

RESEARCH ARTICLE

A geometrical formulation for the workspace of parallel manipulators

Matteo Russo^{1,*} and Marco Ceccarelli²

¹Rolls-Royce UTC in Manufacturing and On-Wing Technology, Faculty of Engineering, University of Nottingham, Jubilee Campus, Nottingham NG8 1BB, UK and ²LARM2: Laboratory of Robot Mechatronics, University of Rome “Tor Vergata”, Via del Politecnico 1, Rome 00133, Italy

*Corresponding author. E-mail: matteo.russo@nottingham.ac.uk

Received: 11 October 2021; **Accepted:** 17 November 2021; **First published online:** 13 December 2021

Keywords: workspace, parallel manipulators, kinematics, algebraic workspace, geometric workspace

Abstract

In this study, a geometric formulation is proposed to describe the workspace of parallel manipulators by using a recursive approach as an extension of volume generation for solids of revolution. In this approach, the workspace volume and boundary for each limb of the parallel manipulator is obtained with an algebraic formulation derived from the kinematic chain of the limb and the motion constraints on its joints. Then, the overall workspace of the mechanism can be determined as the intersection of the limb workspaces. The workspace of different kinematic chains is discussed and classified according to its external shape. An algebraic formulation for the inclusion of obstacles in the computation is also proposed. Both analytical models and numerical simulations are reported with their advantages and limitations. An example on a 3-SPR parallel mechanism illustrates the feasibility of the formulation and its efficiency.

1. Introduction

Robotic manipulators with parallel architectures are characterized by the relative motion of two platforms that are connected by two or more independent kinematic chains (“limbs”) [1]. The limbs can have either an identical or different design, and even though most parallel kinematic manipulators are characterized by a single actuator per limb (“fully parallel manipulators”), designs with idle limbs (i.e., that act as a passive constraint, without any actuator) or with two or more motors per limb exist [2]. When compared to serial manipulators, parallel kinematic mechanisms are characterized by a smaller workspace, as they are usually constrained by the mechanical limits of the passive joints; a reduced motion range of the actuators; singularities and assembly conditions that can split the workspace into separated regions; and additional limitations to avoid self-collisions between the different elements of the robot [1, 2]. Furthermore, as the degrees of freedom (DoF) of the robot are coupled, the workspace is often difficult to represent for a direct geometrical interpretation.

A conventional approach for the calculation of the reachable position workspace of parallel manipulators determines a point cloud of reachable poses by iterating kinematics with discretized pose parameters [3, 4]. The conventional approach can be summarized by the following steps: closed kinematic loops are defined for each limb; a loop-closure equation is written for each kinematic loop; the system of equations is solved by using typical geometry such as polygons, prisms, or polyhedrons made with links in specific manipulator configurations to obtain the robot pose as a function of the actuation variables; and finally the reachable poses of the robot are evaluated from a discrete set of the actuation variables, which must be representative of all the possible combinations of the actuation parameters. However, computational methods are inaccurate for large discretization steps or present high computational costs for small ones [5].

Conversely, geometrical approaches can be used to determine workspace boundaries in a fast and accurate way, providing an exact solution to obtain workspace volume and topology [6, 7]. These herein novel approaches are characterized by the generation of a workspace volume for each limb as an open-loop kinematic chain and by the computation of the overall workspace of the manipulator as an intersection of the limb workspace volumes as solids of revolution. Thus, they do not require the numerical or analytical solution of the input-output relationship of the manipulator, which might be difficult to define or to solve, especially for overconstrained or underconstrained structures and singular positions. Geometrical methods are popular for the workspace analysis of planar parallel robots, as their constraints and boundary are easy to define. The solution for planar designs is discussed in ref. [8] for n -DoF designs and expanded in ref. [9] to the study of planar serial-parallel hyper-redundant designs. A similar geometrical approach has been used to evaluate the workspace of spherical parallel manipulators, with early works [10] on symmetric mechanisms and recent ones on asymmetric designs [11].

The complex geometry and constraints of spatial parallel robots poses additional challenges to the determination of their workspace with a geometrical approach. There are solutions for specific lower-mobility designs, such as the 3-DoF double-octahedral architecture in ref. [12], the translational manipulators in refs. [13, 14], and the reconfigurable design in ref. [15], as well as partial solution for 6-DoF robots [16–18]. More recent works report applications to redundant mechanisms, such as the cable-driven manipulator in ref. [19] and the hyper-redundant modular design in ref. [20]. All these solutions refer to a specific design or to a limited family of manipulators. A more inclusive approach is adopted in ref. [21], where the workspace of parallel manipulators is optimized with a geometrical method that is based on CAD simulations rather than algebraic formulations. Further advantages of geometrical methods are highlighted in ref. [22], where two parallel manipulators are analyzed with algebraic and numerical procedures and the results are compared. However, a general algebraic method for the determination of the workspace of a parallel manipulator has not been defined yet despite its advantages, given by an exact algebraic formulation of the boundaries of the reachable position workspace and an efficient computation that only requires the kinematic architecture of the mechanism.

