Determining the maximal singularity-free circle or sphere of parallel mechanisms using interval analysis

Mohammad Hadi Farzaneh Kaloorazi, Mehdi Tale Masouleh and Stéphane Caro

Robotica / *FirstView* Article / June 2014, pp 1 - 15
DOI: 10.1017/S0263574714001271, Published online: 13 June 2014

Robotica: page 1 of 15. © Cambridge University Press 2014
doi:10.1017/S0263574714001271

Determining the maximal singularity-free circle or sphere of parallel mechanisms using interval analysis

Mohammad Hadi Farzaneh Kaloorazi†*, Mehdi Tale Masouleh† and Stéphane Caro‡

†*Human-Robot Interaction Lab (TaarLab), Faculty of New Sciences and Technologies, University of Tehran, North Kargar, Tehran, Iran*
‡*IRCCyN/CNRS, UMR 6597, 1 rue de la Noë, 44321 Nantes, France*

(Accepted April 24, 2014)

SUMMARY
This paper proposes a systematic algorithm based on the concept of interval analysis to obtain the maximal singularity-free circle or sphere within the workspace of parallel mechanisms. As case studies the 3-RPR planar and 6-UPS parallel mechanisms are considered to illustrate the relevance of the algorithm for 2D and 3D workspaces. To this end, the main algorithm is divided into four sub-algorithms, which eases the understanding of the main approach and leads to a more effective and robust algorithm to solve the problem. The first step is introduced to obtain the constant-orientation workspace and then the singularity locus. The main purpose is to obtain the maximal singularity-free workspace for an initial guess. Eventually, the general maximal singularity-free workspace is obtained. The main contribution of the paper is the proposition of a systematic algorithm to obtain the maximal singularity-free circle/sphere in the workspace of parallel mechanisms. The combination of a maximal singularity-free circle or sphere with the workspace analysis by taking into account the stroke of actuators, as additional constraint to the latter problem, is considered. Moreover, the center point of the circle/sphere is not restrained to a prescribed point.

KEYWORDS: Parallel manipulators; Singularity analysis; Workspace analysis; Kinematics; Interval analysis.

Nomenclature

[.]	An interval variable
$[\mathcal{W}]$,$[\mathcal{S}]$	Lists of interval variables
$[\mathbf{b}_C]$	A two dimensional interval variable, called current box
$\mathcal{D}$	List of scalar variables, namely distance
C_f	Final center point
CAD	Computer Aided Design
CAM	Computer Aided Machining
DOF	Degree Of Freedom
EE	End-Effector
FKP	Forward Kinematic Problem
IKP	Inverse Kinematic Problem
MSFC	Maximal Singularity-Free Circle
MSFS	Maximal Singularity-Free Sphere
P	Prismatic joint
PPM	Planar Parallel Mechanisms
R	Revolute joint
S	Spherical joint, Consists of 3 perpendicular revolute joint
U	Universal joint, Consists of 2 perpendicular revolute joint

* Corresponding author. E-mail: mhfarzane@ut.ac.ir

1. Introduction

Due to the remarkable properties of Parallel Mechanisms (PMs),[1–5] such as higher payload to weight ratio and higher accuracy, these are now the state of the art for a wide range of commercial contexts, such as the Gough–Stewart platform for flight simulator and Delta robots for pick and place applications. More details on the true origins of PMs are elaborated in ref. [6]. However, there are some major deterrents to widespread PMs in the industrial context, including, among others, mathematical complexities in analyzing their kinematic properties, and extensive presence of uncontrollable configurations, referred to as singularities,[7] within their restricted workspace.

Since the limited workspace of PMs is coupled with singularities, in the design stage of a PM, the workspace analysis and singularity analysis are of paramount importance, which should be analyzed in such a way that leads to a PM with *singularity-free workspace*.[5,8,9] In singular configuration, a PM loses its inherent rigidity[10,11] and can be related mathematically to the singularity of some Jacobian matrices.[7,12] In fact, Jacobian matrices provide mapping between joint rates and the Cartesian velocities of the mechanism arising from the first-order kinematic properties of mechanism, i.e., equations coming from the differentiation of IKP with respect to time. In the literature, the singularities of PMs are classified from different perspectives,[13] and in this paper, the one proposed in ref. [7] is used, whose perspective is perhaps the closest in spirit to the logic of this paper. Directly from ref. [7], the singularities of PMs fall into three types: (1) Type I: inverse kinematic singularity, Type II: direct kinematic singularity and Type III: a combination of Type I and II. Due to high importance of Type II singularity, the latter is considered for the purpose of this paper.

There has been an extensive study conducted on the singularity-free workspace of PMs where most of them are based on complicated numerical approaches and entail some limits. These could not be extended to different kinds of parallel mechanisms and are proposed for a prescribed center point. It is of paramount importance to study the singularity-free workspace of a PM before going into the design stage, and this can be exemplified by the number of papers published on this issue. Bonev *et al.*[14] conducted an exhaustive study on the singularity locus of planar 3-DOF PMs by resorting to the screw theory. In ref. [15], a method based on the geometrical parameters is proposed, for which the singularity-free workspace of a three-legged PM is obtained. Li *et al.*,[16] by using the fact that the problem of maximum singularity-free circle of 3-DOF PMs can be expressed mathematically as an optimization problem accompanied with a constraint, resorted to the Lagrangian multipliers to solve the problem, and maximal singularity-free zone, which is a circle for a prescribed point, was obtained. Jiang and Gosselin[17–19] proposed some numerical recipes to find a singularity-free workspace of 3-DOF PMs. Recently, in ref. [20], upon resorting to particle swarm optimization, the maximum singularity-free circle of a 3-DOF PM was obtained for a prescribed center point. Moreover, in ref. [11], the problem of closeness to singularity is addressed by formulating the question in terms of constrained optimization problem.

The motivation behind choosing the circle/sphere is twofold: (1) The workspace could be replaced by a convex shape, i.e., a circle/sphere, which would be of great importance in kinematic properties optimization, and (2) it could be used for the problem of dimensional synthesis, i.e., the design parameters, such as the base and end-effector size and leg length of the PM, are of interest by prescribing the circle/sphere. In practice, usually it is desired to have a symmetric shape of the workspace. The singularity-free workspace is interesting for trajectory planning.[21] A method was presented in ref. [22] to determine whether there is a singularity in a given region defined in the workspace. The answer is definite and can be used to identify singularity-free zones inside the workspace. The singularity-free workspace of planar parallel manipulators with prismatic joints was addressed in ref. [23]. Both base and platform of the used manipulator are collinear. The singularity problem of planar 3-RPR parallel mechanisms was studied in ref. [16], in which a circle for a prescribed point was obtained. The singularity problem of the general Gough–Stewart platform was addressed in ref. [24], where a procedure was presented to determine a maximal singularity-free zone, which is a sphere around a point of interest P_0 for a prescribed orientation. This method was also extended to the six-dimensional (6D) workspace.

This paper aims at establishing a systematic approach based on *interval analysis*[25] to obtain MSFC – in the case of 3-DOF planar PMs (PPM) – and MSFS – in the case of 6-DOF Gough–Stewart platforms – for which the center of the circle is not given. Moreover, the boundaries of the workspace are also taken into account in the analysis. In the literature, some analytical approaches are introduced,

i.e., Lagrangian equations, which are hard to be extended to other problems. The aforementioned points distinguish this work from others reported in the literature.[8,15,16,20]

The proposed algorithm does not depend on the structure of the mechanism and is applicable to almost every mechanism. But resorting to interval analysis, in the case of complicated mechanisms having a high degree singularity expression, it may lead to a very time-consuming and inefficient process to obtain the aforementioned workspace. Use of consistency techniques may solve such a problem.[26]

The remainder of the paper is organized as follows. First, the general concept of interval analysis is presented. In Section 3, the proposed algorithm is fully explained and four *pseudo-codes* are provided for a better understanding of the problem. Moreover, some results are given that certify to show the effectiveness of the proposed algorithms. It noteworthy to say that the kinematic modeling of some case studies are addressed in the last section, i.e., the so-called 6-UPS PM and 3-RPR PPM.

