A novel method for direct kinematics solution of fully parallel manipulators using basic regions theory

K Kamali and **A Akbarzadeh***
Mechanical Engineering Department, Ferdowsi University of Mashhad, Mashahd, Iran

The manuscript was received on 10 November 2009 and was accepted after revision for publication on 17 September 2010.

DOI: 10.1177/2041304110394494

Abstract: Model-based controllers potentially offer a higher positioning accuracy for robotic systems. The direct kinematics solution is an essential part of these controllers. However, the direct kinematics problem for parallel manipulators is usually very complicated and in general does not have a closed-form solution. This problem usually leads to multiple solutions. When a control application or dynamics simulation is considered, only one of the solutions is acceptable. Although there are some numerical methods that obtain all possible solutions, specifying the one acceptable solution among them is still a challenging problem. It is shown that the tool space of a parallel manipulator with one inverse kinematic solution can be categorized into special regions called basic regions. In this paper, a new concept for basic regions is proposed that extends the concept to non-cuspidal fully parallel manipulators with multiple inverse and multiple direct kinematics solutions. Then, for general non-cuspidal fully parallel manipulators, a numerical algorithm is proposed that determines the basic region domains in the tool space. Finally, a novel method is proposed which utilizes the basic regions theory to determine which direct kinematics solution is acceptable. The proposed method presents a general solution to the direct kinematics problem of non-cuspidal parallel manipulators in trajectory following. The provided solutions are reliable and can be refined up to an arbitrary accuracy. The proposed method is illustrated using a 3-RRR planar parallel manipulator.

Keywords: direct kinematics, parallel manipulator, basic regions, multiple solutions

1 INTRODUCTION

Many researchers have investigated the kinematics of parallel manipulators. Kinematics problems can be thought of as belonging to one of two different branches: direct kinematics problems and inverse kinematics problems. Obtaining a solution to a direct kinematics problem is an essential part in model-based control and dynamics simulation of parallel manipulators [**1**, **2**]. Model-based controllers are those that use the dynamics of a manipulator. As a result, these controllers are more accurate and reliable. Furthermore, dynamics simulation is a common utility for design and optimization of the manipulator and controller before building the physical robot and its controller. In both model-based control and simulation applications, there is a data stream of joint values (joint space vector) which should be converted to orientations and positions of the end effector (tool space vector). In the control case, the data stream is provided as sensor feedback in the joint space. Using direct kinematics, in each iteration one joint space vector is converted to one tool space vector.

Obtaining the direct kinematics of a parallel manipulator is a challenging problem because of its complexity [**3–6**]. It involves the resolution of a system of non-linear equations whose solution may not be unique. Moreover, in general, such a problem does not admit closed-form solutions and numerical algorithms need to be used. Normally, multiple solutions are found to the direct kinematics problem [**3**] and therefore multiple branches of direct kinematics solutions (assembly modes) exist. However, in a trajectory-following application, only one of these solutions is the acceptable solution.

Numerous approaches have been implemented in order to find the solution to direct kinematics problems [**7–9**]. There are numerical iterative methods

**Corresponding author: Mechanical Engineering Department, Ferdowsi University of Mashhad, Mashahd, Iran.*
email: Ali_Akbarzadeh_T@yahoo.com

which are convenient for finding one of the direct kinematics solutions and use an estimated point as the initial guess. Most of these methods such as: the 'iterative method with kinematic Jacobian', 'iterative method with Euler angle's Jacobian matrix', and 'reduced iterative method' [**10**] are based on the classic Newton's method. If the initial guess is chosen near the acceptable solution, these methods may eventually converge to it. In applications such as trajectory following, simulation, or control applications, the manipulator's prior position may be used as the initial guess. If a good initial guess is made, the Newton-based methods are usually fast enough to be used in online control applications. One would expect that Newton-based schemes will converge if a good initial estimate is made and that this convergence will lead to the solution that is the closest to the initial guess. However, it has been shown that both of these assumptions are not always correct [**11**]. These problems create an important reliability problem. For example, in a control application, an incorrect answer will result in incorrect control.

Another approach is that due to Siciliano [**12**] who proposed the closed-look direct kinematics (CLDK) algorithm to solve the direct kinematics problem associated with a trajectory-following application. This algorithm uses robot differential kinematics (Jacobian matrix) to estimate the direct kinematics solution. The convergence of the method was shown to be ensured through the stability of a closed-loop dynamic system (such as closed-loop control) of the tracking error. However, a close-loop condition is not always available.

All the discussed methods have some drawbacks which make them either inapplicable or unsafe to use. Another approach is to compute all possible solutions of the direct kinematics for given joint values. This approach consists of methods such as: 'the elimination method', 'the continuation method', 'the Gröebner basis method', and 'interval analysis' [**13**]. These methods do find all the possible solutions with an arbitrary accuracy. However, the problem is that to date no one has proposed an algorithm which allows determination of the acceptable solution. Therefore, this remains a challenging kinematics problem.

It is well known that parallel manipulators have singularities in their workspace where stiffness is lost [**3**]. These singularities coincide with the set of configurations in the workspace where two direct kinematics solutions meet [**14**]. There used to be common thought that if the end effector of a robot did not cross singular points, no change would happen in the branch of the solution. This belief used to be used as a basis for determining the acceptable solution [**1**]. The method says that if a manipulator does not pass a singular point then there will be no change in the branch of acceptable solution; therefore, the solution in the same branch of the start point is the acceptable solution. However, it was shown by Innocenti and Parenti-Castelli [**11**] that in a planar robot, two different direct kinematics solutions may be connected through a singularity-free trajectory in the tool space of the robot. In other words, the change in the branches of the direct kinematics solutions (assembly modes) could also be accomplished without passing through a singularity. As a result, designing an algorithm for a complete direct kinematic verification is difficult and proving that it will lead to the acceptable solution is still an open problem.

The work in reference [**11**] gave rise to a theoretical work that introduced the concepts of characteristic surfaces and uniqueness domains (basic regions) in the workspace [**15**]. Wenger and Chablat [**15**] defined characteristic surfaces for parallel manipulators which have only one inverse kinematic solution. These surfaces, divide the tool space of parallel manipulators into basic regions. It was shown that if the manipulator's tool crosses a characteristic surface then the direct kinematics branches will change. Subsequently, Wenger and Chablat [**16**] studied, using a typical example, the distribution of the different assembly modes in the workspace and their effective role in the trajectory planning application. The singular and non-singular changes of assembly modes were described and compared.

Despite previous studies on the basic regions theorem and the direct kinematics problem of parallel manipulators, there still remain open problems, including:

(a) a basic regions concept for parallel manipulators with multiple inverse kinematics solutions has not been studied;
(b) a general algorithm that numerically finds the basic region in the workspace has not been proposed;
(c) the application of the basic regions theory to find the acceptable direct kinematics solution among multiple possible solutions has not been attempted.

This paper seeks to rectify this situation in that these three problems are addressed and solutions are proposed. The first two contributions are used in the third to propose a method for solving the direct kinematics problem which is more accurate and reliable than previous works in the literature.

This paper is organized as follows. First, a new concept of basic regions is proposed that extends the previous research to non-cuspidal fully parallel manipulators with multiple inverse and multiple direct kinematics solutions. Second, a general numerical algorithm is proposed to determine the basic regions in the tool space. Finally, a novel method is proposed which identifies the acceptable solution among multiple solutions of direct kinematics. In all iterations the proposed method uses the information of the previous iteration to first determine if the end effector is near the borders of the basic regions. Next, depending on the position of the robot's end effector various strategies are proposed to determine the acceptable direct kinematics solution.

To demonstrate the proposed ideas, a 3-RRR planar parallel manipulator is selected. Basic regions in the workspace and solution branches are shown. Finally, given a trajectory in the workspace, the proposed method for solving the direct kinematics is presented.

