configuration space [Lat93]. Using our parameterization it is easy to visualize the boundary of the forbidden space, as can be seen in Figure 9 and in [AR12].

Previous Work.

The work of Lozano-Pérez and Wesley [LPW79] put the so-called *configuration space* under the spotlight. Surveys like [WB00, HA92] provide a broad overview at least on the early study of this fundamental concept. As we already pointed out, the literature aims mainly at the motion planning problem and hardly considers the boundary of the forbidden space per se, let alone parameterizing it.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

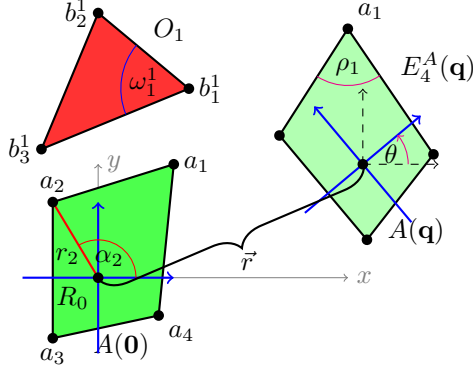


Figure 1: An example of a work space with a robot $A(\mathbf{0})$ in its rest position (dark green) and in a configuration resulting from a translation by a vector \vec{r} and rotation in angle θ (light green), that is $A(\mathbf{q})$ where $\mathbf{q} = (\vec{r}, \theta)$. Local frame is in solid blue and one obstacle O_1 is in red.

Two interesting examples are [BA88, MST13]. Both attempt to solve the motion planning problem itself, although they provide, as a byproduct, some idea on the geometrical nature of the boundary of the forbidden space. Yet, neither of them provides an explicit, simple and concise representation of the various elements of the boundary.

1.1 Definitions and Notations

In this section we describe all the notations and notions that will be used throughout the paper. Further details can be found in standard textbooks like [LaV06, Lat93, CLH⁺05, Lau98]. In addition refer to Figure 1 for illustrations of the various definitions.

The Robot.

A robot A is a convex planar polygon with n vertices denoted by $\{a_i\}_{i=1}^n$; we assume that they are given in counterclockwise order. The robot can translate and rotate in a *work space* scattered with polygonal (convex) obstacles. The work space is denoted by \mathcal{W} and we take it to be $\mathcal{W} = \mathbb{R}^2$. The *reference point* of the robot is denoted by R_0 , and we assume that in the *rest position* it is at the origin. Furthermore, we assume that the *local frame* of the robot aligns with the coordinates of the work space when in the rest position.

The vertices of the robot are either given with Cartesian coordinates, or with polar coordinates. For the vertex a_i we use the following notation $a_i = r_i(\cos \alpha_i, \sin \alpha_i)$ where $r_i = \|a_i\|$ and α_i denotes the angle with respect to the local frame of the robot. We assume that the vertices are in an increasing angular order, that is $0 \leq \alpha_1 \leq \dots \leq \alpha_n < 2\pi$. Finally, for the sake of simplicity and consistency, we assume that the reference point lies in

the interior or on the boundary of the robot. We denote by E_i^A the edge $\overline{a_i a_{i+1}}$ which connects a_i with a_{i+1} . Lastly, we denote by ρ_i the internal angle corresponding to the i -th vertex.

Obstacle(s).

Let $\{O_k\}_{k=1}^m$ be m obstacles in the work space. The *vertices* of O_k for some k are given in counterclockwise order, and are denoted by $\{b_j^k\}$. Analogously to the notations we use for the robot, $E_j^{O_k}$ will denote the edge which connects b_j^k and b_{j+1}^k . Finally, the interior angle at the j -th vertex will be denoted by ω_j^k . If the context introduces no confusion then we shall omit the index k .

Configuration Space and Poses.

We let \mathcal{C} denote the *configuration space* of the robot A in the work space \mathcal{W} which is scattered with the obstacles $\{O_k\}$. The *free* and *forbidden* part of \mathcal{C} are denoted by $\mathcal{C}_{\text{free}}$ and $\mathcal{C}_{\text{forb}}$ respectively. An element $\mathbf{q} \in \mathcal{C}$ is called a *configuration point*, or *configuration* for short. Given a configuration $\mathbf{q} \in \mathcal{C}$ we denote by $A(\mathbf{q})$ the portion of the work space which is covered by A when it assumes the configuration \mathbf{q} and it is called either *placement*, or *pose*, or simply *configuration*, when there is no risk of confusion. Similarly $R_0(\mathbf{q})$, $a_i(\mathbf{q})$ and $E_i^A(\mathbf{q})$ denote respectively the position, in the work space, of the reference point, i -th vertex or i -th edge of the robot. In particular, for a configuration point $\mathbf{q} = (\vec{r}, \theta)$ with translation component \vec{r} and rotation component θ , we have

$$R_0(\mathbf{q}) = \vec{r} + R_0,$$

$$E_i^A(\mathbf{q}) = \overline{a_{i+1}(\mathbf{q}) a_i(\mathbf{q})}.$$

In order to express $a_i(\mathbf{q})$ for an arbitrary configuration \mathbf{q} , we have to choose a model of the configuration space. To that end, we consider two possible models

$$\mathcal{C}^{\text{geom}} = \{(x, y, \theta) \mid (x, y) \in \mathbb{R}^2, \theta \in [0, 2\pi)\} \quad (1)$$

$$\mathcal{C}^{\text{rat}} = \{(x, y, \tau) \mid (x, y) \in \mathbb{R}^2, \tau \in \mathbb{R} \cup \infty\} \quad (2)$$

which we call the *geometrical* and *rational* models respectively. Note that

$$\mathcal{C}^{\text{geom}} = \mathbb{R}^2 \times S^1 \text{ and } \mathcal{C}^{\text{rat}} = \mathbb{R}^2 \times \mathbb{RP}^1.$$

These models are related by $\tau = \tan \frac{\theta}{2}$. In particular, for $\mathbf{q} = (\vec{r}, \theta) \in \mathcal{C}^{\text{geom}}$ we have

$$a_i(\mathbf{q}) = \vec{r} + R^\theta a_i \quad (3)$$

with R^θ denoting the standard *rotation matrix* and where a_i is the i -th vertex of A in the rest position.