In this study, we attempt a first step towards this result by proposing a systematic procedure, which approaches the problem by determining the workspace boundary of each limb of the robot. This novel workspace determination is formulated by generating workspace volume as an envelope of generating loci due to joint mobility. Then, the manipulator workspace is obtained as the intersection of the computed limb volumes. Finally, an example on a lower-mobility manipulator is reported to validate the proposed method. The results show that algebraic methods can compute a workspace as accurately as methods based on discretized kinematics in a shorter time, from ten to hundreds of times shorter (from 1.20 s to 0.02 in the reported example). This reduction in computation time is particularly significant in optimization problems, such as dimensional synthesis, when the determination of the workspace is iterated over a large number of repetitions with different geometric parameters.

2. Problem formulation

A parallel manipulator consists of two platforms connected by n independent kinematic chains, as illustrated in the scheme in Fig. 1. A fixed reference frame $\{S_0\}$ can be defined on one of the platforms (“base”) of the robot, and a mobile reference frame $\{S_H\}$ describes the pose of the other platform (“end-effector”). In serial architectures, a single kinematic chain determines the pose of the end-effector, which can be described with a homogeneous transformation $T_{0H} \in SE(3)$ from $\{S_0\}$ to $\{S_H\}$ [1]. This transformation defines $R_{0H} \in SO(3)$ as the rotation and $t_{0H} \in R^3$ as the translation from $\{S_0\}$ to $\{S_H\}$, with each joint of the chain controlled by an actuator. Conversely, in parallel manipulators each limb includes both active and passive joints. Because of the presence of idle joints, the pose of a limb cannot be defined by the limb’s motor only but depends on the other limbs.

As it is difficult to decouple the pose of each limb from the others, the inverse and forward kinematic problems are usually defined through a system of loop-closure equations, which is tailored to the specific

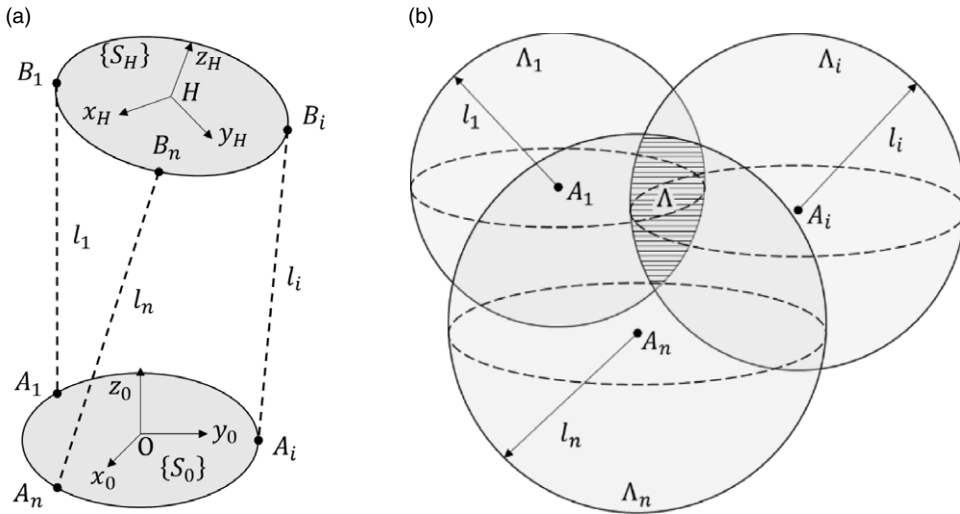


Figure 1. Elements of a parallel manipulators: (a) Kinematic scheme with base, end-effector, and n independent kinematic chains; (b) workspace Λ of a parallel manipulator as intersection of the limb workspaces Λ_i .

architecture of the parallel manipulator under examination [1, 2]. However, when the kinematic chain of the i th limb is known in both topology and constraints, the locus of all the poses Λ_i that describe a possible location of the end-effector (“limb workspace”) can be defined by considering that chain only and neglecting the constraints imposed by the others. From there, the workspace of the parallel manipulator Λ can be obtained from the intersection of the limb workspaces as

$$\Lambda = \bigcap_{i=1}^n \Lambda_i, \tag{1}$$

similarly to the generation of the workspace of n -R serial manipulators [6].

3. Workspace volume generation

In order to obtain the workspace of each limb of a parallel manipulator, conventional techniques [6] can be used as long as the kinematic chain of the limb has a serial architecture. Each kinematic chain consists of a succession of rigid links connected by joints such as revolute (R), prismatic (P), universal (U), and spherical (S) joints. In the notation used in this paper, an active joint (i.e., actuated by a motor) is underlined (e.g., \underline{X}). As outlined in the previous section, the workspace of each limb, given its architecture, can be obtained as a combination of rotations and translations along its links. Each R joint corresponds to a rotation around a single axis, while U joints and S joints result in consecutive rotations around two and three normal axes, respectively; a P joint is defined as a translation along a single axis. In this section, we use the mobility of each joint to determine the limb workspace with examples describing the limb architectures of different popular 3-DoF and 6-DoF fully parallel manipulators with identical limbs. In particular, as illustrated in Fig. 2, we consider the following chains as examples: UPS, PUS, UPU, Delta robot.

3.1. UPS and UPU kinematic chains

Arguably, the most common kinematic chain, the UPS limb, in Fig. 2(a), is used in Gough-Stewart platforms among other architectures, with many applications from flight simulators [2] to high-accuracy

Table I. Nomenclature.