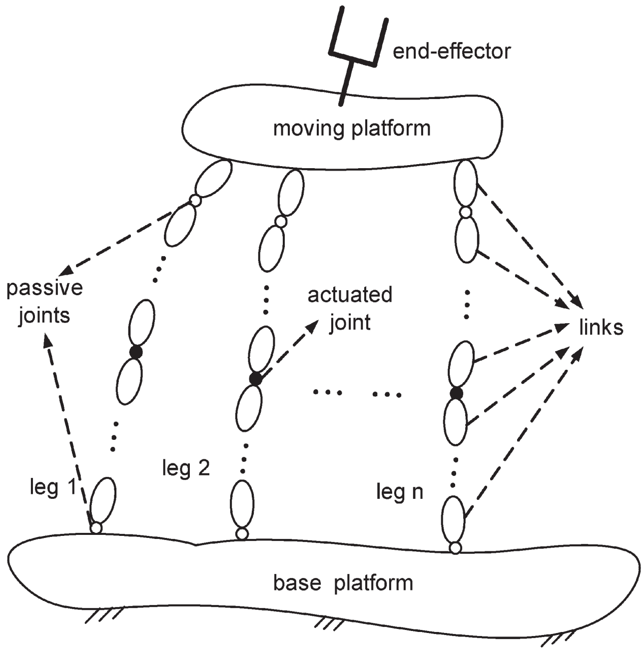

Fig. 1 A general fully parallel manipulator

2 KINEMATICS OF PARALLEL MANIPULATORS

Kinematic analysis of a manipulator consists of studying the relations between position, velocity, and acceleration of different parts of the manipulator. In this paper fully parallel manipulators with non-cuspidal legs are considered. A fully parallel manipulator is a mechanism that includes as many elementary kinematic chains as the number of degrees of freedom of the mobile platform. Moreover, every elementary kinematic chain possesses only one actuated joint (prismatic or revolute). Also, no segment of an elementary kinematic chain can be linked to more than two bodies [**15**]. A general fully parallel manipulator is shown in Fig. 1.

2.1 Kinematic relations

For a manipulator, there is a relation which connects relative values of displacement or angle (dependent on actuator's type) of the actuated joints ($\boldsymbol{q}^{\mathrm{a}}$) to the displacement and angle of the moving platform ($\boldsymbol{X}$) [**15**]

$$F(\boldsymbol{X}, \boldsymbol{q}^{\mathrm{a}}) = 0 \tag{1}$$

where

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{\Theta} \end{bmatrix} \tag{2}$$

where $\boldsymbol{p}$ is the three-dimensional position vector of the end effector and $\boldsymbol{\Theta}$ is the three-dimensional orientation vector of the moving platform. This definition can be applied to serial or parallel manipulators. Differentiating equation (1) with respect to time leads to the velocity model [**3**]

$$\mathbf{J}\dot{\boldsymbol{X}} + \mathbf{K}\dot{\boldsymbol{q}}^{\mathrm{a}} = 0 \tag{3}$$

where $\mathbf{J}$ and $\mathbf{K}$ are the direct-kinematics and the inverse-kinematics matrices of the manipulator, respectively. A singularity condition occurs whenever $\mathbf{J}$ or $\mathbf{K}$ (or both) can no longer be inverted. Three types of singularities exist [**3**]

$$\begin{cases} \text{type-1} \quad \text{singularity}: \ \det(\mathbf{J}) = 0 \\ \text{type-2} \quad \text{singularity}: \ \det(\mathbf{K}) = 0 \\ \text{type-3} \quad \text{singularity}: \ \det(\mathbf{J}) = 0 \text{ and } \det(\mathbf{K}) = 0 \end{cases} \tag{4}$$

2.2 Direct and inverse kinematics problems

Kinematic analysis of parallel manipulators includes the solution to both direct and inverse kinematics problems as well as velocity and acceleration inversion. As in the case of serial manipulators, the direct kinematics problem is defined as the one in which the coordinates (position and orientation) of the end effector are obtained from the actuated joint angles or displacements. The inverse kinematics problem is therefore the one in which the actuated joint angles or displacements are obtained from the coordinates of the end effector [**3**].

A well known feature of parallel manipulators is the existence of multiple solutions to the direct kinematics problem. That is, the mobile platform (end effector) can admit several configurations in the tool space for a given set of input joint values. Moreover, parallel manipulators can have multiple inverse kinematic solutions. This means that there are several input joint values corresponding to a given configuration of the end effector [**3**].

2.3 Postures and assembly modes

Postures and assembly modes are introduced to help define different configurations and they represent inverse and direct kinematics solutions, respectively [**15**]. Different inverse kinematics solutions for a parallel manipulator result in different configurations. These different configurations are called postures. Similarly, multiple direct kinematics solutions for a parallel manipulator results in different configurations. These different configurations are called assembly modes. (Examples of assembly modes will be presented in Figs 11 and 12.)

Cuspidal manipulators are defined for serial manipulators, which can change their posture without passing any singularity [**17**]. In this study, non-cuspidal parallel manipulators are defined as parallel manipulators with none of their legs, as a serial structure, being cuspidal. A list of the most current non-cuspidal serial chains is given in reference [**18**].

Theorem 1

For non-cuspidal parallel manipulators, if the end effector does not pass any singularity, the posture of the manipulator cannot change. A proof is provided in reference [**17**].

This paper considers non-cuspidal fully parallel manipulators.

2.4 Direct kinematics problem

The direct kinematics problem is the problem of determining the pose of the end effector of a parallel manipulator from its actuated joint coordinates. This relation has a clear practical interest for model-based control and dynamics simulation of parallel manipulators [**1**, **2**]. A general model-based control diagram for a parallel manipulator is shown in Fig. 2. The figure shows that in a model-based controller, the direct kinematics need to be solved in an online process. The dynamics of a parallel manipulator can be expressed as follows [**19**]

$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}^{\mathrm{a}} + \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}^{\mathrm{a}} + \mathbf{G}(\boldsymbol{q}) \tag{5}$$

where $\boldsymbol{\tau}$ is the vector of torques or forces provided by the actuators, **M**, **C**, and **G** are the inertia, Coriolis, and gravity matrices, respectively, and $\boldsymbol{q}$ is the vector of all joint coordinates (displacement or orientation) consisting of actuated joint coordinates ($\boldsymbol{q}^{\mathrm{a}}$) and unactuated joint coordinates ($\boldsymbol{q}^{\mathrm{u}}$) and

$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{q}^{\mathrm{a}} \\ \boldsymbol{q}^{\mathrm{u}} \end{bmatrix} \tag{6}$$