Similarly, for $\mathbf{q}' = (\vec{r}, \tau) \in \mathcal{C}^{\text{rat}}$ we have

$$a_i(\mathbf{q}') = \vec{r} + M^\tau a_i$$

with the so-called *rational rotation matrix*

$$M^\tau = \frac{1}{1+\tau^2} \begin{pmatrix} 1-\tau^2 & -2\tau \\ 2\tau & 1-\tau^2 \end{pmatrix}.$$

Note that since $\lim_{\tau \rightarrow \infty} M^\tau = \lim_{\tau \rightarrow -\infty} M^\tau = R^\pi$ we can safely set $M^\infty = R^\pi$.

When using the rational model of the configuration space and taking rational coordinates for the translation vector \vec{r} and letting $\tau \in \mathbb{Q}$, it is possible to establish *exact* computations of placements. On the other hand, the geometrical representation is of more use when one is trying to visualize elements of the configuration space.

Remark 1. The configuration space in our case, namely the one corresponding to a planar robot that is free to rotate and translate (this kind of robot is also called *holonomic*), is homeomorphic to the *special Euclidean group* $SE(2)$. Indeed, the following homeomorphisms hold

$$SE(2) \cong \mathbb{R}^2 \times S^1 \cong \mathbb{R}^2 \times \mathbb{RP}^1.$$

See [LaV06, §4.2] for further details.

1.2 Contacts and the Boundary of the Forbidden Space

We say that $A(\mathbf{q})$ *touches* or is *in contact* with an obstacle O for a configuration \mathbf{q} if

$$\partial A(\mathbf{q}) \cap \partial O \neq \emptyset \wedge \text{int}(A(\mathbf{q})) \cap \text{int}O = \emptyset.$$

If only

$$\partial A(\mathbf{q}) \cap \partial O \neq \emptyset,$$

then we say that $A(\mathbf{q})$ *pseudo touches* or is *in pseudo contact* with the obstacle O . For a configuration \mathbf{q} , such that A pseudo touches *or* just touches an obstacle O , one or more of the following *contact types* can hold:

Name	Notation	Definition
Vertex-Edge	(v_i-e_j)	$a_i(\mathbf{q}) \cap \text{int}E_j^O \neq \emptyset$
Edge-Vertex	(e_i-v_j)	$\text{int}E_i^A(\mathbf{q}) \cap b_j \neq \emptyset$
Vertex-Vertex	(v_i-v_j)	$a_i(\mathbf{q}) = b_j$
Edge-Edge	(e_i-e_j)	$ \text{int}E_i^A(\mathbf{q}) \cap \text{int}E_j^O > 1$

Note that the contact type alone does not imply whether the interiors of the robot and of the obstacle intersect or not. Note, in addition, that a robot can maintain various pseudo contacts and contacts with the same obstacle simultaneously (cf. O_3 in Figure 2). In the presence of more than one obstacle in \mathcal{W} the robot can maintain multiple contacts as well. The following definitions refer to portions of $\mathcal{C}_{\text{forb}}$ which maintain a fixed contact type.

Definition 1 (Contact Surface). *The set of all configuration points that correspond to a pseudo contact between a fixed vertex (or an edge) of the robot and a fixed vertex (or an edge) of an obstacle is called a contact surface.*

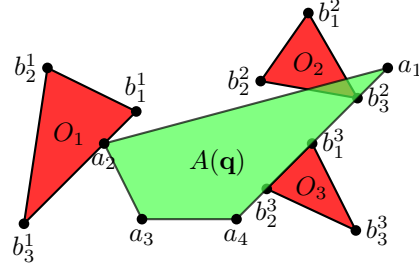


Figure 2: In this example, the configuration \mathbf{q} corresponds to a (v_2-e_3) contact with O_1 and a (e_4-v_3) pseudo contact with O_2 . Finally, $A(\mathbf{q})$ maintains two $(v-v)$ contacts and one $(e-e)$ contact with O_3 simultaneously. This means that \mathbf{q} belongs to four different contact patches and to one contact surface.

Note that a contact surface is a subset of $\mathcal{C}_{\text{forb}}$, since a configuration which realizes a pseudo contact may realize at the same time an intersection of the interiors of the robot and an obstacle. The following definition focuses on the configurations which realize *contacts*.

Definition 2 (Contact Patch). *The set of all configuration points that correspond to a contact between a fixed vertex (or an edge) of the robot and a fixed vertex (or an edge) of an obstacle is called a contact patch.*

Every contact between a robot and an obstacle is also a pseudo contact, thus each contact patch is a subset of a contact surface. Furthermore, it is a subset of $\partial \mathcal{C}_{\text{forb}}$. In addition, the union of all contact patches is the boundary of the forbidden space. Finally, a contact surface which maintains either a $(v-e)$ or $(e-v)$ contact type is of dimension two, whereas a contact surface which maintains either a $(v-v)$ or $(e-e)$ contact type is of dimension one. This means that the boundary $\partial \mathcal{C}_{\text{forb}}$ is a union of contact patches of dimension two which are “glued” together with contact patches of dimension one. Figure 2 illustrates the notions discussed in this section.

In this paper we will formulate an explicit parameterization of the contact surfaces depending on the properties of the robot and the obstacles. Furthermore, we will find a subset of the parameter domain, of each contact surface, which corresponds to the respective contact patch. Thus, we will be able to parameterize the whole boundary of the forbidden space.