Symbol	Description
Λ	Workspace of the parallel manipulator
Λ_i	Workspace of the i th limb of the parallel manipulator
V_Λ	Workspace volume of the parallel manipulator
V_{Λ_i}	Workspace volume of the i th limb of the parallel manipulator
$\{S_0\}$	Fixed reference frame
$\{S_H\}$	Mobile reference frame
T_{0H}	Homogeneous transformation from $\{S_0\}$ to $\{S_H\}$
R_{0H}	Rotational component of T_{0H}
t_{0H}	Translational component of T_{0H}
i	Limb index
j	Joint index
n	Number of limbs of the parallel manipulator
O	Center of base reference frame
H	Center of end-effector reference frame
A_i	Attachment point of the i th limb on the base
B_i	Attachment point of the i th limb on the end-effector
x_i, y_i, z_i	Coordinates of A_i in the base reference system
$\vartheta_{i,j}$	Motion variable for the j th idle joint of the i th limb
$\vartheta_{i,j,\min}$	Lower limit of the motion variable $\vartheta_{i,j}$
$\vartheta_{i,j,\max}$	Upper limit of the motion variable $\vartheta_{i,j}$
l_i	Motion variable for the linear actuator of the i th limb
$l_{i,\min}$	Lower limit of the motion variable l_i
$l_{i,\max}$	Upper limit of the motion variable l_i
d_i	Geometrical parameter of the i th limb
$Rot_k(\alpha)$	Transformation matrix describing a rotation of α around the k -axis
$Tr_k(a)$	Transformation matrix describing a translation of a around the k -axis
k	Obstacle index
m	Number of obstacles in the workspace
Ω_k	Volume of the k th obstacle

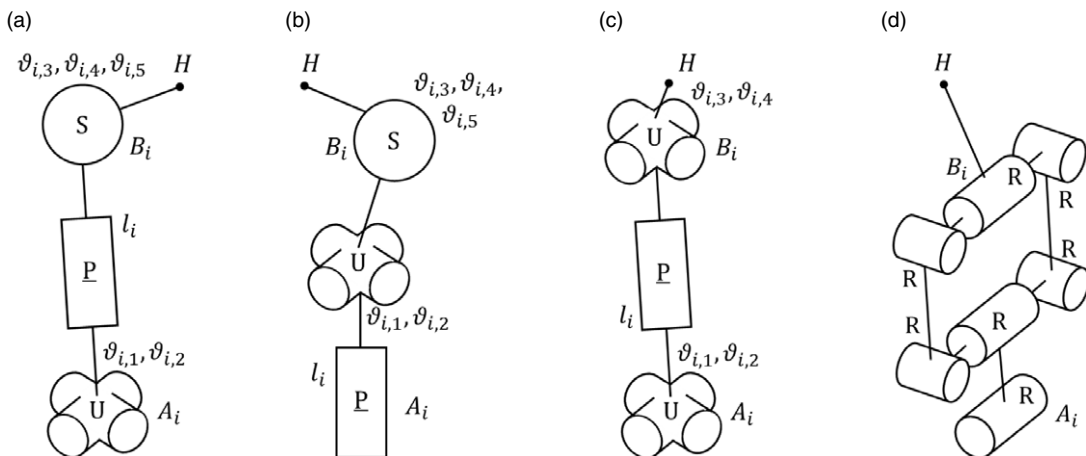


Figure 2. Kinematic schemes of common limb architectures with their motion variables: (a) UPS kinematic chain [22]; (b) PUS kinematic chain [23]; (c) UPU kinematic chain [24]; (d) Delta robot kinematic chain [25].

machining [23]. The workspace of a UPS limb can be obtained by modeling the universal (U) and spherical (S) joints as a succession of two and three revolute (R) joints, respectively. Thus, following the procedure in ref. [4] and with reference to the general scheme in Fig. 1, the workspace of this joint can be described by

$$\Lambda_i = Rot_x(\vartheta_{i,1}) \cdot Rot_y(\vartheta_{i,2}) \cdot Tr_z(l_i) \cdot Rot_x(\vartheta_{i,3}) \cdot Rot_y(\vartheta_{i,4}) \cdot Rot_z(\vartheta_{i,5}), \tag{2}$$

where Rot_k describes a rotation around the k-axis, Tr_k describes a translation along the k-axis, $\vartheta_{i,j}$ are the motion variables for the passive joints of the chain, and l_i is the length of the limb, as controlled by the linear actuator. Each rotational degree of freedom has been numbered sequentially from base to mobile platform, so that indices 1 and 2 correspond to the two rotational degrees of freedom of the universal joint and indices 3, 4, and 5 refer to the three rotational degrees of freedom of the spherical joint.