2. Interval Analysis and Mathematical Framework

Several researchers independently had the idea bounding rounding errors by computing with intervals, e.g., Dwyer,[27] Sunuga,[28] Warmus,[29] and Wilkinson.[30] However, interval mathematics can be said to have begun with the appearance of Moore's book *Interval Analysis* in 1966.[31] Moore's book transformed this simple idea into a viable tool for error analysis. Instead of merely treating rounding errors, Moore extended the use of interval analysis to bound the effect of errors from all sources, including approximation and errors in data.[32] In the literature, interval analysis is regarded as a powerful numerical method to solve a wide range of problems such as, among others, circumventing round-off errors,[33] solving system of equations, optimization problem[32] and proper workspace presentation etc..[32,34–36] Furthermore, interval analysis provides an interactive visualization in the progress of calculation which is a definite asset in 2D and 3D representations of manipulator workspaces. Recently, upon revealing some remarkable features of interval analysis, such as finding the solution of a problem within some finite domain and taking into account the numerical computer round-off errors, it has stimulated the interests of many researchers in robotic community to deal with complicated problems such as IKP, FKP, calibration and the determination of a singularity-free workspace of parallel manipulator, the main concern of this paper.

Here is a list of advantages of using interval analysis instead of evolutionary approaches such as genetic algorithm. In the evolutionary techniques the chance of being trapped in a local optima is highly depended on initial population and initial search space. However, in the case of interval analysis, the only parameter to obtain global optima is choosing proper search space. In order to compute the maximum singularity-free workspace of PMs and other kinematic properties,[34,35] interval analysis entails the following advantages: (1) Contrary to other tools, which would result in a lengthy computation process and may converge to a local optimum, interval analysis is not a *black box*, since it requires combination of heuristics and numerical concepts to be effective; (2) it allows us to find all the solutions with inequalities within a given search space;[33,37] (3) for 2D and 3D problems, it leads to see the evolution of solutions and to monitor the procedure in order to have better insight to the problem and (4) it allows us to consider uncertainties in the model of a robot.

Interval analysis is a branch of mathematics that basically works with closed intervals instead of accurate numbers. An interval $[x]$ is a set of real numbers between two bounds and can be represented as:

$$[x] = [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}, \ (\underline{x} \leq \overline{x}), \tag{1}$$

where $\underline{x}$ and $\overline{x}$ are lower bound and upper bound respectively. All mathematical operations such as addition and multiplication can be performed on intervals. For instance[25]:

$$[x] + [y] = [\underline{x}, \overline{x}] + [\underline{y}, \overline{y}] = [\underline{x} + \underline{y}, \overline{x} + \overline{y}], \tag{2}$$

$$[x]\,[y] = [\min(S)\ ,\ \max(S)]\ , \quad S = \{\underline{x}\,\underline{y}, \underline{x}\,\overline{y}, \overline{x}\,\underline{y}, \overline{x}\,\overline{y}\}. \tag{3}$$

Moreover, a function of real numbers such as $f(x)$ can be evaluated as an interval from a given interval, $[x]$, which results in an interval $[f] = f([x])$. For example, for a monotonic function such

Algorithm 1 The pseudo-code representing the reasoning of algorithm to obtain the constant-orientation workspace of PMs. After % are comments

Initialize: list $[\mathcal{W}]$ containing initial search space interval vector (box)
Initialize: empty lists $[\mathcal{W}_{\text{in}}]$, $[\mathcal{W}_{\text{out}}]$ and $[\mathcal{W}_{\text{b}}]$ as inside, outside and boundary boxes respectively
Initialize: $[\mathbf{b}_{\text{C}}]$ and $[\text{r}_{\text{C}}]$ as the current box and the result corresponding to $[\mathbf{b}_{\text{C}}]$ respectively
while $[\mathcal{W}]$ is not empty **do**
    Extract first component of $[\mathcal{W}]$ and **copy** it to $[\mathbf{b}_{\text{C}}]$
    Delete first component of $[\mathcal{W}]$
    $[\text{r}_{\text{C}}] \leftarrow Eq.\ (5)|_{[\mathbf{b}_{\text{C}}]}$
    if $[\text{r}_{\text{C}}]$ is inside the workspace **then**
        Add $[\mathbf{b}_{\text{C}}]$ at the end of $[\mathcal{W}_{\text{in}}]$
    else if $[\text{r}_{\text{C}}]$ is outside the workspace **then**
        Add $[\mathbf{b}_{\text{C}}]$ at the end of $[\mathcal{W}_{\text{out}}]$
    else if $[\text{r}_{\text{C}}]$ lies within the boundary **then**
        if Diam$([\mathbf{b}_{\text{C}}] > \epsilon)$ **then**
            Bisect$([\mathbf{b}_{\text{C}}])$ by the largest edge
            Add two new boxes at the end of $[\mathcal{W}]$
        else% dimension threshold ϵ has been reached
            Add $[\mathbf{b}_{\text{C}}]$ at the end of $[\mathcal{W}_{\text{b}}]$
        end if
    end if
end while
Return $[\mathcal{W}_{\text{b}}]$ as the workspace boundaries

as $f(x) = x^3$:

$$[f] = f([x]) = [f(\underline{x}),\ f(\overline{x})] = [\underline{x}^3, \overline{x}^3]. \tag{4}$$

The whole concept of interval analysis is based on bisecting a box (or a hyper-box in higher dimensional space), called *branch & prune* approach,[35] upon considering some well-defined algebra on intervals, in such a way that the latter box will converge toward the desired solution. In this paper, we are taking a step back from what it has been done up to now in the the literature. In fact, more emphasis is placed on how one can define a procedure to evolve the first box, which is chosen arbitrarily by the user to generate new ones to converge to the desired solution. In short, as it will be explained in an upcoming section, upon blending some classical concepts of numerical analysis with interval analysis, some obstacles to obtain a singularity-free workspace can be eliminated.

3. Obtaining Workspace, Singularity-free Zone and MSFC/MSFS for a Given Box and Prescribed Orientation of a Moving Platform

In this section, four algorithms, denoted as Algorithm 1, 2, 3 and 4 are proposed. The first three algorithms are for a given orientation of a moving platform and applicable for a prescribed box, called current box $[\mathbf{b}_{\text{C}}]$, which is defined by the user. In other words, in the aforementioned algorithms, there is no evolution on $[\mathbf{b}_{\text{C}}]$ and no new boxes are generated outside this box (creating boxes inside $[\mathbf{b}_{\text{C}}]$ is an inherent concept of interval analysis) and all the results are only valid for $[\mathbf{b}_{\text{C}}]$. It should be noted that the result of Algorithm 3 in practice is similar to what has been done in refs. [16, 20]. Therefore, in this paper, in order to circumvent this problem, Algorithm 3 is improved, called Algorithm 4, in such a way that based on some imposed rules new boxes are generated, which conduct us to the optimum solution for MSFC/MSFS of the workspace as a whole. For the sake of clarity, lists are represented in calligraphic ($\mathcal{C}$).