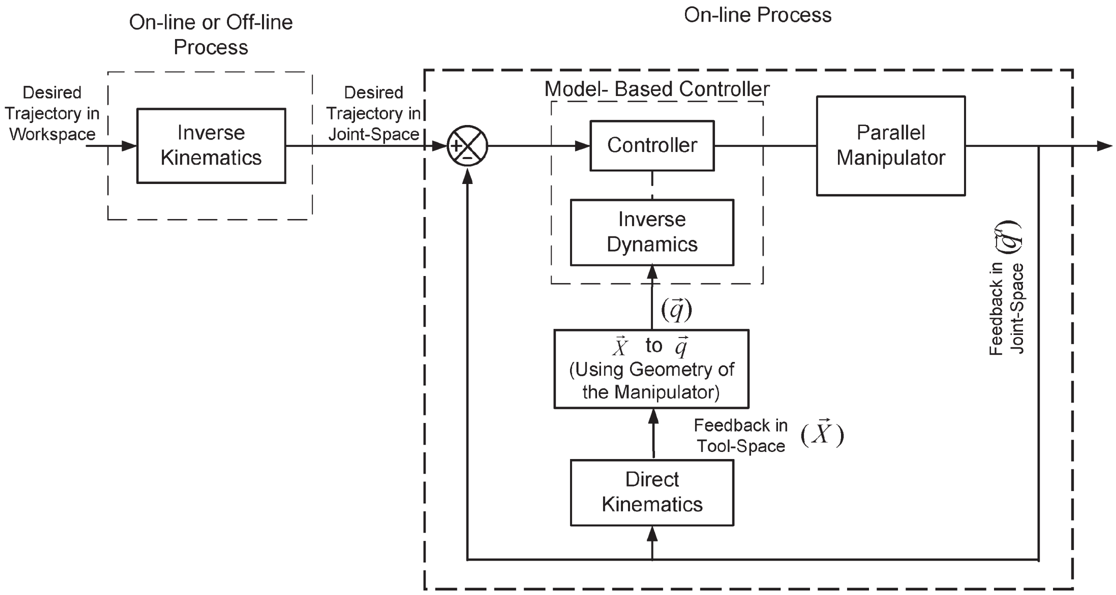

Fig. 2 A typical model-based control process for a parallel manipulator

In order to calculate matrices **M**, **C**, and **G**, one must know the value of $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$. Normally, sensors provide values for $\boldsymbol{q}^{\mathrm{a}}$ and thus using differentiation $\dot{\boldsymbol{q}}^{\mathrm{a}}$ can be easily obtained. There is also a linear relationship between $\dot{\boldsymbol{q}}^{\mathrm{a}}$ and $\dot{\boldsymbol{q}}$ that makes the calculation of the value of $\dot{\boldsymbol{q}}$ straightforward. This relation is modelled with $\dot{\boldsymbol{q}} = \mathbf{L}\boldsymbol{q}^{\mathrm{a}}$, where the **L** matrix is a function of $\dot{\boldsymbol{q}}$ [**19**]. The problem is the calculation of $\dot{\boldsymbol{q}}$ which contains both active and passive joint values. To do this, first, the direct kinematics problem should be solved to obtain $\boldsymbol{X}$. Next, using the geometry of the manipulator, $\dot{\boldsymbol{q}}$ can be obtained from $\boldsymbol{X}$. The need for a reliable and accurate enough method to solve direct kinematics is obvious.

The application of direct kinematics in a dynamic simulation is shown in Fig. 3. Dynamic simulation of a parallel manipulator consists of numerical calculation of $\dot{\boldsymbol{q}}^{\mathrm{a}}$ given $\boldsymbol{\tau}$ as a function of time. This problem is called the inverse dynamics problem and can be derived from equation (5) as follows [**19**]

$$\ddot{\boldsymbol{q}}^{\mathrm{a}} = \mathbf{M}^{-1}(\boldsymbol{q})[\boldsymbol{\tau} - \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}^{\mathrm{a}} - \mathbf{G}(\boldsymbol{q})] \tag{7}$$

Equation (7) is an ordinary differential equation. There are various methods to solve these kinds of equations such as the Runge–Kutta technique [**20**]. A block diagram of this equation in the Laplace domain is shown in Fig. 3 which demonstrates the application of direct kinematics.

In general, the solution for the direct kinematics problem of a parallel manipulator is not unique, i.e. there are several ways of assembling a parallel manipulator with given actuated joint coordinates. Furthermore, generally it is not possible to express the end effector's coordinates as a function of the actuated joint coordinates in an analytical manner.

As previously mentioned, iterative methods such as Newton-based methods [**1**] are usually suggested to solve direct kinematics problems. These methods provide only a single solution which is dependent on the initial guess and may not be an acceptable solution. Therefore, other methods should be considered. These methods include: the elimination method, the continuation method, the Gröebner basis method, and interval analysis [**13**] which find all possible solutions to the direct kinematics problem.

In robotics, many problems including the direct kinematics of a parallel manipulator may be formulated in terms of a set of polynomial equations. Among the methods that may be used to solve this set of polynomial equations, Bezout's elimination method allows the problem to be reduced from having to solve a system of n-equations to solving a univariate polynomial equation [**21**, **22**]. There are numerical approaches to find all roots of a univariate polynomial equation [**23**]. Consequently, all possible solutions of the direct kinematics problem are calculated. Upon obtaining all possible solutions to the direct kinematics problem, an additional method still needs to be utilized to determine the acceptable solution.

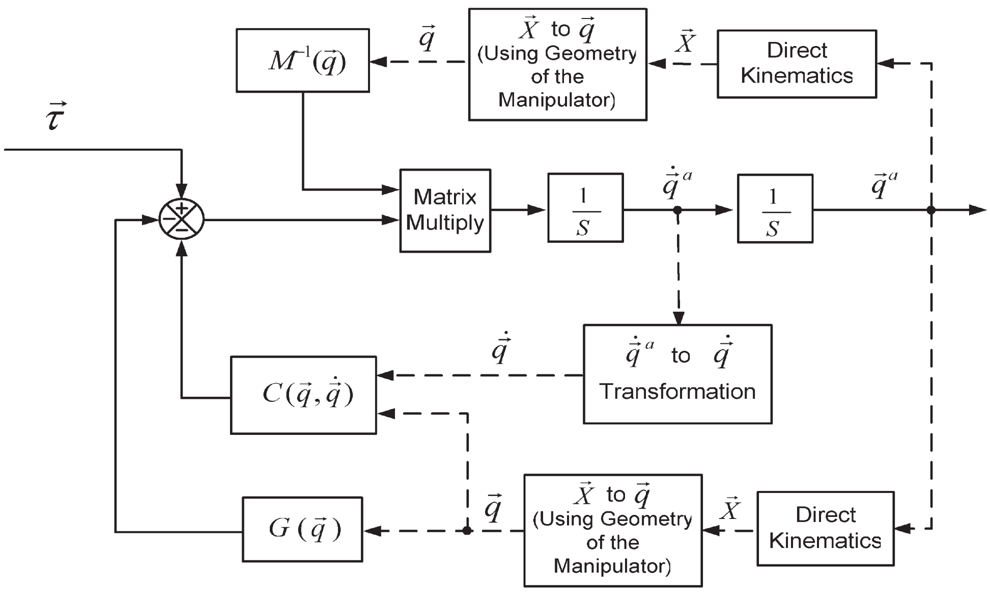

Fig. 3 A typical dynamics simulation process for a parallel manipulator

2.5 Substitution of multiple inverse kinematics with a number of single inverse kinematic systems

In a general fully parallel manipulator there can be multiple solutions to kinematics problems. Therefore, there is no unique function that maps the joint space of a manipulator to the tool space or *vice versa*. To cope with this problem in this paper multiple functions are defined which map the tool space of the manipulator to the joint space. In other words, a problem with multiple inverse kinematics solutions is changed into a number of problems each having a single inverse kinematics solution.

