Static of Parallel Manipulators and Closeness to Singularity

J. Hubert
e-mail: julien.hubert@inria.fr

J.-P. Merlet
e-mail: jean-pierre.merlet@inria.fr

Institüt National de Recherche
en Informatique et en Automatique,
BP 93,
06902 Sophia-Antipolis,
France

Singularity is a major problem for parallel robots as in these configurations the robot cannot be controlled, and there may be infinite forces/torques in its joints, possibly leading to a robot breakdown. In the recent years classification and detection of singularities have made large progress. However, the issue of closeness to a singularity is still open and we propose in this paper an approach that is based on a static analysis. Our measure of closeness to a singularity is based on the very practical issue of having the joint forces/torques lower than a given threshold. We consider a planar parallel robot whose end-effector has a constant orientation and is submitted to a known wrench and we show that it is possible to compute the border of the region that describes all possible end-effector location for which the joint forces are lower than the fixed threshold.
[DOI: 10.1115/1.2961335]

1 Introduction

Singularity analysis of parallel robots has a long history starting with the pioneer work of Borel [1], Bricard [2], and Cauchy [3]. A major problem is that in these configurations the manipulator may exhibit infinitesimal motion although its actuators are locked. In the modern era such configurations have been studied by Hunt [4] and by Gosselin [5], and then this issue has been addressed by many authors [6–8].

It is usually claimed that singularities should be avoided because in the vicinity of a singularity the joint forces/torques can go to infinity, leading to a breakdown of the robot. This has led to major works to determine the singularity loci [9–13], to define a "distance" to a singular pose [14,15], to determine trajectory that avoid singularity [16–19], to investigate the relation between singularities and kinematics analysis [20], and finally to determine if a given workspace is singularity free in spite of the uncertainties in the robot modeling [21].

In this paper, we will follow another approach, which is motivated by a very practical consideration: avoiding breaking the robot by imposing a threshold on the maximal forces/torques in the legs. Hence if τ represents the forces/torques in the leg and $\tau_{\max}$ their allowed maximal value, we will define the *force workspace* as the set of poses so that

$$-\tau_{\max} \leqslant \tau \leqslant \tau_{\max} \tag{1}$$

for a given load on the platform. The purpose of this paper is to present an algorithm that allows one to compute the border of the *force workspace* for a given orientation of the platform. This algorithm will be illustrated on the planar $3-R\underline{P}R$ parallel robot, whose singularity analysis has been extensively developed [22].

2 Static Analysis

We will first consider a 6DOF parallel robot such as the Gough platform with legs whose attachment points on the base (platform) will be denoted by A_i (B_i). We define a reference frame O, $(\mathbf{x},\mathbf{y},\mathbf{z})$ and a mobile frame C $(\mathbf{x_r},\mathbf{y_r},\mathbf{z_r})$ that is attached to the moving platform, where C is an arbitrary point on the platform. If $\mathbf{J}$ denotes the kinematic Jacobian matrix and $\mathcal{F}$ is an external wrench exerted on the platform, then mechanical equilibrium is obtained if the forces in the leg satisfy

$$\mathcal{F} = \mathbf{J}^{-T}\tau \tag{2}$$

If ρ_i denotes the length of leg i, then the i-th row J_i^{-1} of $\mathbf{J}^{-1}$ is given by:

$$J_i^{-1} = \begin{pmatrix} \dfrac{\mathbf{A}_i\mathbf{B}_i}{\rho_i} & \dfrac{\mathbf{CB}_i \times \mathbf{A}_i\mathbf{B}_i}{\rho_i} \end{pmatrix}$$

We define the matrix $\mathbf{M}$ as the matrix whose i-th row is

$$M_i = (\mathbf{A}_i\mathbf{B}_i \quad \mathbf{CB}_i \times \mathbf{A}_i\mathbf{B}_i)^T$$

We also define the matrix $\mathbf{N}_i$ as the one obtained by substituting the i-th column of $\mathbf{J}^T$ by $\mathcal{F}$. Note that such Jacobian matrix formulation is valid not only for the Gough platform but also for many other types of parallel robot (e.g., the Hexa robot). A singularity is obtained if $\mathbf{J}^{-T}$ is singular, and it is easy to show that this is equivalent to $\mathbf{M}$ being singular.

According to Cramer's rule the i-th component τ_i of τ may be obtained from Eq. (2) as:

$$\tau_i = \rho_i \frac{|\mathbf{N}_i|}{|\mathbf{M}|} \tag{3}$$

provided that the determinant $|\mathbf{M}|$ of matrix $\mathbf{M}$ is not equal to 0, i.e., the platform is not in a singular configuration. At a singular pose Eq. (2) is an underconstrained linear system whose solution set may include solutions satisfying Eq. (1). Hence in terms of statics being exactly in a singularity may not be an issue. On the other hand, a necessary condition for a singular pose $\mathbf{X}_s$ to be reachable from a non-singular pose $\mathbf{X}$ while satisfying the constraints (1) is that the determinant $|\mathbf{N}_i|$ should go to 0 when the trajectory comes close to $\mathbf{X}_s$.