The passive joints can be characterized by mechanical limits as $\vartheta_{i,j} \in [\vartheta_{i,j,\min}; \vartheta_{i,j,\max}]$, while the limb's actuator makes the limb's length range between its extremities as $l_i \in [l_{i,\min}; l_{i,\max}]$. Therefore, the position workspace of this limb is a family of spheres S (or spherical sectors in case of angular limits), centered on the base extremity of the limb A_i and with radius l_i . This family is defined as

$$S : (x - x_{A_i})^2 + (y - y_{A_i})^2 + (z - z_{A_i})^2 = l_i^2, \tag{3}$$

and its workspace can be obtained as the union of these spheres

$$\Lambda_i = \bigcup_{l_i} S. \tag{4}$$

When no limits are considered for the rotational joints, the volume of this workspace can be determined as

$$V_{\Lambda_i} = \frac{4\pi(l_{i,\max}^3 - l_{i,\min}^3)}{3}. \tag{5}$$

A more general formulation for the volume can be obtained by assuming a symmetric limit for the universal joint ($\vartheta_{i,1,\min} = \vartheta_{i,2,\min}; \vartheta_{i,1,\max} = \vartheta_{i,2,\max}$). Thus, the volume of the workspace can be evaluated from the volume of a spherical sector with a cone angle equal to the angular range of the joint ($\vartheta_{i,1,\max} - \vartheta_{i,1,\min}$), as

$$V_{\Lambda_i} = \frac{2\pi(l_{i,\max}^3 - l_{i,\min}^3)}{3} \left(1 - \cos \frac{\vartheta_{i,1,\max} - \vartheta_{i,1,\min}}{2} \right). \tag{6}$$

Thus, by assuming a constraint on the linear actuator that limits the limb length l_i to range between l_{\min} and l_{\max} , the workspace of the robot can be defined as the volume comprised between two spheres, both with center in A_i and radii of l_{\min} and l_{\max} , respectively.

When considering the reachable position workspace only, by fixing one or more degrees of freedom in the chain, this formulation can be used to model similar kinematic chains that start with a universal joint and a prismatic joint. A third joint allows rotations only (e.g., UPR and UPS architectures). For example, the UPU chain is used as a limb of lower-mobility manipulator structures [19] because of the additional constraint. Its design is shown in Fig. 2(c), while its position workspace is equivalent to the one of a UPS leg as explained in Eq. (2), by imposing $\vartheta_{i,5} = 0$.

3.2. PUS kinematic chain

Obtained from a kinematic inversion of the Gough-Stewart design, this kinematic chain [14, 19], which is shown in Fig. 2(b), is usually characterized by a lower inertia and better performance than the previous one as the motor is not embedded in a mobile part of the robot but has a larger footprint due to the linear actuators on the base. The limb workspace for this architecture can be defined as

$$\Lambda_i = Tr_z(l_i) \cdot Rot_x(\vartheta_{i,1}) \cdot Rot_y(\vartheta_{i,2}) \cdot Tr_z(d_i) \cdot Rot_x(\vartheta_{i,3}) \cdot Rot_y(\vartheta_{i,4}) \cdot Rot_z(\vartheta_{i,5}), \tag{7}$$

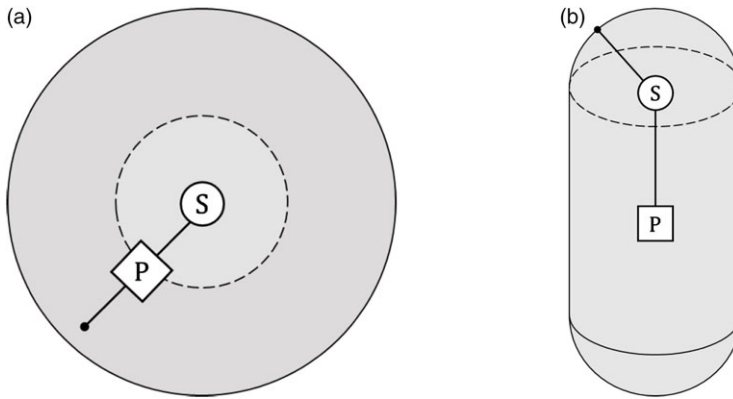


Figure 3. Resulting workspace volumes of common kinematic chains: (a) sphere (with a spherical void) from an unconstrained SP or UP kinematic chain; (b) cylinder with sphere caps from an unconstrained PS or PU chain.

where d_i is a geometrical parameter that describes the length of the rigid link that connects the double revolute (RR) and spherical (S) joints of the kinematic chain. The position workspace of this limb is again a family of spheres S (or spherical sectors in case of angular limits), centered on the base extremity of the limb A_i translated by l_i along the z -axis and with radius d_i . This family is defined as

$$S: (x - x_{Ai})^2 + (y - y_{Ai})^2 + (z - z_{Ai} - l_i)^2 = d_i^2, \tag{8}$$

and its workspace can be obtained as the union of these spheres as in Eq. (4), with a resulting volume of

$$V_{\Lambda_i} = \frac{4\pi d_i^3}{3} + \pi d_i^2 (l_{i,max} - l_{i,min}) \tag{9}$$

when not considering angular limitations.

3.3. Delta robot kinematic chain

The limb of a Delta robot is made of a RR(Pa)R structure, where (Pa) represents a 4R parallelogram, as per Fig. 2(d). This peculiar design results in a pure translational motion of the end-effector, as reported in ref. [2]. Its workspace can be described again as a family of spheres, with an envelope coming from two solid spheres with equal radius and a solid cylinder with the same radius connecting them, similar to the one observed for the PUS kinematic chain. When the centers of the two spheres are at a distance of twice their radius or less, there is a void in the enveloping solid in the volume of their intersection, as shown in the detailed formulation and volume in ref. [26].