Let OS_m denote an m-dimensional space that contains the tool space of the moving platform and JS_n an n-dimensional space containing the joint space. Assume that vector $\boldsymbol{X}$ represents the end effector's position and orientation. If there are s_{inv} different solutions to the inverse kinematics problem of a fully parallel manipulator, then it is possible to define $g_j, j=1,\ldots,s_{\text{inv}}$ to be the jth function which maps the moving platform space to the actuated joint vectors (Fig. 4)

$$\begin{aligned} g_j : OS_m &\rightarrow JS_n \\ \boldsymbol{X} &\rightarrow \boldsymbol{q}_j^{\text{a}} = g_j(\boldsymbol{X}) \end{aligned} \tag{8}$$

It is assumed that $m=n$, that is, only non-redundant manipulators will be studied in this paper. Based on Theorem 1, if the manipulator does not pass any singularity there will be no change in posture and therefore in the inverse kinematics solution. This means that, as long as a manipulator does not pass a singularity, the problem reduces to dealing with a posture that has a single inverse kinematic solution. As an example in Fig. 4, three different points, A, B, and C in the joint space are considered. The figure

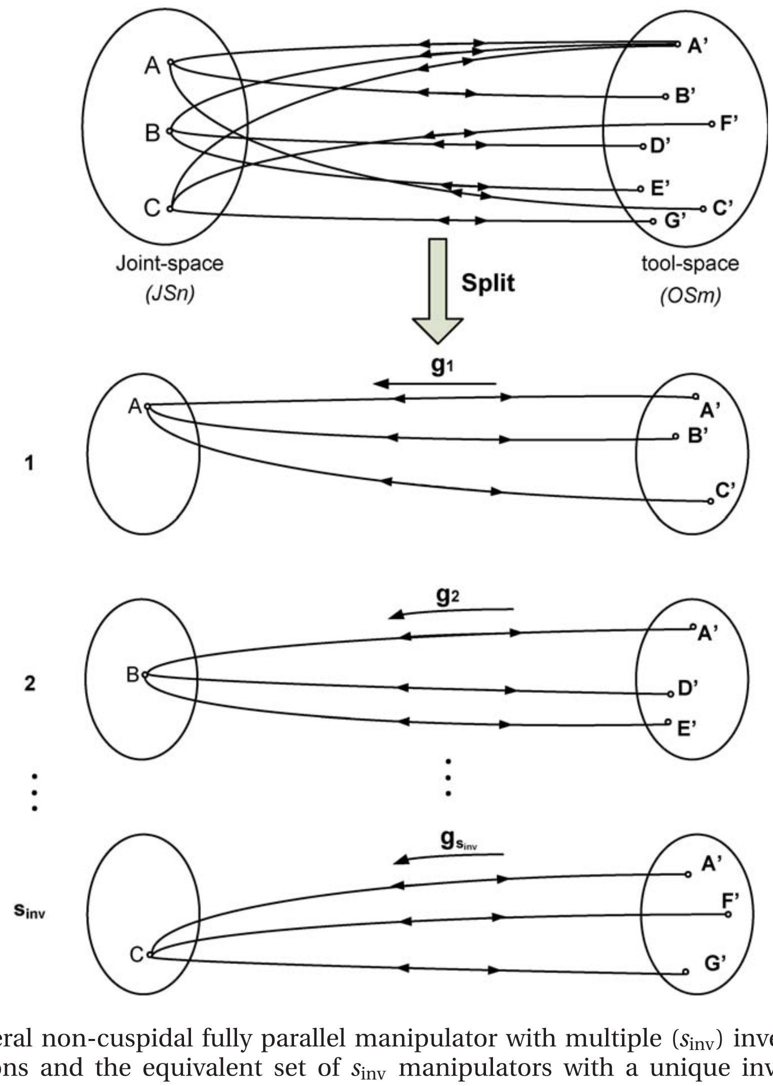

Fig. 4 A general non-cuspidal fully parallel manipulator with multiple (s_{inv}) inverse kinematics solutions and the equivalent set of s_{inv} manipulators with a unique inverse kinematic solution

shows that for a general parallel robot, each of these points may be mapped into a number of points in the tool space. Additionally, more than one point in the joint space may be mapped to the same point in the tool space e.g. A, B, and C are all mapped to A′. As a result, it is obvious that the inverse kinematics of point A′ has more than one solution. However, as shown in the figure, after considering the manipulator as consisting of s_{inv} equivalent manipulators with different postures, the inverse kinematics problem of A′ in each posture will have a unique solution.

Let W_j be the reachable tool space for the jth posture of the manipulator, i.e. the set of all positions and orientations reachable by the moving platform in the considered posture. Furthermore, consider Q_j to be the reachable joint space for the jth posture of the manipulator, i.e. the set of all joint vectors reachable by the actuated joints

$$Q_j = \left\{ \boldsymbol{q}^{\mathrm{a}} \in JS_n | \forall i \leqslant n, q_{i\,\min(j)} \leqslant q_i \leqslant q_{i\,\max(j)} \right\} \tag{9}$$

$$W_j = \left\{ \boldsymbol{X} \in OS_m | g_j(\boldsymbol{X}) \in \Re^n \right\} \tag{10}$$

where $q_{i\,\min(j)}$ and $q_{i\,\max(j)}$, respectively, are the minimum and maximum positions or orientations (depending on the joint type) that the ith joint in the jth posture of the manipulator could reach.

3 BASIC REGIONS

As happens with serial manipulators, the conviction that fully parallel manipulators cannot switch assembly modes without running into a singularity is quite common. This rooted opinion probably stems from misunderstanding the role that singularities have in defining the pattern of related configurations inside the configuration space. However, it has been shown that an assembly mode can change without crossing a singularity [**11**].

An additional set of surfaces, namely the characteristic surfaces, are characterized which divide the tool space into basic regions. The basic regions in the tool space of parallel manipulators were introduced in reference [**15**] with a focus on a case of parallel manipulators that only have one inverse kinematic solution. Likewise, the mapping of each basic region from the tool space into the joint space is its corresponding basic component. It has been shown that assembly modes can be changed just by crossing basic region borders (characteristic surfaces). This paper extends the definition of a basic region to a general non-cuspidal fully parallel manipulator as follows.

Definition 1

Let W_j be the tool space for the jth posture of a non-cuspidal fully parallel manipulator. Basic regions of W_j, denoted by $\left\{W_j b_i, i = 1, 2, \ldots, r\right\}$ where r represents the number of regions in W_j [**17**], are defined as sub-regions of W_j in which there is no trajectory that connects two direct kinematic solutions without passing the borders of the region.

It should be noted that the number of basic regions in W_j is equal to or less than the number of direct kinematic solutions $(r \leqslant s_{\mathrm{dir}})$. Moreover, a basic region may consist of more than one closed area. (An example of this kind of behaviour will be presented in Fig. 6 in which basic region 5 consists of three separate regions.)

Definition 2

Let $Q_j b_i = g_j(W_j b_i)$. The $Q_j b_i$ is a domain in the reachable joint space Q_j and are called the basic components.

In contrast to basic regions, basic components can overlap one another. This means that given a jth posture there may be some areas in the joint space that have more than one direct kinematics solution. These areas are where the basic components overlap one another. The number of overlapped basic components is equal to the number of direct kinematics solutions. Figure 5(a) depicts the basic regions' basic components for a specific posture of a fully parallel manipulator. It shows that each basic region is the map of a basic component from the joint space to the tool space and *vice versa*. Figure 5(b) illustrates that any point, such as A or B, in the overlapped area in the joint space has multiple direct kinematics solutions. The number of solutions is equal to the number of overlapped basic components [**17**].

The following theorem shows that for a specific jth posture, if just one basic region and its corresponding basic component are considered, each point in this basic region has a unique inverse kinematics solution in the corresponding basic components in the joint space, and each point in the corresponding basic component has a unique direct kinematics solution in that specific basic region.

Theorem 2

The restriction of g_i to any basic region is a bijection. In other words, there is only one direct solution in each basic region. A proof is provided in reference [**17**].