This motivates the study of the the force workspace where $-\tau_{\max} \leqslant \tau \leqslant \tau_{\max}$. For a spatial mechanism the force workspace is embedded in a six-dimensional space and is therefore difficult to draw. Hence we will restrict our study to the case where there are only two free parameters, thereby allowing to draw planar cross sections of the force workspace. For that purpose we will consider a $3-R\underline{P}R$ parallel robot (Fig. 1). The pose of such platform may be parametrized by the coordinates (x,y) of C in the reference frame and by the rotation angle θ between the $\mathbf{x}$ axis of the reference frame and the $\mathbf{x}_r$ axis of the mobile frame. Furthermore, this study will be performed under the assumption that the orientation angle θ and the wrench applied on the platform have both constant values (so that we have only (x,y) as free parameters).

It must be noted that in this case we have

$$\rho_i = \sqrt{(x-U_i)^2 + (y-V_i)^2} \tag{4}$$

where U_i and V_i are the constants that depend only on the geometry of the robot, the leg number, and on θ.

Contributed by the Mechanisms and Robotics Division of ASME for publication in the Journal of Mechanisms and Robotics. Manuscript received June 2, 2008; final manuscript received June 20, 2008; published online August 5, 2008. Review conducted by Andrew P. Murray. Paper presented at the ARK 2008.

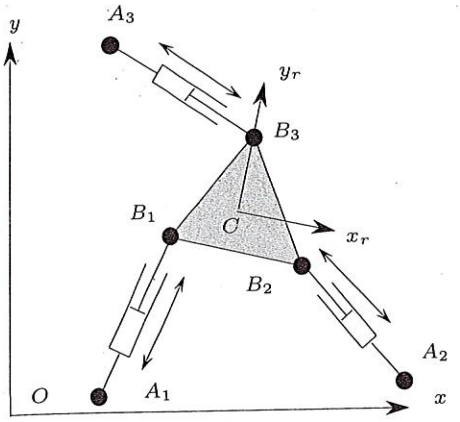

Fig. 1 3-*R̲PR*

3 Border of the Force Workspace

As a preliminary we will demonstrate the following theorem.

THEOREM 1. *A pose $\mathbf{X}_0$ will belong to the border of the force workspace if and only if at this pose the joint forces in one leg i satisfies $|\tau_i| = \tau_{\max}$. At this pose there is at least one motion direction such that even for an infinitesimal displacement along this direction, the change in one joint force will be such that $|\tau_i| > \tau_{\max}$.*

Proof. Let us assume that at $\mathbf{X}_0$ all joint forces satisfy $-\tau_{\max} < \tau_i < \tau_{\max}$ while $\mathbf{X}_0$ belongs to the force workspace border. Let us consider a ball centered at $\mathbf{X}_0$ of radius r and a pose $\mathbf{X}_f$ belonging to this ball. By using the implicit function theorem the joint force at $\mathbf{X}_f$ is obtained as

$$\tau(\mathbf{X}_f) = \left(\mathbf{J}^T(\mathbf{X}_0) + \frac{\partial \mathbf{J}^T(\mathbf{X}_1)}{\partial \mathbf{X}}(\mathbf{X}_f - \mathbf{X}_0)\right)\mathcal{F} \quad (5)$$

where $\mathbf{X}_1$ is a pose belonging to the ball. Within the ball $\partial \mathbf{J}^T(\mathbf{X}_1)/\partial \mathbf{X}$ is a bounded matrix and if ϵ is such that $\mathbf{J}^T(\mathbf{X}_0)\mathcal{F} = \tau_{\max} - \epsilon$, then there will always be a value of r such that in the ball around $\mathbf{X}_0$ the joint forces will verify $|\tau| < \tau_{\max}$. Hence $\mathbf{X}_0$ cannot be on the border of the workspace. ∎

As a consequence the border of the force workspace will be determined by looking at the poses for which $\tau_i = \tau_{\max}$. If $|M| \neq 0$ the constraint $\tau_i = \tau_{\max}$ may be written by using Eq. (3) as

$$\tau_{\max}|M| - \rho_i|N_i| = 0 \quad (6)$$

This equation in the two unknowns (x,y) defines a curve in the x-y plane that will be called *n-type curve*. Similarly if $\tau_i = -\tau_{\max}$ then Eq. (3) may be written as

$$\tau_{\max}|M| + \rho_i|N_i| = 0 \quad (7)$$

which defines a curves in the x-y plane, called *m-type curve*.