2 ROTATING THE ROBOT

Since the robot A that we consider is holonomic, every point $P \in \mathcal{W}$ can be a center of rotation of the robot. In particular, given a configuration point $\mathbf{q} \in \mathcal{C}$, the robot can rotate about every boundary point $\partial A(\mathbf{q}) \in \mathcal{W}$. This kind of motion is the corner stone of the parameterization that we formalize in this paper.

Let us set a point $P \in \mathcal{W}$ and mark a point $a \in \partial A(\mathbf{0})$ on the boundary of the robot. According to the notations that we use, $a(\mathbf{0}), a(\mathbf{q})$ denote the position of the

marked point when the robot is in either the rest position or in some pose corresponding to a configuration \mathbf{q} . We will parameterize the set of configuration points in which the marked point a is fixed to the point P . More precisely, we want to parameterize the following set

$$P_a = \{\mathbf{q} \in \mathcal{C} : a(\mathbf{q}) = P\}.$$

Lemma 1. *Given a point $P \in \mathcal{W}$ and a point $a \in \partial A$ the set P_a is parameterized by*

$$\mathbf{q}_a(\phi) = \begin{pmatrix} \vec{r}_a(\phi) \\ \theta_a(\phi) \end{pmatrix} = \begin{pmatrix} P - R^\phi a \\ \phi \end{pmatrix} \quad (4)$$

for $\phi \in [0, 2\pi)$. That is, $\mathbf{q} \in P_a$ if and only if $\mathbf{q} = \mathbf{q}_a(\phi)$ for some $\phi \in [0, 2\pi)$.

Proof. Since $a \in E_i^A$ for some index i , we can write $a = (1-t)a_i + ta_{i+1}$ for some $t \in [0, 1)$. First we show that if $\mathbf{q} = \mathbf{q}_a(\phi)$ for some ϕ then $\mathbf{q} \in P_a$, that is $a(\mathbf{q}) = P$. For every ϕ , using Equation (3), we have

$$\left. \begin{aligned} a_i(\mathbf{q}_a(\phi)) &= P - R^\phi a + R^\phi a_i \\ a_{i+1}(\mathbf{q}_a(\phi)) &= P - R^\phi a + R^\phi a_{i+1} \end{aligned} \right\}.$$

It is easy to show that

$$a(\mathbf{q}_a(\phi)) = (1-t)a_i(\mathbf{q}_a(\phi)) + ta_{i+1}(\mathbf{q}_a(\phi)) = P.$$

That is, for every ϕ , the point $a(\mathbf{q}_a(\phi))$ is fixed to P .

Conversely, given $\mathbf{q} = (\vec{r}, \theta) \in P_a$ we have

$$P = a(\mathbf{q}) = \vec{r} + R^\theta a.$$

Thus, $\vec{r} = P - R^\theta a$. Finally, for $\phi = \theta$ we have that $\mathbf{q} = \mathbf{q}_a(\phi)$. \square

We observe that for $\phi = 0$ the parameterization given in Equation (4) is merely a translation. That is, the local frame which is assigned to the robot for $\mathbf{q}_a(0)$ is aligned with the global frame of the work space. At this point it is important to point out that $\mathbf{q}_a(\phi)$ is meaningful only when interpreted as a point in $\mathcal{C}^{\text{geom}}$. Finally, Equation (4) is a parameterization of a *helix* in the configuration space. We conclude this section with the following remark.

Remark 2 (Covering \mathcal{C} -Space with Helices). Instead of taking $a \in \partial A$, we can generalize the idea and consider an arbitrary linear combination of the vertices of the robot, $a = \sum_{i=1}^n \lambda_i a_i$, and some point $P \in \mathcal{W}$. The set of configurations which correspond to a rotation of the robot such that a is fixed to P is again a helix. As a matter of fact, every configuration point $\mathbf{q} \in \mathcal{C}$ is contained in infinitely many helices of this form.

3 PARAMETERIZING CONTACT SURFACES

In this section we consider the robot A and *one convex* obstacle O . Later, an arbitrary obstacle can be decomposed into convex subsets and each sub-obstacle can be treated in a similar way. Given a contact type of A and O we will derive an explicit parameterization of the corresponding contact surface and patch.

3.1 Vertex-Edge Contact

A (v_i-e_j) vertex-edge pseudo contact occurs when a vertex a_i of the robot lies in the interior of an edge E_j^O of the obstacle (see O_1 and $A(\mathbf{q})$ in Figure 2 for an example). In this section, we utilize the parameterization obtained in Section 2, and provide an explicit parameterization of the contact surface and the contact patch in the configuration space corresponding to the prescribed (v_i-e_j) pseudo contact and contact respectively.

Let $P(t) = (1-t)b_j + tb_{j+1}$ be an arbitrary point in the interior of E_j^O . The configurations that correspond to the (v_i-e_j) contact can be derived from Equation (4) by replacing P with $P(t)$ and a with a_i and is given by

$$\begin{aligned} S(t, \phi) &= \begin{pmatrix} P(t) - R^\phi a_i \\ \phi \end{pmatrix} \\ &= c(\phi) + t\vec{r}(\phi), \end{aligned} \quad (5)$$

for $t \in (0, 1)$ and $\phi \in [0, 2\pi)$. As ϕ varies in the interval $[0, 2\pi)$, the configuration points on S represent both contacts and pseudo contacts. Clearly, this surface is a *ruled surface* with directrix $c(\phi)$ and $\vec{r}(\phi) \neq 0$ as the vector field. Note that $\frac{d}{d\phi}\vec{r}(\phi) = 0$ which implies that S is a *cylindrical ruled surface* and thus *developable* [dC76]. It is easy to verify that $S(t, \phi)$ is a collection of congruent helices. Note that for $t \in \{0, 1\}$ the parameterization reduces to two helices which correspond to the two pseudo contacts (v_i-v_j) and (v_i-v_{j+1}) respectively (cf. Section 3.3). In Figure 7, such a surface is illustrated with helical arcs in black and rulings in yellow.