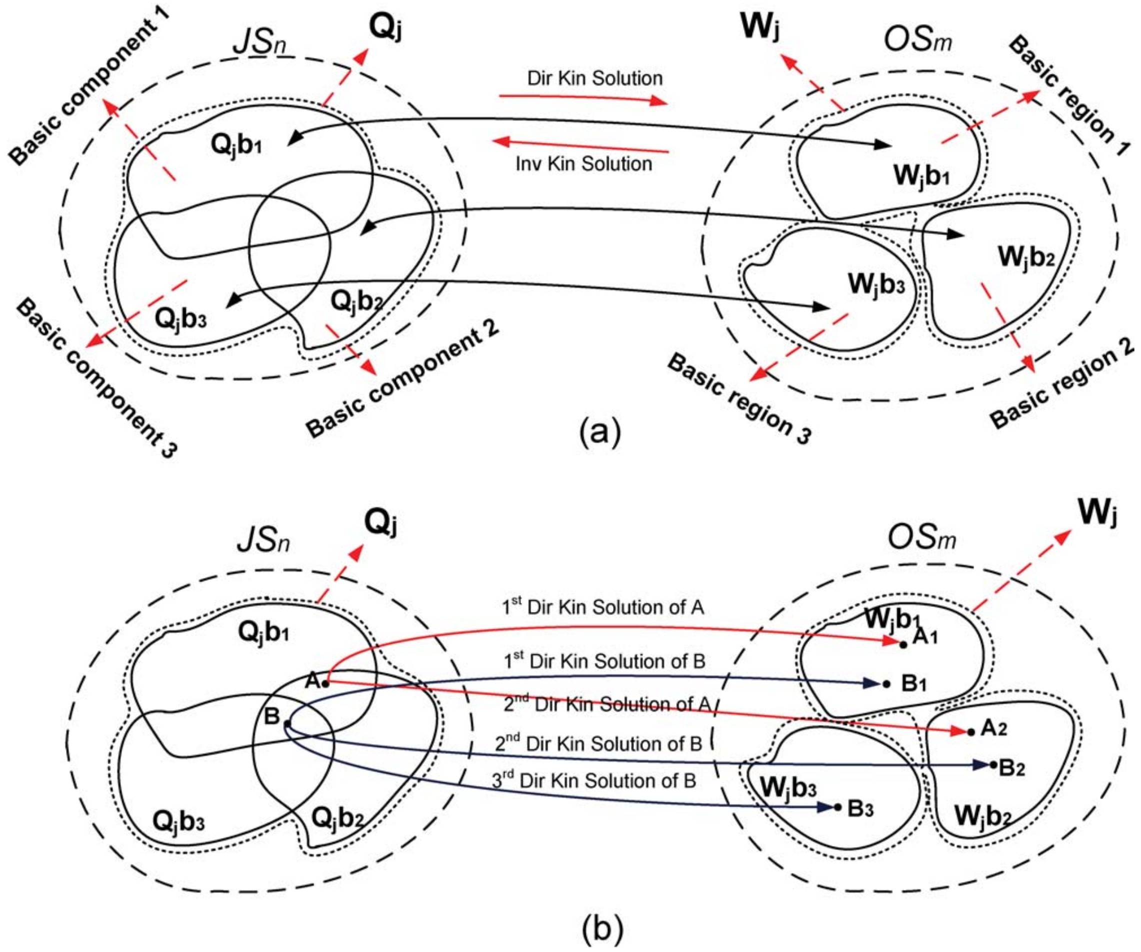

Fig. 5 (a) Basic regions and basic components of a general fully parallel manipulator with only one inverse kinematic solution, and (b) the direct kinematics solutions in overlapped areas

A method for determining the borders of the basic regions for parallel manipulators with one inverse kinematic solution was proposed by reference [**15**]. However, the method in reference [**15**] proposes no straightforward and useful numerical algorithm to determine the basic regions. Moreover, the method is applicable to manipulators with only one inverse kinematic solution. In the current paper, a novel procedure is proposed which utilizes a numerical approach to determine the boundaries of all basic regions in the tool space for any non-cuspidal fully parallel manipulator with multiple inverse and multiple direct kinematics solutions.

3.2 Procedure 1: A numerical method for determining the basic regions in the tool space of parallel manipulators

In this section, a numerical procedure is proposed for finding basic regions for a general fully parallel manipulator. An illustrative example is provided which demonstrates the application of the procedure on a two-dimensional toolspace (Fig. 6). The following algorithm illustrates the steps of the proposed procedure.

1. As was proposed in section 2.5, a general parallel manipulator with multiple (total equal to s_{inv}) inverse kinematics solutions can be considered as s_{inv} different postures. Therefore, a specific posture of the manipulator is selected. Other postures of the manipulator are not considered by this algorithm. If the jth posture is considered, the inverse kinematics of the manipulator can be written in the form of

$$\boldsymbol{q}_j^{\text{a}} = g_j(\boldsymbol{X}) \tag{11}$$

Since $\boldsymbol{q}_j^{\text{a}}$ is the only applicable inverse kinematic solution considered, for the remainder of the algorithm the symbol $\boldsymbol{q}^{\text{a}}$ will be used rather than $\boldsymbol{q}_j^{\text{a}}$.

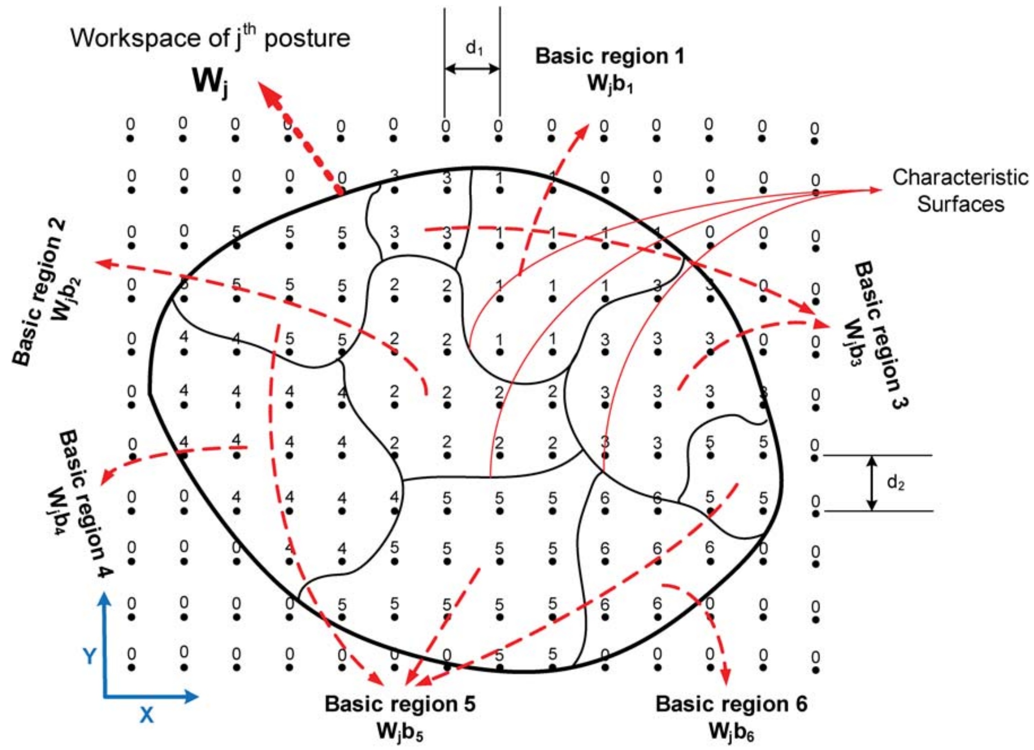

Fig. 6 Determining basic regions using procedure 1 for the *j*th posture of a two-degree-of-freedom parallel manipulator