Consequently there are three n-type curves and three m-type curves. These curves are not algebraic as ρ_i is a square root of an algebraic equation in (x,y), according to Eq. (4). However, it must be noted that all points belonging to an m-curve or an n-curve satisfy the equation

$$\tau_{\max}^2|M|^2 + \rho_i^2|N_i|^2 = 0 \quad (8)$$

which is obtained as the product of Eqs. (6) and (7). This equation defines a curve called the *q-curves* that have the property to be algebraic of total degree 6.

Now, if we assume that $|M| = 0$ (i.e., the robot is in a singularity), then τ_i may be finite only in two cases, namely, if $|N_i| = 0$ or $\rho_i = 0$. In general the equations $|M| = 0$ and $|N_i| = 0$ are simultaneously satisfied only in a finite number of poses, which will be labeled as *N-points*. In the same manner poses at which both equations $|M| = 0$ and $\rho_i = 0$ will be denoted by *R-points*. It must be noted that both the R and N points belong simultaneously to the corresponding n and m-curves.

A consequence of Theorem 1 is that the border of the force workspace will be constituted of the arc of m- and n-curves and may possibly include some R and N points. We will now investigate how these arcs of curves may be determined.

4 Algorithm to Compute the Force Workspace Border

4.1 The Key-Points. Let us consider a point M_1 on the n-curve n_1 (i.e., at this pose we have $\tau_1 = \tau_{\max}$) and assume that $|\tau_2|, |\tau_3| \leq \tau_{\max}$. When moving along n_1 we may arrive at a pose M_2 at which we have $|\tau_2| > \tau_{\max}$. As the variation of τ_2 is a continuous function with respect to the motion on n_1, there must be a pose M_3 on the curve n_1, which lies between M_1 and M_2, such that $|\tau_2| = \tau_{\max}$. Hence, M_3 is an intersection point between n_1 and n_2 or m_2. The poses, which lies on an n-curve n_i or an m-curve m_i and at which $|\tau_j| = \tau_{\max}, i \neq j$, are called *key-points* and by extension the N and R points are also considered as key-points. We will now prove an interesting property of the key-points.

THEOREM 2. *Consider the part of an n-curve (or an m-curve) between two successive key-points. If a pose belonging to this part lie on the border of the force workspace, then the whole part is a component of the border.*

Proof. A pose M different from a key-point is on the border of the force workspace if at this pose we have $|\tau_i| = \tau_{\max}$, $|\tau_j| < \tau_{\max}, j \neq i$, i.e., M belongs to either an n-curve or an m-curve $\mathcal{S}$. Now assume that a pose M_f, belonging to the arc of curve $\mathcal{S}$ between two successive key-points, does not belong to the border, i.e., we have $|\tau_j| > \tau_{\max}, j \neq i$ at M_f. Then there must be a pose between M and M_f at which $|\tau_j| = \tau_{\max}, j \neq i$. Consequently, this pose belongs to both the n_i and n_j curves, i.e., is a key-point, which is contradictory with $\mathcal{S}$ being an arc between two successive key-points. ∎

A direct consequence is that for determining if a point of an n or an m-curve $\mathcal{S}_i$ lies on the border of the force workspace, it is sufficient to calculate the joint forces $\tau_j, j \neq i$ at this point and to verify if $|\tau_j| < \tau_{\max}$.

4.2 Determination of the Key-Points. In summary the key-points are as follows:

- the intersection points between the n and m-curves
- the R and Q points

The intersection between the n- and m-curves can be easily calculated by using the q-curves. Indeed by definition the intersection points between the q-curves, (q_i, q_j) curves are the intersection points of the pair of curves (n_i, n_j), (n_i, m_j), (m_i, n_j), and (m_i, m_j).

The intersection points between the two q-curves (q_i, q_j) may easily be calculated by computing the resultant of their equations and solving the resulting univariate polynomial. The roots of this polynomial are then introduced in the equations of (q_i, q_j), which become polynomials in one variable, and the common roots of these polynomials lead to the intersection points. For each of these intersection points we may then determine if it belongs to n_i, m_i, n_j, m_j just by computing the value of (τ_i, τ_j) at this point.

The calculation of the R and N points uses the same principle as follows:

- R points are obtained by calculating the resultant of $\rho_i^2 = 0, |M| = 0$.
- N points are obtained by calculating the resultant of $|N_i| = 0, |M| = 0$.

However, these key-points may not be sufficient to determine the border as the following two difficulties may occur.