The proposed algorithms are represented in what follows. Note that for the sake of simplicity, all procedures are explained for a 3-RPR planar PM, its constant-orientation is represented in 2D. Then the procedure can be readily developed to higher DOF PMs, and later on results are represented for 6-UPS PM.

3.1. The constant-orientation workspace (Algorithm 1)

The constant-orientation workspace of a PM is a set of all feasible points for which the EE can reach for a prescribed orientation. Several methods are reported in the literature,[10,38] which range from numerical approaches to geometrical approaches (such as the usage of CAD software[39]). It should be noted that several approaches based on interval analysis are proposed to solve the constant-orientation workspace of PMs, and in this paper, following the same reasoning as the one proposed in ref. [10], an algorithm, represented in Algorithm 1 and denoted as `A`, is proposed. The problem of obtaining the constant-orientation workspace consists in solving the IKP, Eq. (9), for given $\rho_{\min}$ and $\rho_{\max}$, characterizing the stroke of actuators, for a given orientation of the moving platform. Thus, the problem reduces to solve inequalities as follows for $i = 1, 2, 3$ for a 3-RPR PM and $i = 1, \ldots, 6$ for a 6-UPS PM in order to obtain the set of feasible points (x, y):

$$\begin{aligned}(x_{Ai} - x_{Bi})^2 + (y_{Ai} - y_{Bi})^2 &\leq \rho_{\max}^2,\\ (x_{Ai} - x_{Bi})^2 + (y_{Ai} - y_{Bi})^2 &\geq \rho_{\min}^2.\end{aligned} \tag{5}$$

Algorithm 1 shows a pseudo-code which solves the above system of inequalities in order to determine the constant-orientation workspace. It should be noted that Algorithm 1 results in the following three types of lists, which are the input data for the computation of singularity-free workspace for Algorithm 3: (1) the inner area of the workspace ($[\mathcal{W}_{\text{in}}]$), (2) the outer area of the workspace ($[\mathcal{W}_{\text{out}}]$) and (3) the boundaries of the workspace ($[\mathcal{W}_b]$). If box $[\mathbf{b}_{\text{C}}]$ satisfies Eq. (5) for all legs simultaneously, then $[\mathbf{b}_{\text{C}}]$ will be inside the workspace and will be added to $[\mathcal{W}_{\text{in}}]$. If $[\mathbf{b}_{\text{C}}]$ does not satisfy Eq. (5) even for one leg, it will be a part of $[\mathcal{W}_{\text{out}}]$, otherwise it will be added at the end of $[\mathcal{W}]$. As can be observed from Algorithm 1, the algorithm continues the procedure until the dimension of $[\mathbf{b}_{\text{C}}]$ is higher than a given threshold ϵ, which stands for the accuracy of the algorithm. Once the desired accuracy is reached, the remaining boxes will be considered as $[\mathcal{W}_{\text{b}}]$.

For a 3-RPR PPM, the $[\mathbf{b}_{\text{C}}]$ is a 2D box in the xy-plane. If only for one leg, $[\mathbf{b}_{\text{C}}]$ is applied to Eq. (5), then this equation is converted to an interval function or equation. Therefore, the result of such a function is interval $[\text{r}_{\text{C}}]$,

$$[\text{r}_{\text{C}}] = (x_{Ai} - [\text{b}_{\text{C}x}])^2 + (y_{Ai} - [\text{b}_{\text{C}}]_y)^2. \tag{6}$$

If both the upper bound of $[\text{r}_{\text{C}}]$, called $\overline{\text{r}_{\text{C}}}$, and lower bound of $[\text{r}_{\text{C}}]$, called $\underline{\text{r}_{\text{C}}}$, are within the range of $\rho_{\min}^2 < \cdot < \rho_{\max}^2$, then it can be deduced that $[\mathbf{b}_{\text{C}}]$ satisfies the mechanical strokes for the corresponding leg. By the same token, $[\mathbf{b}_{\text{C}}]$ must satisfy all legs mechanical strokes.

3.2. The constant-orientation singularity locus (Algorithm 2)

The reasoning applied for Algorithm 2 in order to obtain the inner area ($[\mathcal{S}_{\text{in}}]$), the outer area ($[\mathcal{S}_{\text{out}}]$) and the boundary ($[\mathcal{S}_b]$) of the singularity is to the majority of intents and purposes the same as Algorithm 1 for the constant-orientation workspace, and there is a slight difference to illustrate the algorithm from another viewpoint. In this case, Eq. (13), representing the singularity locus, should be considered. The pseudo-code of this section is represented in Algorithm 2. The differences between Algorithms 1 and 2: If the result of substitution of $[\mathbf{b}_{\text{C}}]$ into Eq. (13) is negative, then it will be added to $[\mathcal{S}_{\text{in}}]$, otherwise it is completely positive and will be a part of $[\mathcal{S}_{\text{out}}]$. In the case the box $[\text{r}_{\text{C}}]$ contains zero, it will be added at the end of $[\mathcal{S}]$. Algorithm 2 pursues the concept of interval analysis on box $[\mathbf{b}_{\text{C}}]$, defined latter, until the desired accuracy ϵ is reached. After reaching ϵ, the remaining boxes are considered as $[\mathcal{S}_{\text{b}}]$.

3.3. The constant-orientation MSFC/MSFS for a given initial box (Algorithm 3)

From Algorithms 1 and 2, having in hand the boundaries of the workspace and the singularity locus in an interval form, Algorithm 3 computes the optimum center and radius of MSFC/MSFS for a given box, called initial guess box and denoted as $[\mathbf{b}_0]$. As mentioned previously, Algorithm 3 works only on the initial box $[\mathbf{b}_0]$, chosen arbitrary by the user, and thus it may not contain the optimum center of the MSFC/MSFS. The algorithm starts first to substitute $[\mathbf{b}_0]$ into $[\mathbf{b}_{\text{C}}]$. Then the concept of interval analysis is applied to $[\mathbf{b}_{\text{C}}]$, meaning that two sub boxes, namely $[\mathbf{b}_{\text{C1}}]$ and $[\mathbf{b}_{\text{C2}}]$, are created by bisecting $[\mathbf{b}_{\text{C}}]$. Then the algorithm calculates the distances of the two new boxes obtained latter

Algorithm 2 The pseudo-code representing the reasoning of algorithm to obtain the constant-orientation singularity locus of PMs. After % are comments.

Initialize: list $[\mathcal{S}]$ containing initial search space interval (box)
Initialize: empty lists $[\mathcal{S}_{\text{in}}]$, $[\mathcal{S}_{\text{out}}]$ and $[\mathcal{S}_{\text{b}}]$ as inside, outside and boundary boxes respectively
Initialize: $[\mathbf{b}_{\text{C}}]$ and $[\text{r}_{\text{C}}]$ as the current box and the result corresponding to $[\mathbf{b}_{\text{C}}]$
while $[\mathcal{S}]$ is not empty **do**
    Extract first component of $[\mathcal{S}]$ and **copy** it to $[\mathbf{b}_{\text{C}}]$
    Delete first component of $[\mathcal{S}]$
    $[\text{r}_{\text{C}}] \leftarrow Eq.\ (13)|_{[\mathbf{b}_{\text{C}}]}$
    if $[\text{r}_{\text{C}}] < 0$ **then** % inside the singularity locus
        Add $[\mathbf{b}_{\text{C}}]$ at the end of $[\mathcal{S}_{\text{in}}]$
    else if $[\text{r}_{\text{C}}] > 0$ **then** % outside the singularity locus
        Add $[\mathbf{b}_{\text{C}}]$ at the end of $[\mathcal{S}_{\text{out}}]$
    else if $0 \in [\text{r}_{\text{C}}]$ **then** % lies within the boundary
        if Diam$([\mathbf{b}_{\text{C}}] > \epsilon)$ **then**
            Bisect$([\mathbf{b}_{\text{C}}])$ by the largest edge
            Add two new boxes at the end of $[\mathcal{S}]$
        else% dimension threshold ϵ has been reached
            Add $[\mathbf{b}_{\text{C}}]$ at the end of $[\mathcal{S}_{\text{b}}]$
        end if
    end if
end while
Return $[\mathcal{S}_{\text{b}}]$ as the workspace boundaries

Algorithm 3 The pseudo-code representing the reasoning of algorithm in order to obtain the MSFC of PMs for an initial box. After % are comments.