2. Discretize the tool space of the manipulator into a number of nodes. For an n-dimensional tool space, the coordinates of the moving platform are contained in the vector $\boldsymbol{X}=[x_1 \cdots x_n]^{\mathrm{T}}$. In order to discretize the tool space into equal sizes, it is assumed that the ith dimension of the tool space is divided into u_i equivalent increments. The coordinates of the nodes can be shown with a set of vectors $\{\boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}\}$, where α_i, $1 \leqslant i \leqslant n$, is the index of the ith dimension and

$$1 \leqslant \alpha_i \leqslant (u_i+1) \tag{12}$$

The coordinates of each node can be calculated as

$$x_i^{\alpha_1\alpha_2\cdots\alpha_i\cdots\alpha_n} = d_i \times \alpha_i + x_{i\,\min} \tag{13}$$

where

$$d_i = \frac{x_{i\,\max} - x_{i\,\min}}{u_i} \tag{14}$$

where, $x_{i\max}$ and $x_{i\,\min}$ are the maximum and minimum reachable tool space of the manipulator in the ith dimension, respectively and $x_i^{\alpha_1\alpha_2\cdots\alpha_j\cdots\alpha_n}$ is the ith component of the vector $\boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}$. The value of u_i is an arbitrary integer number. In a non-homogeneous space, different coordinates may be discretized with different resolutions. Therefore, in non-homogeneous spaces different u_i may be used for different coordinates. A greater value of u_i leads to a more accurate map of the basic regions in the tool space. In order to have an acceptable accuracy, the value of u_i should be large enough so that d_i is smaller than the minimum changes in the trajectory. To obtain a higher accuracy one may use different increment sizes along the same direction.
3. Complete steps 4 and 5 for all nodes.
4. Map nodes into the joint space by calculating the inverse kinematics solution of all nodes for the considered posture. Therefore, the solution for node $\boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}$ is represented by $\boldsymbol{q}^{\mathrm{a}\ (\alpha_1\alpha_2\cdots\alpha_n)}$.
5. Map the results obtained in the previous step ($\boldsymbol{q}^{\mathrm{a}\ (\alpha_1\alpha_2\cdots\alpha_n)}$) from the joint space into the tool space by solving the direct kinematics problem. Since multiple solutions exist a method such as Bezout's elimination method [**22**] may be used to find the solutions of the direct kinematics problem. Next, for all nodes arrange all imaginary and real solutions in an arbitrary order (e.g. increasing or decreasing) and number them consecutively; therefore, the following relation is obtained

$$\boldsymbol{q}^{\mathrm{a}(\alpha_1\alpha_2\cdots\alpha_n)} = g_k\left(\boldsymbol{X}'^{\alpha_1\alpha_2\cdots\alpha_n}_{(l)}\right), \quad 1 \leqslant l \leqslant s_{\mathrm{dir}} \tag{15}$$

where $\boldsymbol{X}'^{\alpha_1\alpha_2\cdots\alpha_n}_{(l)}$ is the lth direct kinematics solution. Hereon, l is referred to as solution number. Also, s_{dir} is the number of direct kinematics solutions (real and imaginary) for the considered manipulator.

6. For all nodes, identify the correct direct kinematics solution for the selected posture. This means identifying values of l for which $\boldsymbol{X}'^{\alpha_1\alpha_2\cdots\alpha_n}_{(l)} = \boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}$. Next, assign this value of l to the corresponding node. For example, consider Fig. 6. The tool space of a hypothetical planar manipulator with two degrees of freedom is considered. It is assumed that there are a total of six solutions for the direct kinematics problem. The values 1,…, 6 specify the solution number assigned to each node in the tool space. The value of zero represents nodes that have no direct kinematics solution and therefore they are out of the robot's reachable tool space.
7. If the same ordering method is used to number the multiple direct kinematics solutions for all nodes, when mapping them back into the tool space, then due to Theorem 2, all nodes in a specific region will have the same solution number.
8. Utilizing the previous step, specify the boundaries of basic regions in the manipulator's tool space. Note that any basic region may be composed of disconnected sub-regions. All adjacent nodes with the same assigned number form a sub-region of a basic region. Each region is numbered with the same solution number in that region. Refer to Fig. 6 where basic regions containing nodes with the same solution number are shown. Note that basic regions 3 and 5 are made up of two and three disconnected sub-regions, respectively.
9. Finally, define an n-dimensional tensor called the map tensor which contains the assigned number to the nodes for an n-degrees-of-freedom manipulator. It should be noted that producing the map tensor is an offline process and therefore the size of the calculation is not important. The map tensor is represented as M and its elements as $m_{\alpha_1\alpha_1\cdots\alpha_n}$. Thus

$$M = \left\{ m_{\alpha_1\alpha_1\cdots\alpha_n} = l | \boldsymbol{X}'^{\alpha_1\alpha_2\cdots\alpha_n}_{(l)} = \boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n} \right\} \tag{16}$$

where

$$1 \leqslant \alpha_i \leqslant (u_i + 1),\ \alpha_i \in Z. \tag{17}$$

4 APPLICATION OF BASIC REGIONS IN IDENTIFYING THE ACCEPTABLE SOLUTION OF DIRECT KINEMATICS

Each solution of the direct kinematics results in placing the robot's end effector in a specific physical location in the tool space. As stated earlier, the direct kinematics problem has multiple solutions, all of which are mathematically acceptable. Because of the uniqueness of the acceptable solution, it can only be located in one of the basic regions. Therefore, if the basic region in which the acceptable solution is located is known, the solution number is the same as the basic region number, and the acceptable solution among real solutions can be selected.

As previously mentioned in both online control and dynamic simulation applications there is a stream of joint space vectors that should be converted to tool space vectors. The combination of these points makes a trajectory in the tool space, see Fig. 7. In this paper the previous point is defined as the point in the tool space trajectory where the solution for direct kinematics is complete and the current point is the point that must be determined using direct kinematics. To determine the current point, first, all possible imaginary and real solutions of the direct kinematics problem should be found. Two procedures, procedure 2 and procedure 3, will be proposed in the following sections to determine the current point. This point is the acceptable solution. Procedure 1 is performed offline whereas procedure 2 and procedure 3 are designed for online control applications.

4.1 Procedure 2: Determining the current basic region and end effector distance with borders of the basic region

A numerical procedure is suggested that given the previous position of the end effector will identify the basic region where the end effector is located as well as whether it is near the borders of the region. The procedure uses the map tensor formed in procedure 1.

1. Find assigned numbers of the near-nodes to the previous position of the end effector. If the previous position of the end effector is $\boldsymbol{X}_{\mathrm{previous}}$, the set of assigned numbers to near-nodes, N, is defined as

$$N = \left\{ m^{\alpha_1\alpha_2\cdots\alpha_n} | \left\| \boldsymbol{X}_{\mathrm{previous}} - \boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n} \right\| \leqslant r \right\} \tag{18}$$

where the notation $\|*\|$ stands for the vector

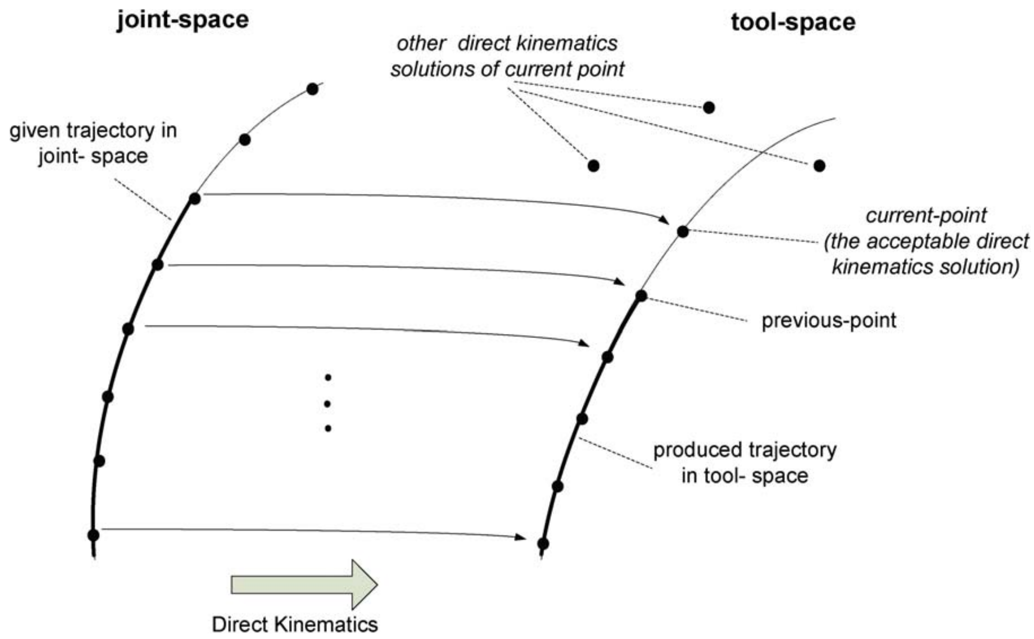

Fig. 7 Joint space trajectory and tool space trajectory

length and r is a constant. If the distance between a node and the previous point is less than r, then the node is called a near-node. For example, if robot has two or three degrees-of-freedom and the coordinates are homogeneous, then r designates the radius of a circle or a sphere, respectively. The centre of this circle or sphere is the position of the previous point, see Fig. 8. A smaller r results in a more accurate determination of the acceptable direct kinematics solution in the following stages. However, r may not be smaller than two or three times the distance between the nodes. In non-homogeneous spaces one may use other meanings for $\|*\|$ and other values for r. For example, for three-dimensional non-homogeneous tool spaces $\|*\|$ may be defined as follows