4. Workspace analysis

As explained in the previous section, the limb workspaces of fully parallel manipulators with at least two rotational joints (i.e., a wide majority of the existing parallel mechanisms) can be usually geometrically defined by a family of spheres, or spherical sectors, with the actuation variable as family parameter, and the overall workspace of the manipulator can be found as the intersection of these limb workspaces.

A first estimation of the workspace of the robot can be obtained from the external shape of the limb workspace, which is characterized by a strong dependence on the first joints in the limb’s kinematic chain, as shown in the examples in Fig. 3. Three types of kinematic chains can be identified, according to their outer workspace boundary shape:

- Type A: Limbs starting with joints that allow two or more rotations. In general, limb chains starting with two or more rotational degrees of freedom will have a spherical surface as external volume boundary, with subsequent linear and rotational degrees of freedom modifying the sphere's radius, as in Fig. 3(a).
- Type B: Limbs with a prismatic joint as first or second joint of their kinematic chains. Chains with a linear degree of freedom as first or second degree of freedom of the kinematic chain will result in a cylindrical outer surface, with hemispheres on the top and bottom faces (Fig. 3(b)).
- Type C: Limbs beginning with multiple prismatic joints with non-parallel axes. Consecutive linear degrees of freedom are rarely used in parallel mechanisms; however, in case two or more prismatic joints (with normal or anyway distinct axes of motion) are at the beginning of a limb chain, the resulting workspace is characterized by a parallelepipedal shape (with rounded corners and edges in case of subsequent rotational degrees of freedom).

Applying limits to the linear and rotational joints constraints these volumes further, depending on their surface type:

- Type A: Linear constraints to the prismatic joints result into a maximum (and minimum) radius for the spherical surface swept by the end-effector. Angular constraints reduce the sphere to a spherical sector.
- Type B: Linear constraints on the first prismatic joint define the height of the cylindrical volume. Linear constraints on subsequent prismatic joints modify the maximum (and minimum) radius of the cylinder, as well as the one of the hemispheres. Angular constraints on rotational degrees of freedom change the hemispheres to spherical caps and affect the height of the resulting domes.
- Type C: Linear constraints on the prismatic joints define the side lengths of the parallelepiped. Angular constraints on subsequent rotational joints, together with the length of the related links, define fillet radius and incidence angle of the rounded corners and edges.

By identifying the type of each kinematic chain and considering joint limits, the workspace volume can be easily visualized and represented. Furthermore, these types can be used to classify parallel manipulators according to their limb characteristics, in order to evaluate the suitability of a given architecture for a desired workspace according to the workspace's features. For example, when working on tasks with a principal direction, Type B limbs with parallel linear actuators as first joint of each limb might be favored, while other tasks on axial-symmetric or near-planar workspaces might be reached better by Type A limbs or Type B limbs with concurrent linear actuators as first joint of each limb. Hybrid manipulators with limbs of different types can be also considered for specific tasks.

Another key information when analyzing the workspace of a parallel manipulator is given by the volume that cannot be reached by the manipulator, either to avoid the collision with an obstacle or for any other kind of hazard. A limitation of the proposed procedure is given by internal obstacles, such as self-collision among limbs, which cannot be implemented directly with the proposed procedure without solving the kinematics of the mechanism. This is a consequence of the proposed two-step procedure, which computes the limb workspaces independently and then generates the overall volume as an intersection. In this way, no information on limb placement for each point of the workspace can be carried over into the intersection operation.

Conversely, external obstacles can be easily implemented in the proposed formulation with a volume subtraction, which can be expressed from (1) as a difference with set algebra as:

$$\Lambda = \bigcap_{i=1}^n \Lambda_i - \bigcap_{k=1}^m \Omega_k, \tag{10}$$

where Ω_k represents the volume occupied by the k th obstacle and m is the number of obstacles.

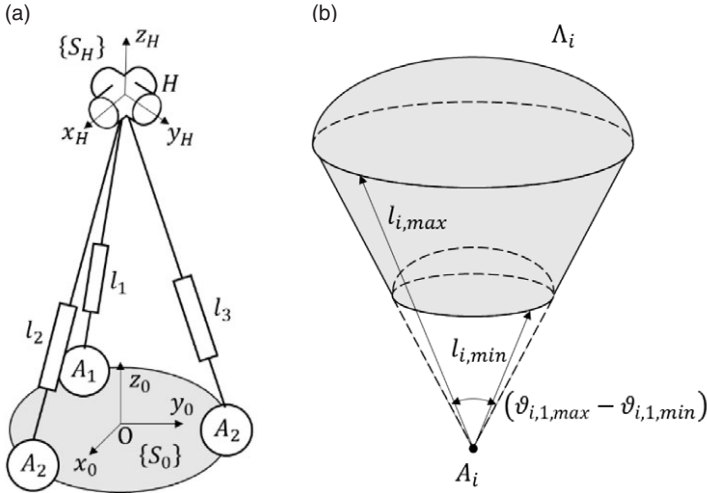


Figure 4. Modeling of a 3-SPR parallel manipulator: (a) Kinematic diagram of the mechanism with main geometrical parameters; (b) limb workspace evaluated with the proposed geometrical method.