Remark 3. If we fix a vertex a_i of the robot and generate all possible vertex-edge contact surfaces with all edges of the obstacle, then *helices* contained in each of these contact surfaces are congruent copies of each other. Note that if the obstacles are regular polygons then for a fixed vertex of the robot the contact surfaces are just congruent copies of each other. If, in addition, the robot is a regular polygon then *all the vertex-edge contact surfaces* are congruent copies of each other.

Our next goal is to find a sub-domain $\Phi \subset [0, 2\pi)$ such that $S(t, \phi)|_{\phi \in \Phi}$ will be the *contact patch* which is contained in S . In Section 3.1.1 we analyze the domain $[0, 2\pi)$ of ϕ and find this sub-domain Φ .

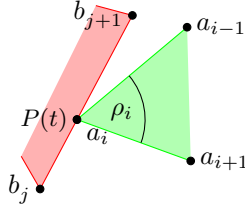


Figure 3: Geometrical components of the (v_i-e_j) contact.

3.1.1 Vertex-Edge Angle Range Analysis

By now, given a vertex a_i of the robot and an edge E_j^O of an obstacle O , we have an explicit parameterization of the contact surface S in the configuration space which corresponds to the (v_i-e_j) pseudo contact. Our goal is to find a contact patch $S' \subset S \in \mathcal{C}$, such that for all $\mathbf{q} \in S'$ we will have that $a_i(\mathbf{q})$ touches the edge of the obstacle.

Since we assume that both the robot and the obstacle are convex, the sub-domain Φ is independent of t , namely independent of the point of contact along E_j^O . This can be seen in Figures 2 and 3. For some fixed $t_0 \in (0, 1)$ let $\mathbf{q}_i(\phi) = S(t_0, \phi)$ be a helix in S which corresponds to the pseudo contact between a_i and $P = P(t_0)$. The sub-domain Φ can be determined by finding two values:

- ϕ_{\min} : The minimal angle for which $a_{i+1}(\mathbf{q}_i(\phi_{\min}))$ lies on the line containing the edge E_j^O and simultaneously $a_{i-1}(\mathbf{q}_i(\phi_{\min}))$ lies to the right of this edge.
- ϕ_{range} : The range of rotation that maintains the contact of a_i with P . In practice this means that we want $a_{i-1}(\mathbf{q}_i(\phi_{\min} + \phi_{\text{range}}))$ to lie on the line containing E_j^O such that $a_{i+1}(\mathbf{q}_i(\phi_{\min} + \phi_{\text{range}}))$ will lie to its right. See Figure 3 for an illustration.

If we let $\phi_{\max} = \phi_{\min} + \phi_{\text{range}}$, then

$$\Phi = \begin{cases} [\phi_{\min}, \phi_{\max}] & \text{if } \phi_{\max} < 2\pi \\ [\phi_{\min}, 2\pi] \cup [0, \phi_{\max} - 2\pi] & \text{if } \phi_{\max} \geq 2\pi \end{cases} \quad (6)$$

is the sub-domain we want to find.

We now compute the values of ϕ_{\min} and ϕ_{range} . The latter is straightforward to find, and depends on the interior angle at the vertex a_i of A , namely

$$\phi_{\text{range}} = \pi - \rho_i.$$

Computing ϕ_{\min} .

We want to find ϕ such that for $\mathbf{q}_i(\phi) \in \mathcal{C}$ the following will hold:

$$P - \|E_i^A\| \frac{E_j^O}{\|E_j^O\|} = a_{i+1}(\mathbf{q}_i(\phi)),$$

x_0	y_0	$\phi_{\min} \in$
≥ 0	≥ 0	$[0, \frac{\pi}{2}]$
< 0	≥ 0	$[\frac{\pi}{2}, \pi]$
< 0	< 0	$[\pi, \frac{3\pi}{2}]$
≥ 0	< 0	$[\frac{3\pi}{2}, 2\pi]$

Table 1: Interval of ϕ_{\min} depending on signs of x_0 and y_0 for the vertex-edge and edge-vertex contact types.

where E_j^O is consider as the vector from b_j to b_{j+1} and $\|\cdot\|$ denotes the length of an edge. Solving this equation for ϕ is equivalent to solving

$$-\|E_i^A\| \frac{E_j^O}{\|E_j^O\|} = M \cdot (x, y)^T,$$

where $x = \cos \phi$, $y = \sin \phi$ and

$$M = \left(E_i^A, R^{\frac{\pi}{2}} \cdot E_i^A \right)^T. \quad (7)$$

Since $\det M = \|E_i^A\|^2 \neq 0$, this system has a unique solution, denoted by $(x_0, y_0)^T$. We define $\{\phi_i\}_{i=1}^4$ as follows

$$\begin{aligned} \{\phi_1, \phi_2\} &= \arccos(x_0) \cap [0, 2\pi) \\ \{\phi_3, \phi_4\} &= \arcsin(y_0) \cap [0, 2\pi) \end{aligned}$$

Note that since (x_0, y_0) is a unit vector, we have that $\{\phi_1, \phi_2\} \cap \{\phi_3, \phi_4\}$ contains exactly one element. As ϕ_{\min} should lie in $[0, 2\pi)$, it satisfies

$$\phi_{\min} = \{\phi_1, \phi_2\} \cap \{\phi_3, \phi_4\}.$$

For any combination of signs of x_0 and y_0 Table 1 suggests in which interval ϕ_{\min} is, and using the definition of the ϕ_i 's it can be easily found.