1. An n- or m-curve has infinite branches.
2. The force workspace includes voids.

To manage the infinite branches, we will restrict the calculation to the part of the force workspace that is included within a given bounding box $\mathcal{B}$. Hence we will delete from our set of key-points those that are outside $\mathcal{B}$, but we will add to this set the intersection points of the n- and m-curves with the border of $\mathcal{B}$. These points are easily determined by setting successively (x,y) to the corresponding lower and upper bound of the (x,y) coordinates of $\mathcal{B}$ and then solving the polynomial of the q-curve (which is now univariate) and excluding the solutions that are outside the bounding box.

In the second case it may occur that an n- or m-curve $\mathcal{S}$ defines a closed region that is a void for the force workspace and that there is no key-point on $\mathcal{S}$. As we will see later on the determination of the force, workspace border is basically based on the assumption that all the elements that constitute the border lie between two key-points. In that particular case we may hence miss the void as $\mathcal{S}$ has no key-point. To deal with this problem it must be noted that if $\mathcal{S}$ defines a closed region, then there must be at least two points on this curve at which the tangent to $\mathcal{S}$ is horizontal. Such points may easily be determined by considering the algebraic system $q_i=0, \partial q_i/\partial y=0$, and then solving it by using the resultant approach. The solutions of this system that (1) lie on a n- or m-curves that has no key-points, and (2) are included in the bounding box are added as key-points for $\mathcal{S}$.

Finally, let us note that an n- or m-curve that will play a role in the border has at least two key-points and that an n- or m-curve that has no key-point should not be considered. Having an empty set of key-points may occur in two cases:

- the bounding box is strictly included in the force workspace and
- there is no intersection between the bounding box and the force workspace.

If the set of key-points is empty, it is sufficient to calculate the joint forces at an arbitrary point of the box. If all τ_i satisfy $|\tau_i| \leq \tau_{\max}$, then the box is included in the force workspace; otherwise if at least one τ_i satisfies $|\tau_i| > \tau_{\max}$, then there is no point of the force workspace in the bounding box. In the remaining sections of this paper, we will assume that the set of key-points is not empty.

4.3 Determining the Border Components. Using the results of the previous section it is easy to design two procedures that will be used for determining the border.

- Key_Points that implements the calculation of all key points as defined in the previous section. Note, however, that for the calculation of the border we may have to add a few additional key-points.
- On_Border (x,y) that will return 1 if (x,y) is on the border, 0 otherwise.

Our task is now to determine for all six n- and m-curves the arcs of curves that lie between two successive key-points (remember that we consider only the curves that have key-points and that the minimum number of key-point is 2). Without lack of generality we will explain the process for an n-curve n_i. We order the key-points by increasing the value of their x coordinates and start from the key-point K_1 with coordinates (x_1,y_1) that has the lowest x coordinates.

We consider the key point $K_f(x_f,y_f), f>1$ that is the next key-point after K_1 in the list (such point will be called a *goal point*). We then compute the two tangent unit vectors $\mathbf{T}_1,\mathbf{T}_2$ of the n-curve at the point K_1. Let us assume that $\mathbf{T}_1$ has components (t_x,t_y) and that $t_x^2 \geq t_y^2$.

- We substitute $x=x_1+\Delta, y=y_1+\epsilon$ in n_i, q_i where Δ is a small fixed increment.
- We solve $q_i=0$ in ϵ, looking only for solutions that also satisfy $n_i=0$ and which are "small," typically of absolute value lower than 10Δ.
- If we have obtained more than one solution, we divide Δ by 2 and starts again, while if no solution is obtained we exit from the process.
- If we get only one of such solution, we have obtained a new point K_n on the n-curve with coordinates $(x_n=x_1+\Delta, y_n=y_1+\epsilon)$.
- If On_Border $(x_n,y_n)=0$, the arc we are following is not part of the border and we exit from the process.

If the process has allowed us to determine a new point K_n, we repeat it by using K_n as the new K_1 point. This procedure will stop if one of the following cases occur.

- The current K_1 point is the last in the list of key-points.
- If $x_1+\Delta > x_f$ we adjust Δ is such way that $x_1+\Delta=x_f$. We then compute $y_n=y_1+\epsilon$:
  - (a) if $y_n=y_f$ we have then determined that K_1, K_f are successive key-points on n_i and have found a polygonal approximation of the arc that joins them.
  - (b) if $y_n \neq y_f$ the process continues, using K_{f+1} as new goal point.
- We compute the tangent unit vectors $\mathbf{T}_1=(t_x,t_y)$ at x_n,y_n. If $t_y^2 > t_x^2$ we stop the process and store (x_n,y_n) as new additional key-point. The motivation here is to increment by Δ the variable x or y that has locally the largest variation so that the other variable will exhibit only minor variation.

As soon as this process has terminated, we repeat it with the list of key-points ordered by descending value of the x coordinate and using $x_n=x_1-\Delta$ as the new x coordinate at each iteration.