5. An example: 3-SPR mechanism

The procedure introduced in the previous section is here validated with a numerical example on a 3-SPR parallel manipulator, in Fig. 4(a). The design and kinematics of this mechanism are detailed in ref. [27], where the reachable position workspace of the mechanism is computed by using forward kinematics through a discretization of the range of motion of each linear actuator. In this section, a geometrical formulation of the workspace is used, and we compare its results and efficiency (measured as computational time to solution) with the discrete formulation used in ref. [27]. Its limb workspace, in Fig. 4(b), can be seen as a subset of the UPU case reported in Section 3, as the 3-SPR mechanism is an overconstrained variant of a conventional 3-UPU manipulator which requires the end-effector to collapse into a single point to enable motion [27].

In this example, joint limits are considered for prismatic and spherical joints. The minimum and maximum stroke of the linear actuator is defined as an inequality constraint that limits the limb length l_i to range between l_{min} and l_{max} . Thus, the workspace of the robot can be defined as the volume comprised between two spheres, both with center in A_i and radii of l_{min} and l_{max} , respectively. The angular limit for the base spherical joint is given as $[-\pi/2; +\pi/2]$ in each axis and is implemented as shown in Eq. (6). By considering limb volumes Λ_i as per Eq. (5), the overall workspace of the manipulator Λ can be evaluated as per Eq. (1). The corresponding workspace volume V_Λ can be evaluated analytically from the formulation of the volume of intersection of three spheres presented in ref. [28], as

$$V_\Lambda = \prod_{i=1}^3 V_{S_{i,max}} - \prod_{i=1}^3 V_{S_{i,min}}, \tag{11}$$

where $V_{S_{i,max}}$ is the volume of $S:(x - x_{Ai})^2 + (y - y_{Ai})^2 + (z - z_{Ai})^2 = l_{max}^2$, and $V_{S_{i,min}}$ is the volume of $S:(x - x_{Ai})^2 + (y - y_{Ai})^2 + (z - z_{Ai})^2 = l_{min}^2$. For the formulation in ref. [28], the geometry described in ref. [27] is considered for the manipulators, with each base joint at a vertex of an equilateral triangle with side d , and $l_{min} > d$. Thus, the volume can be explicitly expressed as

$$V_{\Lambda,max} = \frac{d\sqrt{3l_{max}^4 - d^4}}{6} - \frac{d(12l_{max}^2 - d^2)}{4} \arccos \frac{d\sqrt{3}}{\sqrt{4l_{max}^2 - d^2}} + 2l^3 \arccos \frac{2l_{max}^2 - d^2}{-4l_{max}^2 + d^2} \tag{12a}$$

$$V_{\Lambda,min} = \frac{d\sqrt{3l_{min}^4 - d^4}}{6} - \frac{d(12l_{min}^2 - d^2)}{4} \arccos \frac{d\sqrt{3}}{\sqrt{4l_{min}^2 - d^2}} + 2l^3 \arccos \frac{2l_{min}^2 - d^2}{-4l_{min}^2 + d^2} \tag{12b}$$

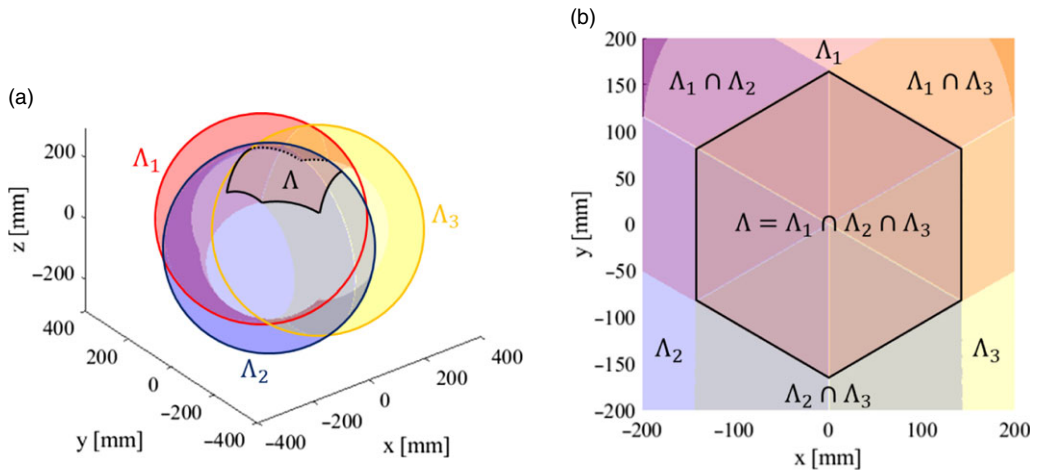


Figure 5. Workspace Λ of a 3-SPR parallel manipulator evaluated by using the proposed geometrical approach as intersection of the limb workspaces Λ_i : (a) Side view of the three spherical limb workspaces, outlined in blue, red and yellow, and the overall workspace of the manipulator given by their intersection, outlined in black; (b) top view of a close-up on the intersection of the limb workspaces, outlined in black.