Finally, in Figure 9 we plot an example of all possible contact patches which correspond to a triangular robot and obstacle. The red patches are the vertex-edge contact patches. Let us conclude this section with one remark.

Remark 4 (On the exactness of computations). The steps that we described so far, in general, cannot yield the exact value of ϕ_{\min} since one has to compute the inverse functions of both sine and cosine. Furthermore, the matrix M in Equation (7) involves the trigonometric functions as well, and thus cannot be represented in an exact manner. In turn, this means that x_0 and y_0 above cannot be computed exactly in the first place. If the vertices of the robot are assumed to lie on a circle of some fixed radius, then it is possible to find Φ without trigonometric functions; further details can be found in [AGR13].

3.2 Edge-Vertex Contact

Recall that Equation (4) parameterizes a rotation of the robot about a point P such that a boundary point $a \in \partial A$ is fixed to P . For any $t \in (0, 1)$ we denote

$$a_{i,t} = (1-t)a_i + ta_{i+1}$$

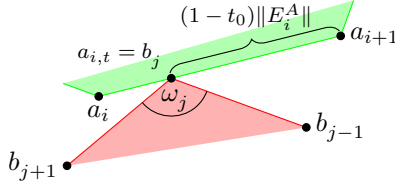


Figure 4: Geometrical components of the $(e-v_j)$ contact.

to be the point on the edge E_i^A of the robot which will be in pseudo contact with the vertex b_j of the obstacle. Next, in Equation (4), we replace P with b_j and a with $a_{i,t}$ and obtain

$$\begin{aligned} S(t, \phi) &= \begin{pmatrix} b_j - R^\phi a_{i,t} \\ \phi \end{pmatrix} \\ &= c(\phi) + t\vec{r}(\phi) \end{aligned} \quad (8)$$

for $t \in (0, 1)$ and $\phi \in [0, 2\pi)$. $S(t, \phi)$ is the contact surface corresponding to the $(e-v_j)$ pseudo contact. Again, like the parameterization in Equation (5) we obtain a ruled surface, obviously with different directrix and vector field. This surface is swept by a horizontal line segment which translates and rotates in \mathcal{C} . In Figure 8 an example of a typical edge-vertex contact surface is plotted with helices in black and rulings in yellow.

Remark 5. In contrast to the case of $(v-e)$ contact surfaces, here in the edge-vertex case, each contact surface is a collection of helices which are *not* congruent since their radii depend on the varying $a_{i,t}$. This suggests that $(e-v)$ contact surfaces are *not* developable. We establish this fact and study the geometry of these surfaces in Section 4.

In order to find the contact *patch* which is contained in S , as before, we have to find a sub-domain $\Phi \subset [0, 2\pi)$ for which $S(t, \phi)_{|\phi \in \Phi}$ is a collection of configuration points which correspond to contacts and *not* to pseudo contacts. Again, as can be seen in Figure 4, the sub-domain Φ does not depend on t .

3.2.1 Edge-Vertex Angle Range Analysis

In this section, similarly to the procedure discussed in Section 3.1.1, we find ϕ_{\min} , ϕ_{range} and ϕ_{\max} , where $\phi_{\max} = \phi_{\min} + \phi_{\text{range}}$, such that for Φ as defined in Equation (6) the sub-surface $S(t, \phi)_{|\phi \in \Phi}$ is a *contact patch*. As ϕ_{\min} and ϕ_{range} depend only on the indices i and j we shall fix some $t_0 \in (0, 1)$ and let $\mathbf{q}_i(\phi) = S(t_0, \phi)$. In this case, $a_{i+1}(\mathbf{q}_i(\phi_{\min}))$ has to lie on the line containing E_{j-1}^O , such that the interiors of the robot and the obstacle do not intersect. Similarly, $a_i(\mathbf{q}_i(\phi_{\max}))$ has to lie on the line segment containing E_j^O (cf. Figure 4). Clearly, we have that

$$\phi_{\text{range}} = \pi - \omega_j.$$

It is left to find the value of ϕ_{\min} .

Computing ϕ_{\min} .

In the case of $(e-v)$ contact, the minimal angle of rotation ϕ_{\min} is the one for which the following will hold

$$a_{i+1}(\mathbf{q}_i(\phi)) - b_j \parallel b_j - b_{j-1}. \quad (9)$$

The condition in (9) together with the restriction that a_{i,t_0} has to coincide with b_j can be formulated as follows:

$$b_j - (1 - t_0) \|E_i^A\| \frac{E_{j-1}^O}{\|E_{j-1}^O\|} = a_{i+1}(\mathbf{q}_i(\phi)),$$

with ϕ as the unknown. Computations similar to those we described in Section 3.1.1 can be applied in this case as well. This last equation can be rewritten as follows

$$(t_0 - 1) \|E_i^A\| \frac{E_{j-1}^O}{\|E_{j-1}^O\|} = M \cdot (x, y)^T$$

with $x = \cos \phi$, $y = \sin \phi$ and

$$M = \left(a_{i+1} - a_{i,t_0}, R^{\frac{\pi}{2}} \cdot (a_{i+1} - a_{i,t_0}) \right)^T.$$

As before, this system has a unique solution denoted by $(x_0, y_0)^T$. Based on the cases given in Table 1 we can find a unique solution ϕ_{\min} .

We conclude the section by referring to the green patches in Figure 9 which correspond to $(e-v)$ contacts.

3.3 Vertex-Vertex and Edge-Edge Contacts

In the previous sections we found explicit parameterizations of the contact surfaces and contact patches of dimension two, namely those that correspond to either vertex-edge or edge-vertex pseudo contacts and contacts. In order to complete the picture we have to consider the configurations that correspond to vertex-vertex and edge-edge pseudo contacts and contacts. To that end, we recall that the boundaries of the two dimensional contact surfaces that we have derived are exactly the one dimensional contact surfaces.