This process is the same in the case where the components (t_x,t_y) of $\mathbf{T}_1$ verify $t_y^2 > t_x^2$ with only a permutation of the role of x,y in the algorithm (i.e., it is the y variable which is incremented by Δ).

Note that the R points require a specific treatment as they have no tangent vectors (indeed the derivatives of the n and m-curves with respect to (x,y) involve $1/\rho$ while at a R point $\rho=0$). For an R point of coordinates (x_1,y_1), we use all the small solutions obtained for $x_1 \pm \Delta$, $y_1 \pm \Delta$ as new key-points and the above procedure is not used for the R points.

As soon as all n- and m-curves that include K_1 as key-point have been processed by the algorithm, we will have determined all parts of the n- and m-curves issued from K_1 that are component of the force workspace border. The procedure Branches (K_i) returns all the arc of curves that belong to the border of the force workspace and are issued from the key-point K_i.

4.4 Border Calculation Algorithm. We may now describe the algorithm that compute the arc of curves belonging to the border of the force workspace. We will denote by $\mathcal{K}$ the set of key-points and by n the number of key-points in the set. The set of arcs of curves that will be returned is denoted by $\mathcal{A}$. Using the previous procedures the algorithm is simply written as

```
A=0
Key_Point( )
for i from 1 to n do
  U=Branches(K_i)
  A=A∪U
and-do
return A
```

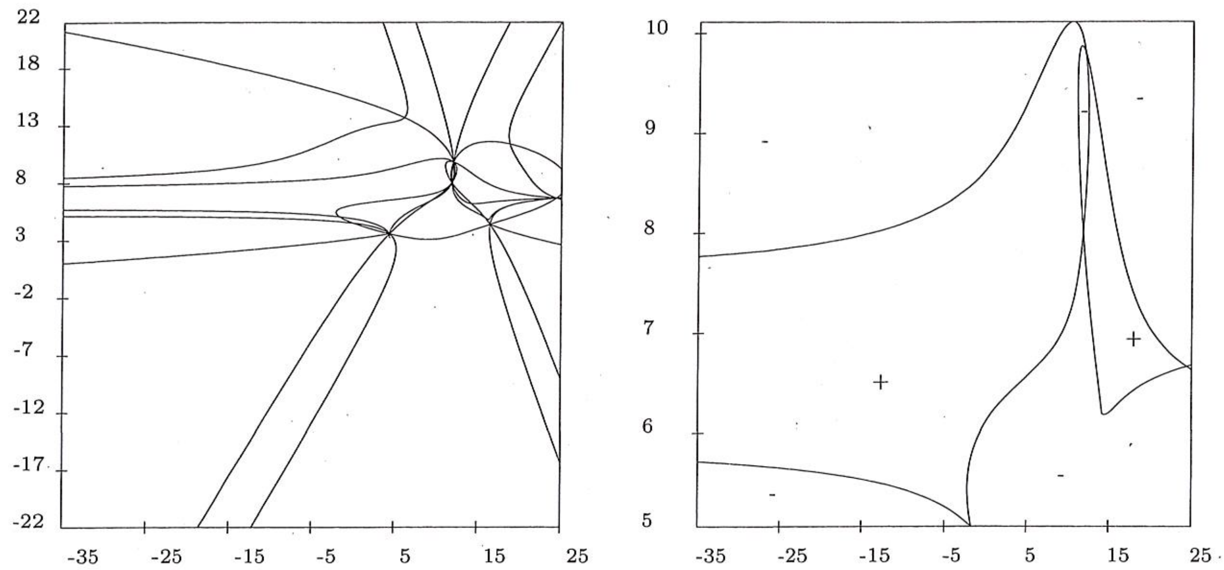

Fig. 2 The curves involved and the force workspace border for θ=0.1 rad and $\mathcal{F}$=(4,0,0) (the force workspace is constituted of the region with a + sign)

4.5 Managing Additional Kinematic Constraints. We may have to consider only the part of the force workspace that satisfy some kinematic constraints (e.g., that satisfy $\rho_{\min} \leqslant \rho \leqslant \rho_{\max}$ where $\rho_{\min}$ and $\rho_{\max}$ are the minimal and maximal allowed leg lengths).

This can easily be done in the procedure Branches by checking the constraints for the key-point and the new points. Only the poses that satisfy the constraints are stored and as soon as a new point satisfies the constraint while its predecessor was not, then a new arc of curve is created.

5 Examples and Analysis

The algorithm described in the paper was implemented as a prototype in MAPLE and was tested for the planar parallel robot 3-$R\underline{P}R$ whose geometry is defined by:

$$OA_1 = (0,0), \quad CB_{r1} = (-4,4)$$

$$OA_2 = (20,0), \quad CB_{r2} = (4,-4)$$

$$OA_3 = (12,10), \quad CB_{r3} = (0,2)$$

We choose a threshold $\tau_{\max}$=3 and an increment Δ=0.05. The computation time for determining the force workspace border varies between 50 s and 500 s according to the number of arcs of curves that are involved, but this computation time will be drastically reduced if the algorithm was implemented in C++.