Initialize: interval-list $[\mathcal{B}]$ containing initial boundaries, such as workspace boundaries ($[\mathcal{W}_{\text{b}}]$ taken from Algorithm 1) and singularity locus boundaries ($[\mathcal{S}_{\text{b}}]$ taken from Algorithm 2)
Initialize: box $[\mathbf{b}_0]$ as initial guess box, prescribed by user
$[\mathbf{b}_{\text{C}}] \leftarrow [\mathbf{b}_{\text{i}}]$
while Diam$([\mathbf{b}_{\text{C}}]) < \epsilon$ **do**
    $\{[\mathbf{b}_{\text{C1}}], [\mathbf{b}_{\text{C2}}]\} \leftarrow$ **Bisect**$([\mathbf{b}_{\text{C}}])$
    % bisect by the largest edge
    $\mathcal{D}_{C1} \leftarrow \|[\mathbf{b}_{\text{C1}}] - [\mathcal{B}]\|$
    $\mathcal{D}_{C2} \leftarrow \|[\mathbf{b}_{\text{C2}}] - [\mathcal{B}]\|$
    % calculate distance from current box to boundaries
    if $\min(\mathcal{D}_{C1}) > \min(\mathcal{D}_{C2})$ **then**
        $[\mathbf{b}_{\text{C}}] \leftarrow [\mathbf{b}_{\text{C1}}]$
        $\mathcal{D} \leftarrow \mathcal{D}_{C1}$
    else
        $[\mathbf{b}_{\text{C}}] \leftarrow [\mathbf{b}_{\text{C2}}]$
        $\mathcal{D} \leftarrow \mathcal{D}_{C2}$
    end if
end while
Return $C_0 \leftarrow$ **Center**$([\mathbf{b}_{\text{C}}])$ % as the center point of MSFC
Return $r \leftarrow \min(\mathcal{D})$ % as the radius of MSFC

and considers the one being farther from the boundaries of the constant-orientation workspace and the singularity locus which are listed in $[\mathcal{B}]$ (this part is done on the basis of the data provided by Algorithms 1 and 2).

If the desired accuracy, represented by ϵ, is achieved then the procedure will end, otherwise the algorithm will substitute the corresponding box into $[\mathbf{b}_{\text{C}}]$ and the procedure will continue. While it reaches the desired accuracy, the minimum value of $\mathcal{D}_{Ci}$, $i = 1, 2$, gives the radius. Using the lower bound of $\mathcal{D}$ leads to ascertain that the obtained radius does not intersect the singularity locus. Then

Algorithm 4 The pseudo-code describing the reasoning of algorithm in order to obtain the improved MSFC of PMs for all orientations. After % are comments.

```
for all orientations do
    Run Algorithm 3 (Algorithm 3), consider initial guess box [b_0] and point C_0 as the center of [b_0]
    Return C_1
    i ← 0
    while C_{i+1} is close to the edges of [b_i] do
        i ← i + 1
        [b_i] ← Create_box(Dimensions([b_0]),C_i)
      % create box [b_i] with C_i as the center point and with the same dimension as [b_0]
        Run Algorithm 3 (Algorithm 3)
        Return C_{i+1}
    end while
    Return C_{i+1} as the center point of MSFC and corresponding r as its radius
end for
Plot the profile of optimum circles for all orientations
```

by a proper rounding, the center of the last interval is the center of the MSFC/MSFS for the current box $[\mathbf{b}_0]$ and the last radius derived from $\mathcal{D}$ is the radius of the MSFC.

In Algorithm 3, if the initial box $[\mathbf{b}_0]$ does not contain the center point of the MSFC/MSFS for the whole workspace, once the procedure reaches the accuracy of ϵ, the obtained center point is close to one edge of $[\mathbf{b}_0]$. Here the definition of closeness is as follows: The distance of the obtained center point is lower than ϵ. The next section will represent an approach, referred to as *Improved MSFC/MSFS*, which circumvent the latter problem and find the MSFC/MSFS for the workspace as a whole whether or not the initial guess box contains the center of optimum circle/sphere.

3.4. Algorithm to obtain improved MSFC/MSFS independent of the initial guess box (Algorithm 4)

Reaching this step, to the end of circumventing some shortcomings of Algorithm 3, new features are given to Algorithm 3 in order to present a new algorithm which guarantees that the obtained MSFC/MSFS is an optimum solution for the workspace as a whole. This algorithm, called Algorithm 4, can generate new boxes from the initial box $[\mathbf{b}_0]$ defined by user in order to converge to the optimum MSFC/MSFS for the entire workspace.

Assume that $[\mathbf{b}_0]$ is an initial box for which Algorithm 3 is run. Then in the case that the obtained optimum center point corresponding to $[\mathbf{b}_0]$ is close to its edges, Algorithm 4 generates a new box, called $[\mathbf{b}_1]$, for which the same reasoning as Algorithm 3 is applied for. Algorithm 4 pursues to generate new boxes until the obtained center point would not be close to the edges of the box under study. The closeness to the edges of the box under study is based on a given criterion, such as ϵ defined previously.

In other words, when the obtained center point from Algorithm 3 is close to the edges of $[\mathbf{b}_0]$, one can predict the direction for which the center point tends to approach toward its optimum value leading to the MSFC/MSFS of the workspace as a whole. Using this point as the center of a new box, called $[\mathbf{b}_1]$, which has the same size as $[\mathbf{b}_0]$, Algorithm 4 can be used. The improved algorithm continues until the obtained center point would not be close to the edges of the box, $[\mathbf{b}_i]$, $i = 1, 2, \ldots, n$, under study. The above algorithm is run for a given orientation of the moving platform and can be repeated for a range of moving platform orientations.

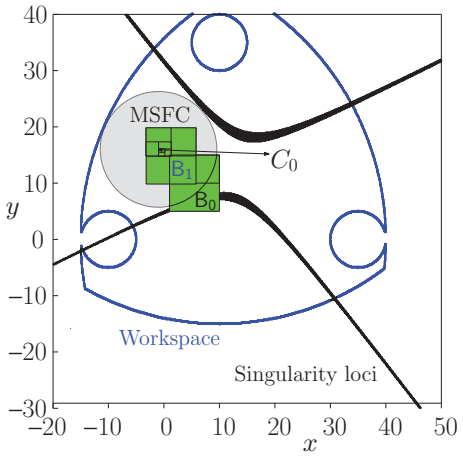
4. Results

4.1. Results obtained for 3-RPR planar parallel mechanism

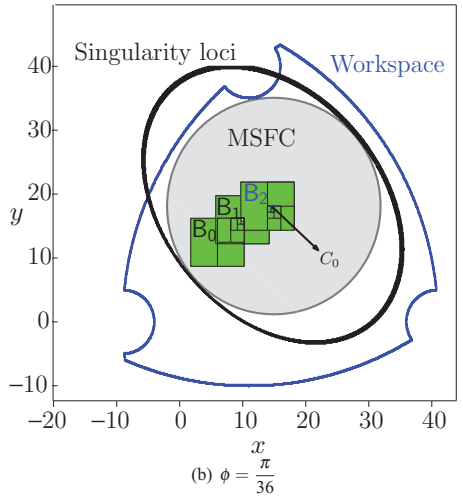
In order to have a better insight into the reasoning of all algorithms proposed in this paper, this section is devoted to the analysis of the obtained solutions from Algorithms 1, 2, 3 and 4 for MSFC of a 3-RPR with design parameters as given in Table I.