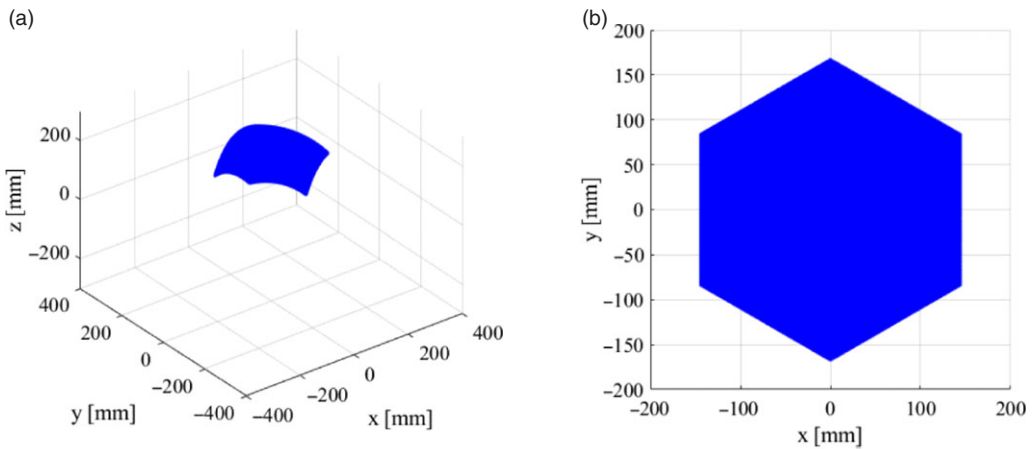


Figure 6. Workspace Λ of a 3-SPR parallel manipulator computed as a point cloud of reachable locations by discretizing pose parameters as per [26]: (a) side view; (b) top view.

$$V_{\Lambda} = V_{\Lambda, \max} - V_{\Lambda, \min}. \tag{12c}$$

Both the proposed geometrical method and a discretized approach [27] have been used to compute the workspace volume and shape of a 3-SPR manipulator with $l_{i, \min} = 200$, $l_{i, \max} = 300$, $d = 100$. The calculation has been implemented in MATLAB on a Windows 10 computer with a 2.60 GHz quad-core CPU (Intel i7-6700HQ). The results of the geometrical approach are reported in Fig. 5, in which the boundaries of the three limb workspaces (in red, yellow, and blue) are shown with the boundary of their intersection (in black), whereas the discrete workspace computed as a point cloud of 106 reachable locations is shown in Fig. 6.

The geometrical method provides an exact solution in less than 0.02 s, while the numerical approach returns an approximated result after 1.20 s. The two computed workspaces perfectly overlap. The volume obtained with the algebraic geometric approach is the geometrically exact one, as per Eq. (9) as reported

also in ref. [27], whereas the value computed with the discretized approach comes from a voxelization of the workspace and therefore includes an approximation error. Thus, the proposed method proves to be more efficient and better than the numerical one when determining the workspace volume of a parallel manipulator with given joint constraints.

6. Conclusion

In this paper, a systematic approach is proposed for the evaluation of the workspace of parallel mechanisms through a geometrical method that determines the boundary of the workspace of each limb when considered as an isolated kinematic chain and then obtains the overall workspace as an intersection of the limb workspace volumes. The method is validated with a numerical example that compares the proposed method to a discrete one, highlighting the efficiency of the proposed algebraic formulation in achieving the same result. Moreover, when compared to previous methodologies, the proposed formulation does not require previous knowledge of the kinematics of the robot, but only the structure of the analyzed mechanism. Thus, the proposed method is suited to procedures where the workspace of a class of mechanisms is computed multiple times, such as optimization algorithms, thanks to its low computational time, and preliminary workspace analysis of complex mechanisms without the need of a full kinematic model. However, the proposed geometric approach only considers joint motion limits and does not model self-collision, as well as being limited to the reachable position workspace of the manipulator without including the orientation. By addressing both aspects in future developments, the proposed method can become a feasible and more efficient alternative to conventional ones.

Author contributions. MR and MC conceived, designed the study, performed simulations, and wrote the article.

Financial support. This research received no specific grant from any funding agency, commercial, or not-for-profit sectors.

Conflicts of interest. The authors declare none.

Ethical considerations. The authors declare none.