Considering an external wrench $\mathcal{F}$=(4,0,0) and the platform orientation θ=0.1 rad, the involved curves and the resulting workspace are presented in Fig. 2. For this example the force workspace is separated into two components that are connected at two points, which are singular poses. The shape of the force workspace evolves slowly when changing the orientation of the platform. For example, Fig. 3 presents the force workspace for $\mathcal{F}$ =(4,0,0), θ=−0.1 rad. Note that in this example the force work-

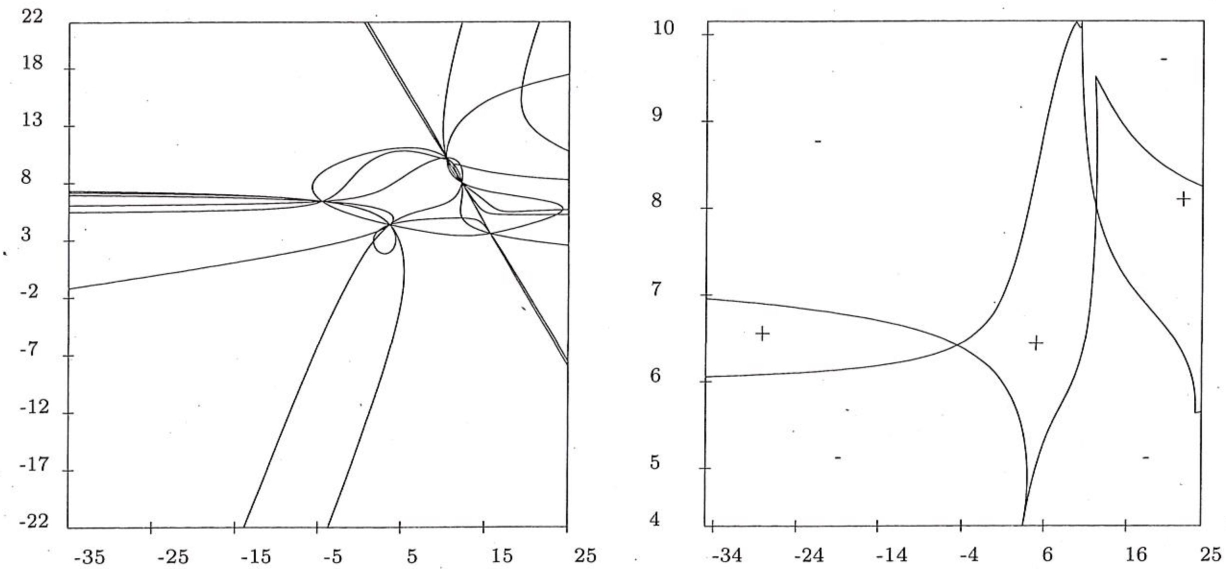

Fig. 3 The curves involved and the force workspace border for θ=−0.1 rad and $\mathcal{F}$=(4,0,0) (the force workspace is constituted of the region with a + sign)

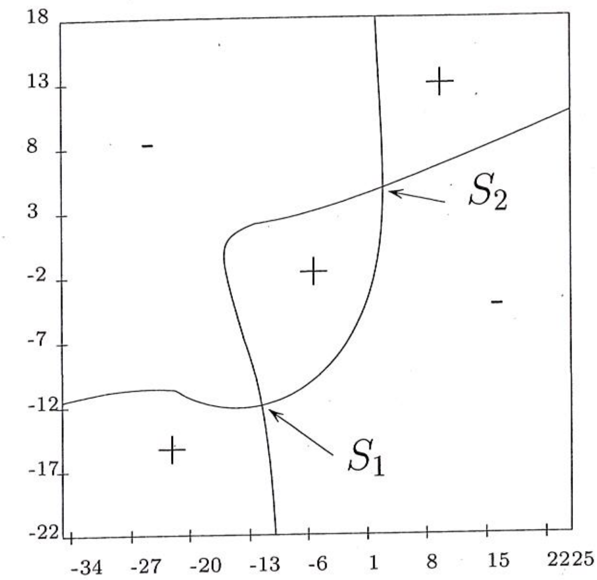

Fig. 4 The force workspace for θ=2.91 rad and $\mathcal{F}$=(0,0,5). The regions of the force workspace are denoted by a +.

space may seem to be unbounded but this occured only because we have restricted the drawing to lie within a bounding box. Indeed at infinity the legs are parallel, which leads to a singular configuration, and consequently there must be a force workspace border that prohibits the robot to move toward infinity. In our example this part is well outside the bounding box and does not appear in the drawing.