Table I. Geometric parameters of a 3-RPR PM (all lengths are given in mm).

i	B_{ix}	B_{iy}	$\rho_{\min}$	$\rho_{\max}$	P'_{ix}	P'_{iy}
1	3.78	4.34	0.5	5	−4.83	−3.19
2	34.47	−3.78	0.5	5	12.04	−3.19
3	16.23	34.76	0.5	5	8.23	12.09



(a) $\phi = 0$



(b) $\phi = \frac{\pi}{36}$

Fig. 1. (Colour online) Results for the workspace, singularity locus and MSFC for two different orientations of the moving platform in which new boxes are generated from B_0 in order to find the center C_0 of the optimum center of the circle. For the sake of clarity and in order to not overload the figure, some lines corresponding to small boxes produced by interval analysis are omitted.

Figure 1(a) represents the constant-orientation workspace obtained from Algorithm 1, the singularity – a hyperbola in this case – obtained from Algorithm 2, for $\phi = 0$. Then, in order to find the MSFC, an initial box $[\mathbf{b}_0]$ is selected, for which Algorithm 4 is applied. It should be noted that Algorithm 3 is inside Algorithm 4 and will be applied for each box generated by Algorithm 4, including the initial box $[\mathbf{b}_0]$. As can be observed in Fig. 1(a), Algorithm 4 generates a new box, $[\mathbf{b}_1]$, since upon applying Algorithm 3 into $[\mathbf{b}_0]$, the obtained center point lies on the boundary of $[\mathbf{b}_0]$. Finally, by applying Algorithm 3 on $[\mathbf{b}_1]$, the obtained center point is not on the boundary of $[\mathbf{b}_1]$ and consequently the obtained center point, C_f, is the optimum one for the MSFC, and the radius can be readily obtained using the distance formula of two intervals.

Figure 1(b) represents a situation for which $\phi = \frac{\pi}{36}$, where, as can be observed, the singularity locus is an ellipse. It can be readily concluded that for the initial box, Algorithm 4 converges to

Table II. Geometric parameters of 6-UPS parallel robot under study (all lengths are given in mm).

i	1	2	3	4	5	6
x_{ai}	92.58	132.58	40.00	−40.00	−132.58	−92.58
y_{ai}	99.64	30.36	−130.00	−130.00	30.36	99.64
z_{ai}	23.10	23.10	23.10	23.10	23.10	23.10
x_{bi}	30.00	78.22	48.22	−48.22	−78.22	−30.00
y_{bi}	73.00	−10.52	−62.48	−62.48	−10.52	73.00
z_{bi}	−37.10	−37.10	−37.10	−37.10	−37.10	−37.10
$\rho_{i_{\min}}$	454.5	454.5	454.5	454.5	454.5	454.5
$\rho_{i_{\max}}$	504.5	504.5	504.5	504.5	504.5	504.5

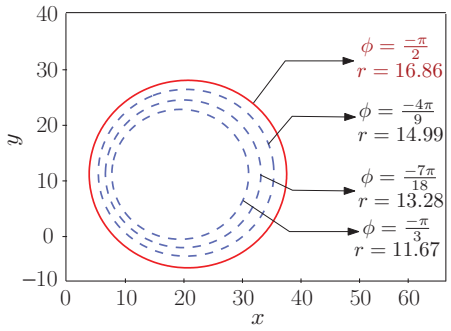


Fig. 2. (Colour online) The MSFC for $\phi = [-\frac{\pi}{2}, -\frac{\pi}{3}]$ for a 3-RPR PM with geometrical parameters defined in Table 1.

the optimum MSFC by generating two new boxes where $[\mathbf{b}_2]$ contains the center of the MSFC, C_f. Finally, Fig. 2, obtained with Algorithm 4, depicts obtained MSFC for several moving platform orientations, $\phi = [-\frac{\pi}{2}, -\frac{\pi}{3}]$. Figure 2 shows that the radius r of the MSFC is maximum for $\phi = -\frac{\pi}{2}$, $r = 16.86$ mm.

4.2. Results obtained for 6-UPS planar parallel mechanism

This section represents the results obtained by using the proposed algorithm to find MSFS for a 6-UPS parallel robot with the design parameters given in Table II.

Figure 3 represents the constant-orientation workspace for $\theta = 0$, $\phi = 0$, $\psi = 0$ and $z = [510, 540]$ obtained with Algorithm 1. In turn, for the same set of orientations used in Fig. 3, Figs. 4(a) and (b) represent respectively the implicit and interval-based (Algorithm 2) representations of the singularity loci of a parallel robot under study.

Moreover, as it can be observed in Fig. 5, the MSFS is tangent to both singularity locus and workspace boundaries. By inspection, it can be inferred that the obtained MSFS corresponds to the optimal inscribed sphere bounded by workspace and singularity locus.

For the sake of better understanding, Fig. 6 represents a cross-sectional view of the result depicted in Fig. 5. It is worth mentioning that in Fig. 6 the gray circles are cross-section views of the MSFS in different xy-planes. As can be observed in Fig. 6, the gray circles are tangent neither to the workspace boundaries nor to the singularity locus. The latter statement is in accordance with the fact that the MSFS should absolutely be tangent to the constraints of the problem, since this takes place in a 3D space. Indeed, the set of points tangent to workspace or singularity locus, is not lying in the prescribed cross-sectional planes along the z-axis.

5. Conclusions and Future Works

This paper presented four interval-based algorithms, Algorithms 1, 2, 3 and 4, in order to obtain respectively the constant-orientation workspace, the singularity, the maximal singularity-free workspace for a given box and the maximal singularity-free workspace for the entire workspace. As case studies, the proposed algorithms were used to obtain MSFC for a 3-DOF planar parallel

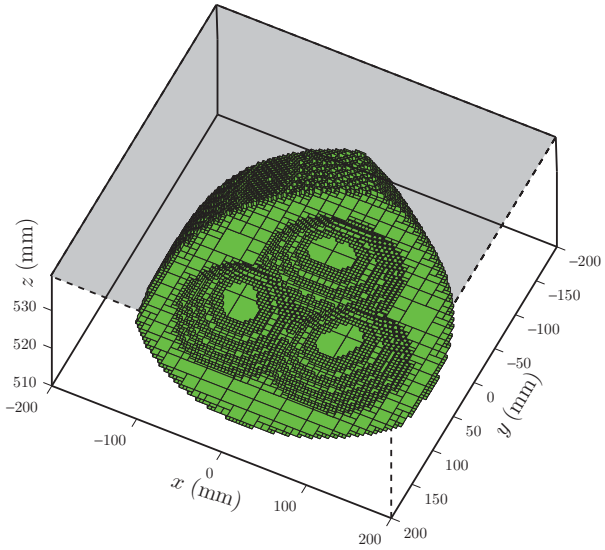
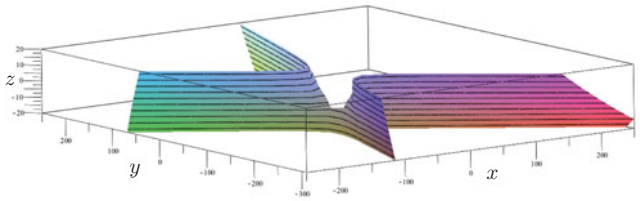
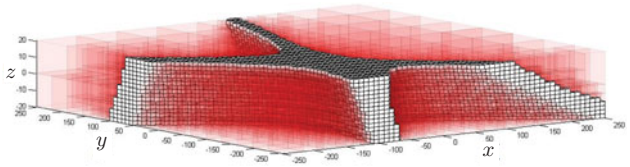


Fig. 3. (Colour online) The constant-orientation workspace of 6-UPS parallel robot with design properties as given in Table II, using interval analysis section of $z = [510, 540]$.