$$\|\boldsymbol{X}' - \boldsymbol{X}\| = \sqrt{a(x'_1 - x_1)^2 + b(x'_2 - x_2)^2 + c(x'_3 - x_3)^2} \tag{19}$$

where the coefficients a, b, and c are used to normalize the non-homogeneous coordinates.

2. Check all members of the set N.

(a) if all members are the same numbers, then the end effector is located in the region with the same number. Thus, the end effector is far away from the borders;
(b) else, if among members of the set N, there is any member that is different, then the end effector is near a border, see Fig. 8.

4.2 Procedure 3: Determination of the acceptable direct kinematics solution (current point)

This procedure is proposed to find the current point using the information on the previous point in the tool space trajectory. An illustrative example is depicted in Fig. 9.

1. Find all possible direct kinematics solutions, using a method such as Bezout's method, for the current configuration of the manipulator. Then, number them in the same manner as was used for the direct kinematics solutions for the nodes in procedure 1.
2. Use procedure 2 to determine if the previous position of the end effector is near a region border.

(a) If the end effector is near a border, then region of end effector may change for the current point. Equation (3) is used to estimate the region for the current point. The differential form of this equation is

$$\mathbf{J}\,\mathrm{d}\boldsymbol{X} + \mathbf{K}\,\mathrm{d}\boldsymbol{q} = 0 \tag{20}$$

since the distance between adjacent points of the trajectory is assumed to be small, equation (20) can be written as

$$\mathbf{J}\Delta\boldsymbol{X} + \mathbf{K}\Delta\boldsymbol{q} \approx 0$$
$$\Rightarrow\ \mathbf{J}(\boldsymbol{X}_{\text{current}} - \boldsymbol{X}_{\text{previous}}) + \mathbf{K}\left(\boldsymbol{q}^{\text{a}}_{\text{current}} - \boldsymbol{q}^{\text{a}}_{\text{previous}}\right) \approx 0 \tag{21}$$

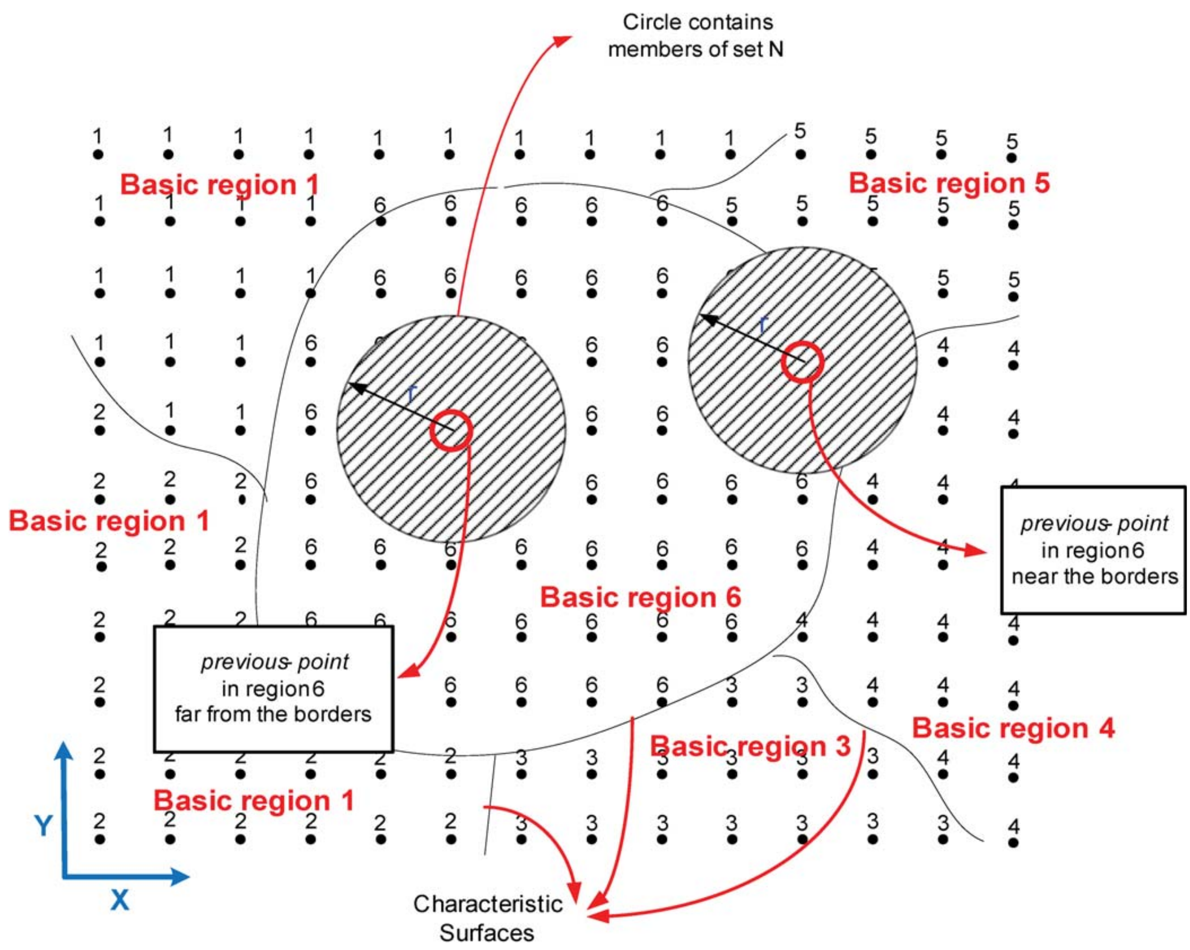

Fig. 8 An example of determining the previous basic region as well as end effector distance with borders of the basic region for a two-degree-of-freedom manipulator