In the example plotted in Figure 9 blue helical arcs correspond to vertex-vertex contacts and the yellow (straight) lines correspond to edge-edge contacts.

3.3.1 Vertex-Vertex Contact

For two indices i and j , the contact surface which corresponds to the (v_i-v_j) pseudo contact is parameterized by

$$C(\phi) = S(0, \phi),$$

for $\phi \in [0, 2\pi)$ and $S(\cdot, \cdot)$ as given in Equation (5). In order to find the sub-domain Φ , which corresponds to the (v_i-v_j) contacts alone, we have to compute ϕ_{\min}

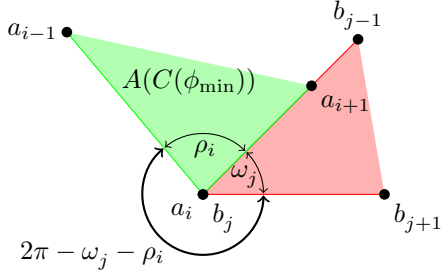


Figure 5: Illustration of a vertex-vertex contact with $C(\cdot)$ and ϕ_{\min} as defined in Section 3.3.1.

that corresponds to the (v_i-e_{j-1}) contact. For the pose which corresponds to $C(\phi_{\min})$ we have that a_i coincides with b_j , a_{i+1} lies on the line containing E_{j-1}^O and a_{i-1} lies to the right of this edge. See Figure 5 for an illustration. Letting ϕ vary in the sub-domain $\Phi = [\phi_{\min}, \phi_{\min} + 2\pi - \omega_j - \rho_i]$ yields the helical sub-arc of $C(\phi)$ which maintain a (v_i-v_j) contact. As before, if $\phi_{\min} + 2\pi - \omega_j - \rho_i \geq 2\pi$ then the sub-domain Φ is the union $[\phi_{\min}, 2\pi) \cup [0, \phi_{\min} - \omega_j - \rho_i]$.

3.3.2 Edge-Edge Contact

Parameterizing the configurations, which correspond to an (e_i-e_j) contact, can be again obtained using the parameterization of a corresponding (v_i-e_j) contact. Let us set

$$C(t) = S(t, \phi_{\min}),$$

where $S(\cdot, \cdot)$ is given in Equation (5) and ϕ_{\min} is the one defined in Section 3.1.1. In this case $C(0)$ corresponds to a (v_i-v_j) contact and $C(1)$ corresponds to a (v_i-v_{j+1}) contact. In both cases a_{i+1} lies on the line containing E_j^O . See Figure 6 for reference where it is shown that for $t \in [0, 1)$ we do not obtain the whole (e_i-e_j) contact. In order to complete the case, we have to let

$$t \in \left[0, 1 + \frac{\|E_i^A\|}{\|E_j^O\|}\right).$$

3.4 Summary of the Parameterization

In this section we derived, based on the fundamental motion described in Section 2, the parameterization of all the elements of the boundary of the forbidden space which correspond to a single obstacle. If the work space contains more obstacles, each one of them contributes another pillar-like element similar to the one depicted in Figure 9. Given an obstacle O , the portion of \mathcal{C} which is bounded “inside” the corresponding pillar-like object is the forbidden space related to O .

4 DIFFERENTIAL GEOMETRY OF CONTACT SURFACES

Using the parameterization that we developed, we study in this section the geometrical properties of the contact

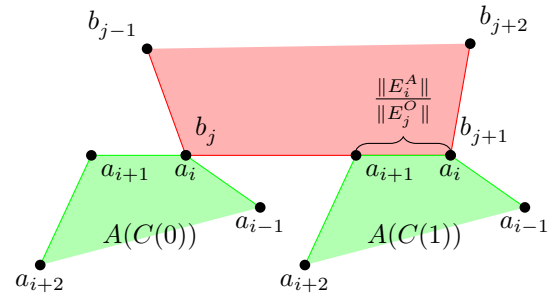


Figure 6: Setting of an edge-edge contact where $C(\cdot)$ is given in Section 3.3.2.

surfaces. As we already pointed out, the contact surfaces which correspond to vertex-edge contacts are developable and thus rather simple. We, therefore, focus in this section on the case of edge-vertex contact surfaces.

For the sake of simplicity, in Equation (8), we assume that b_j is at the origin. Thus we consider the contact surface $S(t, \phi) = c(\phi) + t\vec{r}(\phi)$ with

$$c(\phi) = \begin{pmatrix} -R^\phi a_i \\ \phi \end{pmatrix}, \quad \vec{r}(\phi) = \begin{pmatrix} R^\phi (a_i - a_{i+1}) \\ 0 \end{pmatrix}.$$

Note that $\vec{r}(\phi), \frac{d}{d\phi}\vec{r}(\phi) \neq 0$; this means that the (e_i-v_j) contact surface is a *non-cylindrical ruled surface* [dC76].

We start our study of the geometrical properties of the contact surface by computing its first (denoted E, F, G) and second (denoted e, f, g) fundamental forms:

$$\begin{aligned} E &= \|a_i - a_{i+1}\|^2 & e &= 0 \\ F &= \det(a_i, a_{i+1}) & f &= -\frac{\|a_i - a_{i+1}\|^2}{v} \\ G &= 1 + \|a_{i,t}\|^2 & g &= -\frac{\det(a_i, a_{i+1})}{v} \end{aligned}$$

where

$$v = v(t) = \sqrt{EG - F^2}.$$

It is easy to verify that $v \neq 0$ for all $t \in \mathbb{R}$, and thus, all expressions are well defined. Using standard formulas we can find the curvatures of the surface.