(a) Implicit representation of singularity loci of a 6-UPS parallel robot.



(b) Singularity loci surface of a 6-UPS parallel robot obtained from Algorithm 1. The set of red transparent boxes are outside the singularity loci and white boxes lie on the singularity loci.

Fig. 4. (Colour online) Singularity loci of the 6-UPS parallel robot with design parameters given in Table II: (a) Implicitly depicted, (b) interval-based.

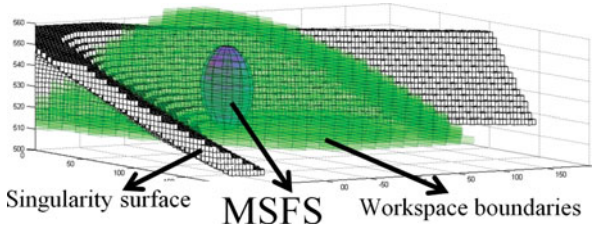


Fig. 5. (Colour online) The MSFS of a 6-UPS parallel robot for $\theta = 0,\ \phi = 0,\ \psi = 0$ and $z = [510, 550]$. The set of white boxes represents the singularity surface, and green-transparent boxes are inside the constant-orientation workspace. MSFS is obtained in such a way that it is tangent to both singularity surface and workspace.

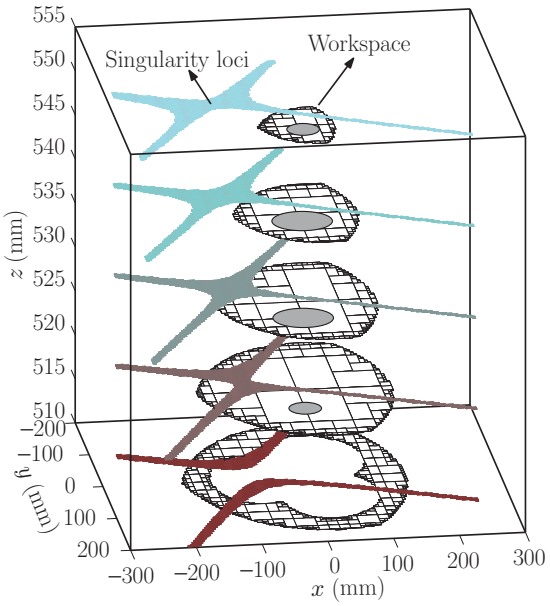


Fig. 6. (Colour online) Interval-based results for MSFS for $\theta = 0$, $\phi = 0$ and $\psi = 0$. Only cross-sectional plans are considered to represent the constant-orientation workspace and the singularity locus of the robot not to overload the figure. The gray circles represent sections of MSFS.

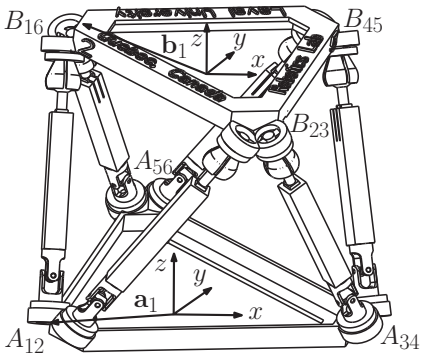


Fig. 7. The MSSM Gough–Stewart platform. The schematic is adapted from ref. [40].

robot, i.e., 3-RPR and MSFS for a 6-DOF parallel robot, i.e., 6-UPS Minimal Simplified Symmetric Manipulator (MSSM) Gough–Stewart platform. The proposed algorithms laid down the state of the art for formulating the problem of finding the maximal singularity-free workspace of parallel robots. However, it could be readily extended to other robots, and opens an avenue to find a systematic approach to do so. The results obtained from these algorithms revealed that the proposed algorithms are robust and could also be used for the optimum synthesis of robots under study. Ongoing works include the extension of the proposed algorithms to the analysis of complex parallel robots and the dimension synthesis of parallel robots based on the algorithms presented in this paper.

A. Review of Kinematic Properties, Inverse Kinematics, Singularity and Workspace

A.1. 6-UPS parallel mechanism

In this section, three important kinematic properties, namely IKP, singularity analysis and workspace, are broadly reviewed. The IKP pertains to finding the values of joint variables for a given position and orientation of EE. Figure 7 represents a MSSM 6-UPS parallel robot. It should be noted that P stands for an actuated prismatic joint. In order to clearly establish the notation used here (inspired from ref. [38]), consider a fixed coordinate frame $R : O\text{-}xyz$ attached to the base platform, and a

moving coordinate frame $R' : O'\text{-}x'y'z'$ attached to EE. Moreover, the ith leg is attached to the base platform at point A_i and to EE at point B_i. Vectors $\mathbf{a}_i$ and $\mathbf{b}_i$, $i = 1, \ldots, 6$, are the position vectors of points A_i and B_i, expressed in frames R and R' respectively. Furthermore, $\mathbf{Q}$ denotes the rotation matrix characterizing the orientation between frames R and R', and ρ_i is the joint variable of the ith prismatic joint. The position vector of point B_i expressed in the fixed frame R can be written as:

$$[\mathbf{b}_i]_R = [\mathbf{r}]_R + \mathbf{Q}[\mathbf{b}_i]_{R'}, \quad i = 1, \ldots, 6, \tag{7}$$

where $[\mathbf{r}]_R = [x_r, y_r, z_r]^T$ stands for the position vector of point O' expressed in frame R and the subscript R indicates that the corresponding vector is expressed in frame R. Subtracting $\mathbf{a}_i$ from both sides of Eq. (7) leads to:

$$[\mathbf{b}_i - \mathbf{a}_i]_R = [\mathbf{r}]_R + \mathbf{Q}[\mathbf{b}_i]_{R'} - [\mathbf{a}_i]_R. \tag{8}$$

The left-hand side of Eq. (8) is clearly the vector connecting point A_i to point B_i, hence by taking the Euclidean norm of each side, one can obtain the IKP of the ith limb as follows:

$$\rho_i = \| [\mathbf{b}_i - \mathbf{a}_i]_R \|_2 = \| [\mathbf{r}]_R + \mathbf{Q}[\mathbf{b}_i]_{R'} - [\mathbf{a}_i]_R \|_2, \tag{9}$$

where $\| \cdot \|_2$ stands for the Euclidean norm. Therefore, for a given robot, the actuated variable ρ_i can be directly computed for a given position and orientation of EE.