The number of components of the force workspace may, however, vary. For example, in Fig. 4 the force workspace obtained for $\mathcal{F}=(0,0,5)$ and $\theta=-2.91$ rad is composed of three components that are connected at point (S_1, S_2), which are singular.

The effect of the kinematic constraints may be seen on Fig. 5. Here we have imposed that the leg lengths are limited to the range [2,22]. It is well known that these constraints restrict the center of the platform to lie within a region that is the intersection of three circles minus the union of three circles. However, if we take into account the force workspace this workspace will usually be reduced. In the example presented in Fig. 5, the workspace has four components (the dashed circles are forbidden regions for the center of the platform).

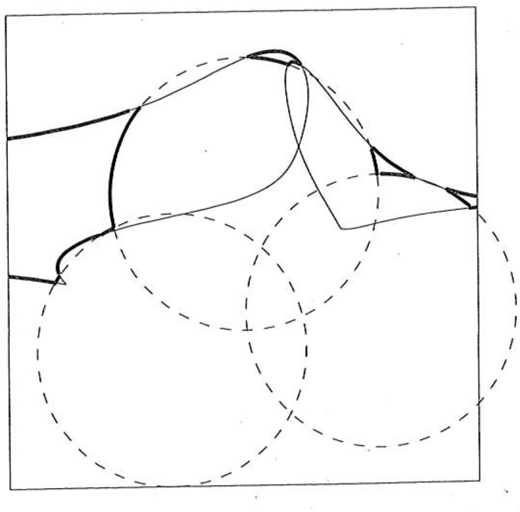

Fig. 5 The constraints induced by the minimal leg lengths impose that the platform center must lie outside the region defined by the union of the three dashed circles. Without taking these constraints into account the force workspace is constituted of two components while the real force workspace has four components (the region delimited by the border in thick line).

Going through a singularity is a major issue for good management of the robot workspace. Strategies have been proposed for avoiding singularities [16,17,19,23], for determining workspace components separated by singularity surfaces [24,25] and for using the dynamics to go through a singularity [26–28] but none of these strategies takes into account possible limits on the joint forces. Clearly joint clearances and deformations of the links may allow to open a "corridor" between two force workspace components, but determining if it is possible to control the robot through this corridor is an open issue.

6 Conclusion

We have presented in this paper a method that allows one to determine the border of the force workspace (i.e., the set of poses at which the joint forces are lower than a fixed threshold, being given a load on the platform) of a planar manipulator for a given orientation of the platform.

This approach is another way, based on an important physical requirement, to manage singularity. It has been illustrated on a planar parallel robot but may be used as well on other structures although it allows one to calculate only planar cross sections of the force workspace.

This work may be extended in various ways. It may be interesting to compute the force workspace for a given orientation range or for a set of wrenches but our algorithm cannot manage an exact calculation in that case. Using our approach we may still calculate the force workspace for various values of the orientation angle and/or external wrench then compute the intersection of all these force workspaces (which is possible as basically the force workspace border is a polygonal line and it exists efficient algorithms to compute the intersection of polygons). The result will be an overestimation of the real force workspace, and we may then use interval analysis to refine the result. A similar approach may be used if the wrench applied on the platform lie within some given range and to take modeling uncertainties (e.g., uncertainties on the location of the A_i, B_i) into account.

References

[1] Borel, E., 1998, "Mémoire sur les déplacements à trajectoire sphériques," Mech. Mach. Theory, **33**(1), pp. 1–128.

[2] Bricard, R., 1897, "Mémoire sur la théorie de l'octaèdre articulé," J. Math. Pures Appl., **3**, pp. 113–148.

[3] Cauchy, A., 1813, "Deuxième mémoire sur les polygones et les polyèdres," J. Ec. Polytech. (Paris), pp. 87–98.

[4] Hunt, K. H., 1978, *Kinematic Geometry of Mechanisms*, Clarendon, Oxford.

[5] Gosselin, C., and Angeles, J., 1990, "Singularity Analysis of Closed-Loop Kinematic Chains," IEEE Trans. Rob. Autom., **6**(3), pp. 281–290.

[6] Husty, M. L., and Karger, A., 2000, "Architecture Singular Parallel Manipulators and Their Self-Motions," in *ARK*, Piran, Jun., 25–29, pp. 355–364.

[7] Wohlhart, K., (1999), "Degrees of Shakiness," Mech. Mach. Theory, **34**(7), pp. 1103–1126.

[8] Zlatanov, D., Fenton, R. G., and Benhabib, B., 1998, "Identification and Classification of the Singular Configurations of Mechanisms," Mech. Mach. Theory, **33**(6), pp. 743–760.