Lemma 2. *The Gaussian curvature $K(t, \phi)$ and the mean curvature $H(t, \phi)$ of an $(e-v)$ contact surface are given by:*

$$K(t, \phi) = -\frac{E^2}{v^4}, \quad H(t, \phi) = \frac{EF}{2v^3}.$$

Note that both the Gaussian and the mean curvature do not depend on ϕ but *only* on t as v depends on t . This comes as no surprise, as the surface is ruled. In other words, these curvatures depend on the point of pseudo contact along E_i^A . Lemma 2 also proves that the $(e-v)$

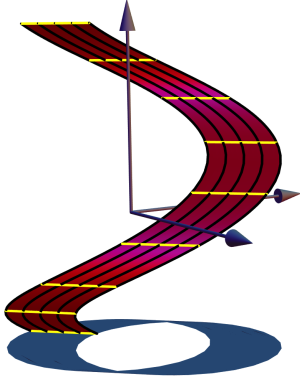


Figure 7: (v-e) contact surface.

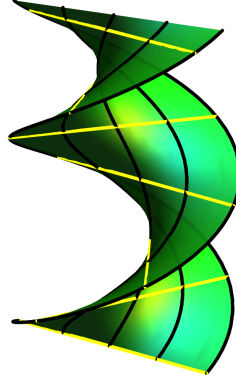
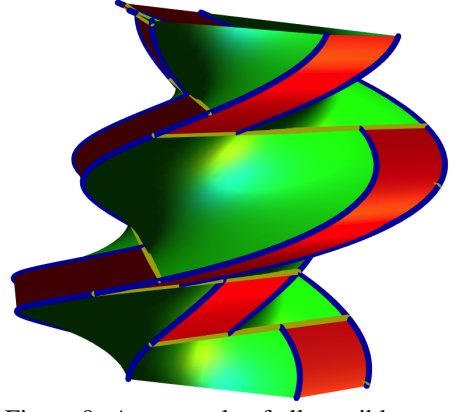


Figure 8: (e-v) contact surface.

Figure 9: An example of all possible contact patches for a given robot and one obstacle in \mathcal{C} .

contact surface is not developable since $K(t) < 0$. The next lemma establishes the cases when the edge-vertex contact surface is a *minimal surface*, and in turn also a *helicoid*.

Lemma 3. *If the line through $E_i^A(\mathbf{0})$ contains the reference point $R_0(\mathbf{0})$, then the corresponding (e_i-v_j) contact surface is a minimal surface.*

Proof. Recall that we assume that $R_0(\mathbf{0})$ is at the origin. Thus, if the line through $E_i^A(\mathbf{0})$ contains $R_0(\mathbf{0})$, then a_i and a_{i+1} , as vectors, are linearly dependent. Therefore $F = 0$, and in turn $H(t) = 0$. \square

It is easy to verify that $\frac{dK}{dt}(t_*) = \frac{dH}{dt}(t_*) = 0$ for

$$t_* = \frac{\langle a_i, a_i - a_{i+1} \rangle}{E}.$$

The *striction curve* [PW01, dC76] can now be found. Indeed, since $\lim_{t \rightarrow \pm\infty} K(t) = 0$, $K(t) < 0$ and $K'(t)$ vanishes only for t_* , it turns out that the Gaussian curvature attains its global extremum along the curve $S(t_*, \phi)$, which is therefore the striction curve. Note however, that t_* may not lie in the interval $[0, 1]$, that is, the striction curve of the (e-v) contact surface is not necessarily contained in it.

Next, we will compute the *normal curvature* of the (e-v) contact surface. Let $\varepsilon(\xi) \in T_p S$ be a unit tangent vector parameterized by a direction ξ . In turn, the *normal curvature* in direction ξ , denoted by $\kappa_N(\xi)$, is given by [Gra93]

$$\kappa_N(\xi) = \frac{E \sin \xi (F \sin \xi - 2v \cos \xi)}{v^3}. \quad (10)$$

Recall that v does not vanish for all $t \in \mathbb{R}$, and therefore Equation (10) is well defined. Note that the normal curvature depends only on the direction in the tangent plane, given by ξ , and the position on the ruling given by t .

Next, we want to find the *asymptotic directions* and the *principal curvature directions* of the contact surface. In other words we want to find the values of ξ (parameterized by t) for which the normal curvature either vanishes or attains an extremum. Furthermore, we will find the extremal values of the normal curvature, that is, expressions of the *principal curvatures*.

As the rulings on the surface are straight lines, the normal curvature in their directions should vanish, and indeed for $\xi \in \{0, \pi\}$ the normal curvature vanishes. The other direction where it vanishes corresponds to

$$\xi = \arctan \frac{2v}{F}.$$

It is easy to verify that $\frac{d}{d\xi} \kappa_N(\xi)$ vanishes for

$$\xi_1 = \frac{1}{2} \arctan \frac{2v}{F}.$$

and the normal curvature attains an extremum in this direction. Since the principal curvature directions are orthogonal, the normal curvature attains its other extremum for $\xi_2 = \xi_1 + \frac{\pi}{2}$. Finally, the principal curvatures are given by

$$\kappa_1 = \kappa_N(\xi_1) \quad , \quad \kappa_2 = \kappa_N(\xi_2).$$

One can verify that $K = \kappa_1 \cdot \kappa_2$. In Figure 10 a small sub-surface of an (e-v) contact surface is plotted together with the asymptotic and principal curvature directions.

Remark 6. The principal curvatures could have been computed directly using formulas which uses the fundamental forms. We took a longer path as we wanted to obtain expressions for the principal curvature directions as well.

5 CONCLUSION

Geometrical vs. Rational Models of \mathcal{C} .