In this paper, the actuation singularity, referred to as Type II[7] is more of concern, which occurs when the moving platform possesses certain degrees of freedom whereas the actuators are locked. As a necessary condition, the rank of the actuation system of a parallel robot in a non-singular configuration should be equal to six. Once this rank decreases, an infinitesimal motion of EE will occur and the platform will be uncontrollable. Upon resorting to the *screw theory*,[1, 14] one can write the kinematical screw system, $\$_i$, for a 6-UPS limb as:

$$\$_\mathbf{i} = \begin{bmatrix} \mathbf{e}_1^T & (\mathbf{a}_i \times \mathbf{e}_1)^T \\ \mathbf{e}_2^T & (\mathbf{a}_i \times \mathbf{e}_1)^T \\ \mathbf{0}^T & \mathbf{e}_{\rho_i}^T \\ \mathbf{e}_1^T & (\mathbf{b}_i \times \mathbf{e}_1)^T \\ \mathbf{e}_2^T & (\mathbf{b}_i \times \mathbf{e}_2)^T \\ \mathbf{e}_3^T & (\mathbf{b}_i \times \mathbf{e}_3)^T \end{bmatrix}, \quad i = 1, \ldots 6, \tag{10}$$

in which $\mathbf{e}_{\rho_i}$ is the unit vector of the direction of the ith prismatic joint. From Eq. (10), it can be concluded that no constraint wrench is imposed by the limb to EE, therefore the robot under study has 6-DOF. Furthermore, the forward Jacobian matrix of a 6-UPS robot takes the following form:

$$\mathbf{J} = \begin{bmatrix} \mathbf{e}_{\rho_1}^T & (\mathbf{b}_1 \times \mathbf{e}_{\rho_1})^T \\ \mathbf{e}_{\rho_2}^T & (\mathbf{b}_2 \times \mathbf{e}_{\rho_2})^T \\ \mathbf{e}_{\rho_3}^T & (\mathbf{b}_3 \times \mathbf{e}_{\rho_3})^T \\ \mathbf{e}_{\rho_4}^T & (\mathbf{b}_4 \times \mathbf{e}_{\rho_4})^T \\ \mathbf{e}_{\rho_5}^T & (\mathbf{b}_5 \times \mathbf{e}_{\rho_5})^T \\ \mathbf{e}_{\rho_6}^T & (\mathbf{b}_6 \times \mathbf{e}_{\rho_6})^T \end{bmatrix}. \tag{11}$$

Each row of the Jacobian matrix is a Plücker line and corresponds to a screw reciprocal to all passive twists of the corresponding limb, but not to the actuated twist. $\det(\mathbf{J}) = 0$ represents the singularity locus of the robot.

The workspace of a robot consists of a set of Cartesian points that can be reached by the EE of the manipulator. The solution of IKP can be used to obtain the workspace of the robot for a given

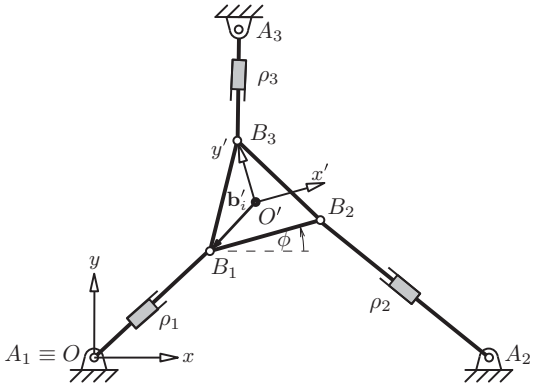


Fig. 8. A schematic representation for a 3-RPR PM.[41]

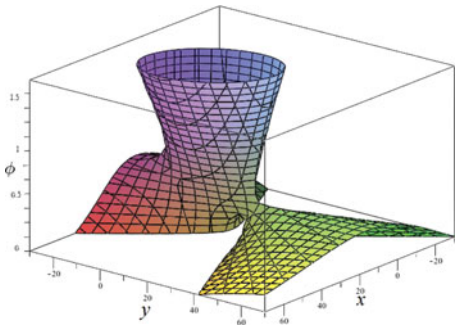


Fig. 9. (Colour online) 3D singularity locus of a 3-RPR PM for $\phi = [0, \frac{\pi}{2}]$.

mechanical stroke associated to each limb, $\rho_{\min} < \rho_i < \rho_{\max}$. $\rho_{\min}$ and $\rho_{\max}$ are the lower bound and the upper bound of the actuated prismatic joints respectively. Hence, each point of the Cartesian space that satisfies Eq. (9) for a given stroke belongs to the workspace of the manipulator. Furthermore, the workspace of a PM can be resorted as a geometrical reasoning. For instance, the workspace of a 6-UPS parallel robot is a common area of intersection of six inner and six outer spheres, known as vertex space. The inner spheres correspond to the lower limits of actuated prismatic joints and the outer ones correspond to the upper limits of actuated prismatic joints.

A.2. 3-RPR planar parallel mechanism

Figure 8 illustrates the schematic representation of a 3-RPR PPM. As depicted in the figure, a planar 3-RPR PM with actuated prismatic joints consists of a fixed triangular ($\Delta_{A_1A_2A_3}$) and a mobile triangular platform ($\Delta_{B_1B_2B_3}$). The passive revolute joints, located at A_i and B_i, are connected by the prismatic actuator of variable length ρ_i, $i = 1, 2, 3$. The IKP of a 3-RPR is similar to Eq. (9) with $i = 1, 2, 3$.

The first-order kinematic relation, coming from the differentiation of IKP with respect to time, of a 3-RPR PM in a matrix form, called the Jacobian, can be expressed as follows:

$$\mathbf{J} = \begin{bmatrix} n_{1x} & n_{1y} & (\mathbf{b}_1 \times \mathbf{n}_1) \cdot \mathbf{k} \\ n_{2x} & n_{2y} & (\mathbf{b}_2 \times \mathbf{n}_2) \cdot \mathbf{k} \\ n_{3x} & n_{3y} & (\mathbf{b}_3 \times \mathbf{n}_3) \cdot \mathbf{k} \end{bmatrix}. \tag{12}$$

In the above, $\mathbf{k}$ is the unit vector along the z-axis, $\mathbf{b}_i$, $i = 1, 2, 3$, is the position vector of point B_i expressed in the fixed frame, and the unit vector along the ith prismatic joint direction is denoted by $\mathbf{n}_i = [n_{i_x}, n_{i_y}, 0]^T$.

More information concerning the kinematic properties of 3-RPR PMs can be found in ref. [41]. In short, the singularity, or more precisely the Type II singularity,[7] is the pose, i.e., position and

orientation (x, y, ϕ), for which the determinant of $\mathbf{J}$ vanishes:

$$D(x, y, \phi) = \det(\mathbf{J}) = 0. \tag{13}$$

From refs. [19, 42], the singularity curves of a 3-RPR fall into three types for different orientations of a moving platform: (1) a hyperbola, (2) a parabola and (3) an ellipse (ellipse may degenerate into a circle). The latter can also be inferred by simple observation of Fig. 9.