[9] Mayer St-Onge, B., and Gosselin, C. M., 2000, "Singularity Analysis and Representation of the General Gough-Stewart Platform," Int. J. Robot. Res., **19**(3), pp. 271–288.

[10] Merlet, J.-P., 1989, "Singular Configurations of Parallel Manipulators and Grassmann Geometry," Int. J. Robot. Res., **8**(5), pp. 45–56.

[11] Pernkopf, F., 2003, "*Workspace Analysis of Stewart-Gough Platforms*" Ph.D. thesis, South Korea, University of Innsbruck, Innsbruck.

[12] Pottmann, H., Peternell, M., and Ravani, B., 1998, "Approximation in Line Space. Applications in Robot Kinematics," in, *ARK*, Strobl, Jun. 29–Jul. 4, pp. 403–412.

[13] Sefrioui, J., and Gosselin, C. M., 1995, "On the Quadratic Nature of the Singularity Curves of Planar Three-Degree-of-Freedom Parallel Manipulators," Mech. Mach. Theory, **30**(4), pp. 533–551.

[14] Voglewede, P. A., and Ebert-Uphoff, I., 2004, "Measuring Closeness to Singularities for Parallel Manipulators," in *IEEE International Conference on Robotics and Automation*, New Orleans, LA, Apr. 28–30, pp. 4539–4544.

[15] Wolf, A., and Shoham, M., 2003, "Investigation of Parallel Manipulators Us-

ing Linear Complex Approximation," ASME J. Mech. Des., **125**(3), pp. 564–572.
[16] Bhattacharya, S., Hatwal, H., and Ghosh, A., 1998, "Comparison of an Exact and an Approximate Method of Singularity Avoidance in Platform Type Parallel Manipulators," Mech. Mach. Theory, **33**(7), pp. 965–974.
[17] Nenchev, D. N., and Uchiyama, M., 1996, "Singularity-Consistent Path Planning and Control of Parallel Robot Motion Through Instantaneous-Self-Motion Type," in *IEEE International Conference on Robotics and Automation*, Minneapolis, MN, Apr. 24–26, pp. 1864–1870.
[18] O'Brien, J. F., and Wen, J. T., 2001, "Kinematic Control of Parallel Robots in the Presence of Unstable Singularities," in *IEEE International Conference on Robotics and Automation*, Seoul, South Korea, May 23–25, pp. 3154–3159.
[19] Sen, S., Dasgupta, B., and Mallik, A. K., 2003, "Variational Approach for Singularity-Path Planning of Parallel Manipulators," Mech. Mach. Theory, **38**(11), pp. 1165–1183.
[20] Zein, M., Wenger, P., and Chablat, D., 2006, "Singular Curves and Cusp Points in the Joint Space of 3-RPR Parallel Manipulators," in *IEEE International Conference on Robotics and Automation*, Orlando, FL, May 16–18, pp. 777–782.
[21] Merlet, J.-P., 2007, "A Formal-Numerical Approach for Robust in-Workspace Singularity Detection," IEEE Trans. Rob. Autom., **23**(3), pp. 393–402.
[22] Li, H., Gosselin, C. M., and Richard, M. J., 2006, "Determination of Maximal Singularity-Free Zones in the Workspace of Planar Three-Degree-of-Freedom Parallel Mechanisms," Mech. Mach. Theory, **41**(10), pp. 1157–1167.
[23] Dash, A. K., 2005, "Workspace Generation and Planning Singularity-Free Path for Parallel Manipulators," Mech. Mach. Theory, **40**(7), pp. 778–805.
[24] Wenger, P., and Chablat, D., 1998, "Workspace and Assembly Modes in Fully Parallel Manipulators: A Descriptive Study," in *ARK*, Strobl, Jun. 29–Jul 4, pp. 117–126.
[25] Zein, M., Wenger, P., and Chablat, D., 2007, "Singular Curves in the Joint Space and Cusp Points Of 3-*RPR* Parallel Manipulators," Robotica, **25**(6), pp. 712–724.
[26] Hesselbach, J., and others, 2002, "Connecting Assembly Modes for Workspace Enlargement," in, *ARK*, Caldes de Malavalla, Jun. 29–Jul. 2, pp. 347–356.
[27] Jui C., K. K., and Sun, Q., 2005, "Path Tracking of Parallel Manipulators in the Presence of Force Singularity," ASME J. Dyn. Syst., Meas., Control, **127**(4), pp. 550–563.
[28] Lee, M. K., and Park, K. W., 1999, "Kinematics and Dynamics Analysis of a Double Parallel Manipulator for Enlarging Workspace and Avoiding Singularities," IEEE Trans. Rob. Autom., **15**(6), pp. 1024–1034.