In this paper we considered $\mathcal{C}^{\text{geom}}$, see Equation (1), as the model of the configuration space. Using this model

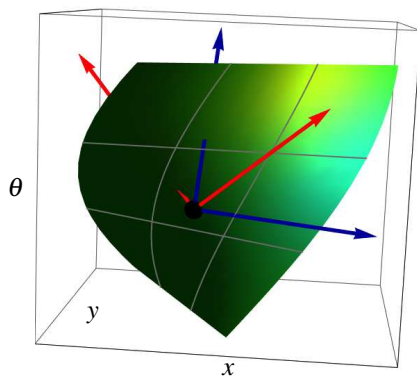


Figure 10: A sub-surface of an (e-v) contact surface. In blue and red the asymptotic and principal curvatures directions at some arbitrary point on the surface are respectively plotted.

it is easy to visualize elements in \mathcal{C} (cf. Figures 7 to 10). This, however, comes with a price. The parameterization of the contact surfaces or patches themselves, given in Equations (5) and (8), involves the trigonometric functions and thus it could not be computed in an exact manner. Furthermore, the computations of the sub-domains that correspond to the contact *patches* involve again trigonometric functions and their inverses, thus, once more, the final result cannot have an exact representation. This inexactness can, for example, have implications when one is using the parameterization to compute intersections between contact surfaces or patches. However, it is easy to replace $\mathcal{C}^{\text{geom}}$ with \mathcal{C}^{rat} , see Equation (2); this change can yield *rational* and *exact* representation of the contact surfaces. The trade-off in this case is that visualizing the contact surfaces in \mathcal{C}^{rat} is not intuitive.

Applications.

Using the parameterization described in this paper we produced a short video which visualizes the configuration space of a convex polygonal robot which moves amid convex polygonal obstacles in the plane. This video is available online [AR12].

Future Work.

The computations of the contact *patches* heavily rely on the assumption that the robot A is convex. Eliminating this restriction can be of interest. Other interesting extensions of the parameterization would be to consider a robot with a boundary which consists of non-linear edges; for example circular arcs (cf. [MST13]).

Once the contact surfaces and patches can be explicitly parameterized, it is natural to consider their discretizations. An approximated version of the configuration space can be used to address the general problem of

motion planning. Furthermore, given either the smooth or discrete representations of the contact surfaces or patches, it is possible to study their mutual intersections and their intersections with other elements in the configuration space, as done for example in [SHRH11]. This study can be of help in the investigation of the arrangement of the contact surfaces in the configuration space.

6 ACKNOWLEDGMENTS

This work is part of the project *Computational Geometric Learning*. The project CG Learning acknowledges the financial support of the Future and Emerging Technologies (FET) program within the Seventh Framework Program for Research of the European Commission, under FET-Open grant number 255827.

7 REFERENCES

- [AGR13] Dror Atariah, Sunayana Ghosh, and Günter Rote. On the parameterization and the geometry of the configuration space of a single planar robot. Technical report, Freie Universität Berlin, 2013. In preparation.
- [AR12] Dror Atariah and Günter Rote. Configuration space visualization. In *Proceedings of the 2012 symposium on Computational Geometry*, SoCG '12, pages 415–416, New York, NY, USA, 2012. ACM. <http://youtu.be/SBFwgR4K1Gk>.
- [BA88] Jean-Daniel Boissonnat and Francis Avnaim. Polygon placement under translation and rotation. Rapport de recherche RR-0889, INRIA, 1988.
- [CLH⁺05] Howie Choset, Kevin M. Lynch, Seth Hutchinson, George A. Kantor, Wolfram Burgard, Lydia E. Kavraki, and Sebastian Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations (Intelligent Robotics and Autonomous Agents series)*. MIT Press, 2005.
- [dC76] Manfredo P. do Carmo. *Differential geometry of curves and surfaces*. Prentice-Hall Inc., Englewood Cliffs, N.J., 1976. Translated from the Portuguese.
- [Gra93] Alfred Gray. *Modern Differential Geometry of Curves and Surfaces*. CRC Press, 1993.
- [HA92] Yong K. Hwang and Narendra Ahuja. Gross motion planning – A survey. *ACM Comput. Surv.*, 24(3):219–291, September 1992.
- [IKP73] M.B. Ignat'ev, F.M. Kulakov, and A.M. Pokrovskii. *Robot-manipulator control algorithms*. Joint Publications Research Service, 1973.

- [Lat93] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 3rd edition, 1993.
- [Lau98] Jean-Paul Laumond, editor. *Robot Motion Planning and Control*, volume 229 of *Lectures Notes in Control and Information Sciences*. Springer, 1998.
- [LaV06] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at <http://planning.cs.uiuc.edu/>.
- [LPW79] Tomás Lozano-Pérez and Michael A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles. *Commun. ACM*, 22(10):560–570, October 1979.
- [MST13] Victor Milenkovic, Elisha Sacks, and Steven Trac. Robust complete path planning in the plane. In Emilio Frazzoli, Tomas Lozano-Perez, Nicholas Roy, and Daniela Rus, editors, *Algorithmic Foundations of Robotics X*, volume 86 of *Springer Tracts in Advanced Robotics*, pages 37–52. Springer Berlin Heidelberg, 2013.
- [PW01] Helmut Pottmann and Johannes Wallner. *Computational Line Geometry*. Springer-Verlag New York, Inc., 2001.
- [SHRH11] Oren Salzman, Michael Hemmer, Barak Raveh, and Dan Halperin. Motion planning via manifold samples. In *ESA*, pages 493–505, 2011.
- [SS83] J.T. Schwartz and M. Sharir. On the piano movers’ problem: I. The case of a two-dimensional rigid polygonal body moving amidst polygonal barriers. *Communications on Pure and Applied Mathematics*, 36:345–398, 1983.
- [WB00] Kevin D. Wise and Adrian Bowyer. A survey of global configuration-space mapping techniques for a single robot in a static environment. *The International Journal of Robotics Research*, 19(8):762–779, 2000.