References

- [1] M. Ceccarelli, *Fundamentals of Mechanics of Robotic Manipulation* (Springer, Cham, 2004).
- [2] L. W. Tsai, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators* (John Wiley & Sons, New York, 1999).
- [3] D. Y. Jo and E. J. Haug, "Workspace Analysis of Closed Loop Mechanisms with Unilateral Constraints," *ASME Design Automation Conference*, Montreal, September, 17–20 (1989) pp. 53–60.
- [4] H. Guo, C. Zhao, B. Li, R. Liu, Z. Deng and J. Tian, "A transformation method to generate the workspace of an n (3rrs) serial-parallel manipulator," *J. Mech. Des.* **141**(9), 092301 (2019).
- [5] W. Sirichotiyakul, V. Patoglu and A. C. Satici, "Efficient singularity-free workspace approximations using sum-of-squares programming," *J. Mech. Rob.* **12**(6), 061004 (2020).
- [6] M. Ceccarelli, "A formulation for the workspace boundary of general N-revolute manipulators," *Mech. Mach. Theory* **31**(5), 637–646 (1996).
- [7] F. Freudenstein and E. J. F. Primrose, "On the analysis and synthesis of the workspace of a three-link, turning-pair connected robot arm," *J. Mech. Trans. Autom.* **106**(3), 365–370 (1984).
- [8] T. Li and M. Ceccarelli, "A Geometric Approach for Workspace Determination of Planar n-DOF Parallel Manipulators," 4th International Congress on Design and Modeling of Mechanical Systems (2011).
- [9] S. Yahya, M. Moghavvemi and H. A. Mohamed, "Geometrical approach of planar hyper-redundant manipulators: Inverse kinematics, path planning and workspace," *Simul. Model. Pract. Theory* **19**(1), 406–422 (2011).
- [10] F. Bulca, J. Angeles and P. J. Zsombor-Murray, "On the workspace determination of spherical serial and platform mechanisms," *Mech. Mach. Theory* **34**(3), 497–512 (1999).
- [11] A. Alamdar, F. Farahmand, S. Behzadipour and A. Mirbagheri, "A geometrical approach for configuration and singularity analysis of a new non-symmetric 2DOF 5R spherical parallel manipulator," *Mech. Mach. Theory* **147**(1), 103747 (2020).
- [12] V. Arun, B. Padmanabhan, K. Kolady and C. F. Reinholtz, "Determination of the Workspace of the 3-DOF Double-Octahedral Variable Geometry Truss Manipulator," *Proceedings of the 22nd ASME Biennial Mechanisms Conference* (ASME, Phoenix, AZ, 1992) pp. 493–500.

- [13] R. Di Gregorio and R. Zangorini, “Workspace analytic determination of two similar translational parallel manipulators,” *Robotica* **21**(5), 555 (2003).
- [14] S. Perreault, P. Cardou, C. M. Gosselin and M. J. D. Otis, “Geometric determination of the interference-free constant-orientation workspace of parallel cable-driven mechanisms,” *J. Mech. Rob.* **2**(3), 031016 (2010).
- [15] L. Nurahmi and D. Gan, “Reconfiguration of a 3-(rR) PS metamorphic parallel mechanism based on complete workspace and operation mode analysis,” *J. Mech. Rob.* **12**(1), 011002 (2020).
- [16] D. J. Kim, W. K. Chung and Y. Youm, “Geometrical Approach for the Workspace of 6-DOF Parallel Manipulators,” *Proceedings of International Conference on Robotics and Automation*, vol. 4 (1997) pp. 2986–2991.
- [17] I. A. Bonev and J. Ryu, “A geometrical method for computing the constant-orientation workspace of 6-PRRS parallel manipulators,” *Mech. Mach. Theory* **36**(1), 1–13 (2001).
- [18] X. Wang and W. Guo, “Chasles’ motion sphere—A useful tool to visually express the motion characteristics over the workspace of robotic mechanisms,” *J. Mech. Rob.* **12**(4), 041008 (2020).
- [19] M. M. Aref and H. D. Taghirad, “Geometrical Workspace Analysis of a Cable-Driven Redundant Parallel Manipulator: KNTU CDRPM,” 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (2008) pp. 1958–1963.
- [20] B. Hu, H. Cui, D. Shi, D. Zhang, A. Wang, Y. Wang and Q. Zhang, “Reachable workspace determination for a spatial hyper-redundant manipulator formed by several parallel manipulators,” *J. Mech. Sci. Technol.* **33**(2), 869–877 (2019).
- [21] B. Aboulissane, D. El Haiek, L. El Bakkali and J. El Bahaoui, “On the workspace optimization of parallel robots based on CAD approach,” *Procedia Manuf.* **32**(1), 1085–1092 (2019).
- [22] M. Russo and M. Ceccarelli, “A Comparison of Algebraic and Iterative Procedures for the Generation of the Workspace of Parallel Robots,” In: *Mechanism Design for Robotics, Mechanisms and Machine Science*, vol. 103 (Springer, Cham, 2021) pp. 53–61.
- [23] M. Russo and X. Dong, “A calibration procedure for reconfigurable Gough-Stewart manipulators,” *Mech. Mach. Theory*, **152**(1), 103920 (2020).
- [24] Z. Baigunchekov and R. Kaiyrov, “Direct kinematics of a 3-PRRS type parallel manipulator,” *Int. J. Mech. Eng. Rob. Res.* **9**(7), 967–972 (2020).
- [25] L. W. Tsai and S. Joshi, “Kinematics and optimization of a spatial 3-UPU parallel manipulator,” *J. Mech. Des.* **122**(4), 439–446 (2000).
- [26] X. J. Liu, J. Wang, K. K. Oh and J. Kim, “A new approach to the design of a DELTA robot with a desired workspace,” *J. Intell. Rob. Syst.* **39**(2), 209–225 (2004).
- [27] M. Russo, M. Ceccarelli and Y. Takeda, “Force transmission and constraint analysis of a 3-SPR parallel manipulator,” *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **232**(23), 4399–4409 (2017).
- [28] L. S. Chkhartshvili, “Volume of the intersection of three spheres,” *Math. Notes* **69**(3–4), 421–428 (2001).