References

1. X. Kong and C. Gosselin, *Type Synthesis of Parallel Mechanisms*, vol. 33 (Springer, Heidelberg, Germany, 2007).
2. B. Li, Y. Cao, Q. Zhang and Z. Huang, "Position-singularity analysis of a special class of the stewart parallel mechanisms with two dissimilar semi-symmetrical hexagons," *Robotica* **31**(1), 123–136 (2013).
3. X.-J. Liu, J. Wang, F. Gao and L.-P. Wang, "Mechanism design of a simplified 6-dof 6-rus parallel manipulator," *Robotica* **20**(1), 81–91 (2002).
4. T. Li and S. Payandeh, "Design of spherical parallel mechanisms for application to laparoscopic surgery," *Robotica* **20**(3), 133–138 (2002).
5. G. Bhutani and T. A. Dwarakanath, "Practical feasibility of a high-precision 3-upu parallel mechanism," *Robotica* **1**(8) 1–15 (2013).
6. "Parallelmic," available at: http://www.parallemic.org/Reviews/Review007.html (accessed January 24, 2003).
7. C. Gosselin and J. Angeles, "Singularity analysis of closed-loop kinematic chains," *IEEE Trans. Robot. Autom.* **6**(3), 281–290 (1990).
8. M. Tale Masouleh and C. Gosselin, "Determination of singularity-free zones in the workspace of planar 3-PRR parallel mechanisms," *J. Mech. Des.* **129**, 649 (2007).
9. S. Bhattacharya, H. Hatwal and A. Ghosh, "On the optimum design of stewart platform type parallel manipulators," *Robotica* **13**(3), 133–140 (1995).
10. J. P. Merlet, *Parallel Robots* (Springer, Heidelberg, Germany, 2006).
11. P. Voglewede and I. Ebert-Uphoff, "Overarching framework for measuring closeness to singularities of parallel manipulators," *IEEE Trans. Robot.* **21**(6), 1037–1045 (2005).
12. J. P. Merlet and P. Donelan, "On the regularity of the inverse Jacobian of parallel robots," **In:** *Advances in Robot Kinematics* (L. Jadran and B. Roth, eds.) (Springer, Heidelberg, Germany, 2006) pp. 41–48.
13. M. Conconi and M. Carricato, "A new assessment of singularities of parallel kinematic chains," *IEEE Trans. Robot.* **25**(4), 757–770 (2009).
14. I. A. Bonev, D. Zlatanov and C. M. Gosselin, "Singularity analysis of 3-DOF planar parallel mechanisms via screw theory," *J. Mech. Des.* **125**, 573 (2003).
15. Y. Yang and J. O'Brien, "A Case Study of Planar 3-RPR Parallel Robot Singularity Free Workspace Design," *Proceedings of the International Conference on Mechatronics and Automation (ICMA)* (IEEE, New York, NY, 2007) pp. 1834–1838.
16. H. Li, C. Gosselin and M. Richard, "Determination of maximal singularity-free zones in the workspace of planar three-degree-of-freedom parallel mechanisms," *Mech. Mach. Theory* **41**(10), 1157–1167 (2006).
17. Q. Jiang and C. Gosselin, "Geometric synthesis of planar 3-RPR parallel mechanisms for singularity-free workspace," *Trans. Can. Soc. Mech. Eng.* **33**(4), 667–678 (2009).
18. Q. Jiang and C. Gosselin, "The Maximal Singularity-Free Workspace of Planar 3-RPR Parallel Mechanisms," *Proceedings of the 2006 International Conference on Mechatronics and Automation* (IEEE, New York, NY, 2006) pp. 142–146.
19. Q. Jiang and C. M. Gosselin, "Geometric optimization of planar 3-RPR parallel mechanisms," *Trans. Can. Soc. Mech. Eng.* **31**(4), 457–468 (2007).
20. G. Abbasnejad, H. Daniali and S. Kazemi, "A new approach to determine the maximal singularity-free zone of 3-RPR planar parallel manipulator," *Robotica* **1**(1), 1–8.
21. Q. Jiang, "Singularity-Free Workspace Analysis and Geometric Optimization of Parallel Mechanisms," PhD Dissertation, Université Laval, Canada (2008).
22. J.-P. Merlet and D. Daney, "A Formal-Numerical Approach to Determine the Presence of Singularity within the Workspace of a Parallel Robot." *Proceedings of the 2nd Workshop on Computational Kinematics* (2001) pp. 167–176.
23. M. Gallant and R. Boudreau, "The synthesis of planar parallel manipulators with prismatic joints for an optimal, singularity-free workspace," *J. Robot. Syst.* **19**(1), 13–24 (2002).
24. H. Li, C. Gosselin and M. Richard, "Determination of the maximal singularity-free zones in the six-dimensional workspace of the general Gough-Stewart platform," *Mech. Mach. Theory* **42**(4), 497–511 (2007).
25. R. E. Moore and F. Bierbaum, *Methods and Applications of Interval Analysis*, vol. 2 (Society for Industrial Mathematics, Philadelphia, PA, 1979).
26. F. Abdallah, A. Gning and P. Bonnifait, "Box particle filtering for nonlinear state estimation using interval analysis," *Automatica* **44**(3), 807–815 (2008).

27. P. Dwyer, "Computation with approximate numbers," **In:** *Linear Computations* (Wiley, New York, NY, 1951) pp. 11–34.
28. T. Sungana, "Theory of interval algebra and application to numerical analysis," *Res. Assoc. Appl. Geom. Mem.* **2** , 29–46 (1958).
29. M. Warmus, "Calculus of approximations," *Bull. Acad. Pol. Sci.* **4**(5), 253–257 (1956).
30. J. Wilkinson, "Turings Work at the National Physical Laboratory and the Construction of Pilot Ace, Deuce, and Ace," **In:** *A History of Computing in the Twentieth Century* (Metropolis *et al.*, eds.) (Academic Press, New York, NY, 1980), pp. 101–114 [MHR80].
31. R. E. Moore, *Interval Analysis. Series in Automatic Computation* (Prentice-Hall, Englewood Cliff, NJ, 1966).
32. E. Hansen and G. Walster, *Global Optimization Using Interval Analysis: Revised and Expanded*, vol. 264 (Boca Raton, FL, 2003).
33. J. P. Merlet, "Solving the forward kinematics of a Gough-type parallel manipulator with interval analysis," *Int. J. Robot. Res.* **23**(3), 221–235 (2004).
34. F. Hao and J. P. Merlet, "Multi-criteria optimal design of parallel manipulators based on interval analysis," *Mech. Mach. Theory* **40**(2), 157–171 (2005).
35. J. P. Merlet, "Interval analysis and robotics," *Robot. Res.* (Springer, 2011) pp. 147–156.
36. D. Chablat, P. Wenger, F. Majou and J. P. Merlet, "An interval analysis based study for the design and the comparison of three-degrees-of-freedom parallel kinematic machines," *Int. J. Robot. Res.* **23**(6), 615–624 (2004).
37. D. Oetomo, D. Daney, B. Shirinzadeh and J. P. Merlet, "Certified Workspace Analysis of 3RRR Planar Parallel Flexure Mechanism," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, New York, NY, 2008) pp. 3838–3843.
38. C. Gosselin, "Determination of the workspace of 6-DOF parallel manipulators," *ASME J. Mech. Des.* **112**(3), 331–336 (1990).
39. M. Saadatzi, M. Tale Masouleh and H. Taghirad, "Workspace analysis of 5-PRUR parallel mechanisms (3T2R)," *Robot. Comput.-Integr. Manuf.* **28**(3), 437–448 (2012).
40. Laboratoire de Robotique de L'Université Laval. Available at: `http://www.robot.gmc.ulaval.ca` (online)
41. I. A. Bonev, "Geometric Analysis of Parallel Mechanisms," Ph.D. dissertation, Laval University, Quebec, QC, Canada (Oct. 2002).
42. J. Sefrioui and C. Gosselin, "On the quadratic nature of the singularity curves of planar three-degree-of-freedom parallel manipulators," *Mech. Mach. Theory* **30**(4), 553–551 (1995).