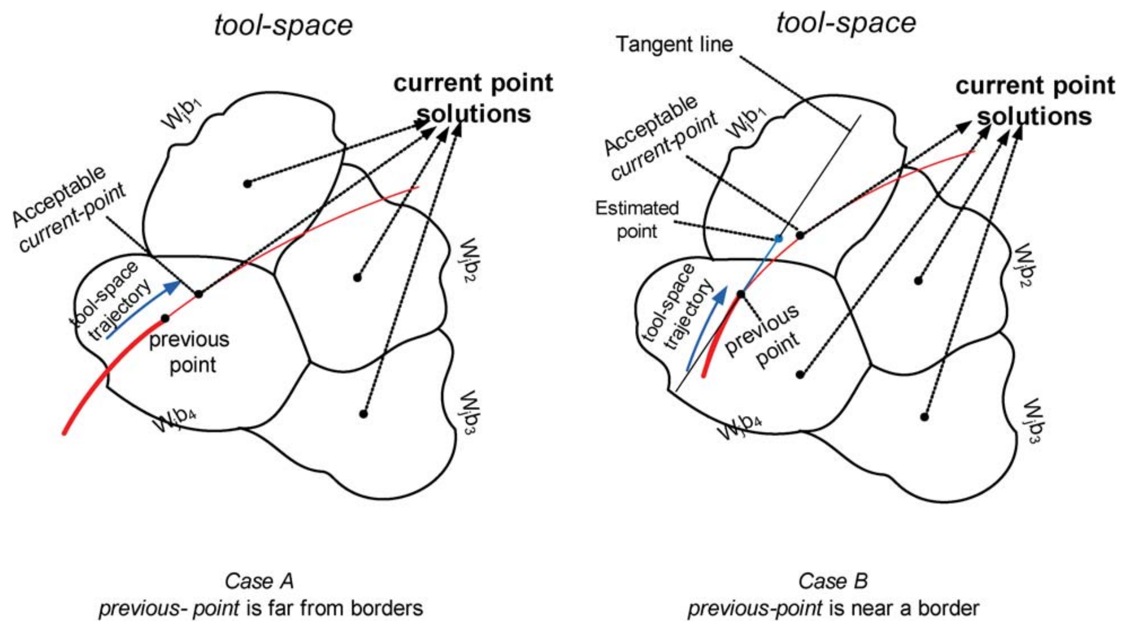

Fig. 9 Utilizing the basic region theorem to find the direct kinematics solution for the current point while the manipulator is following a trajectory in the *j*th posture

The values of $\boldsymbol{X}_{\text{previous}}$, $\boldsymbol{q}^{\text{a}}_{\text{current}}$, and $\boldsymbol{q}^{\text{a}}_{\text{previous}}$ are known. Also, the value of matrices $\mathbf{J}$ and $\mathbf{K}$ can be estimated using the geometrical properties of the manipulator at the previous point. Therefore, the direct kinematics solution for the current point, $\boldsymbol{X}_{\text{current}}$, can be approximated by

$$(\boldsymbol{X}_{\text{current}})_{\text{approximate}} = -\mathbf{J}^{-1}\mathbf{K}\left(\boldsymbol{q}^{\text{a}}_{\text{current}} - \boldsymbol{q}^{\text{a}}_{\text{previous}}\right) + \boldsymbol{X}_{\text{previous}} \quad (22)$$

In order to obtain the exact solution, it is assumed that the region of the current point is the region in which $(\boldsymbol{X}_{\text{current}})_{\text{approximate}}$ is located.

(b) Else, the previous point is far from the region border and use procedure 1 to determine the number of regions in which the previous point is located. Due to the considerable distance of the previous point from the region border, it is very likely that the current point will be located in the same region. Therefore, it is concluded that the current point is located in the same region.

3. The acceptable solution (among solutions obtained in step 1) for the current point is the solution whose assigned number is the same as the number of the predicted region (in step 2).

In a direct kinematics problem, first, procedure 1 should be performed offline to produce the map tensor. In online control or simulation applications one of the existing methods should be used to find all possible direct kinematics solutions in all iterations. Then, procedure 2 and procedure 3 should be utilized to find the acceptable direct kinematics solution in that iteration. This method provides an accurate and reliable way to solve the direct kinematics problem.

5 CASE STUDY: A 3-RRR PLANAR PARALLEL MANIPULATOR

The 3-RRR planar parallel manipulator with three-degrees-of-freedom reported in reference [**3**] is considered (see Fig. 10). Every leg of the manipulator consists of one active and two passive revolute joints. The three motors M_1, M_2, and M_3 are fixed and placed on the vertices of an equilateral triangle. Triangle *ABC* is the moving platform of the manipulator. This manipulator consists of a kinematics chain with three closed loops, namely M_1DABEM_2, M_2EBCFM_3, and M_3FCADM_1. Two of these loops are kinematically independent.

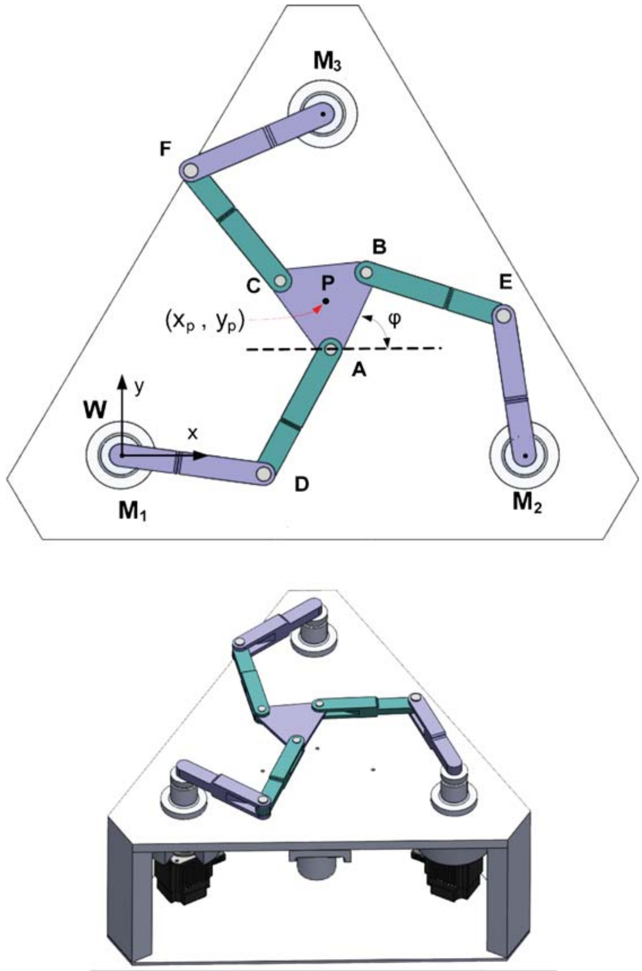

Fig. 10 Planar parallel 3-RRR manipulator

The direct kinematics problem seeks to obtain Cartesian position of point P on the moving platform, x_{p} and y_{p}, as well as the orientation of the moving platform, φ, given the position of the actuated joint angles. Khan *et al.* showed that there exist six solutions for the direct kinematics problem of 3-RRR manipulator [**24**]. They showed that a closed-form solution to the direct kinematics problem is not accessible. Therefore, the solution for the 3-RRR manipulator requires utilization of a numerical method. For example, for the set of input angles (motor angles) shown in Fig. 11, three real solutions exist. The remaining three solutions are all imaginary.

The inverse kinematics problem for this manipulator was solved and was shown to include eight different solutions [**3**]. Consequently, the manipulator has eight different postures. The eight solutions of inverse kinematics problem of the 3-RRR manipulator for a given platform position and orientation are showed in Fig. 12.

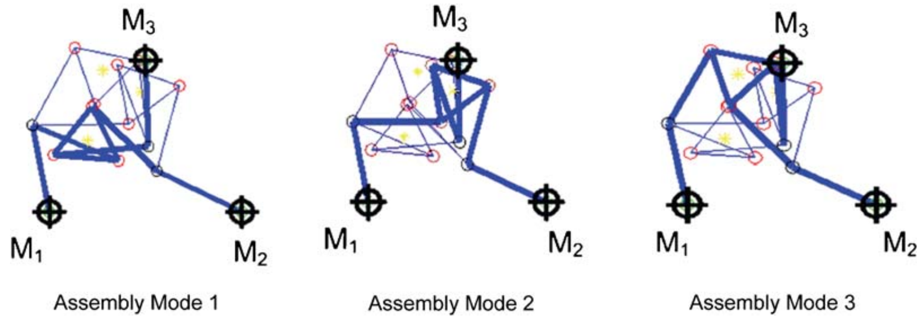

Fig. 11 Three possible direct kinematics solutions (assembly modes) for a set of input angles

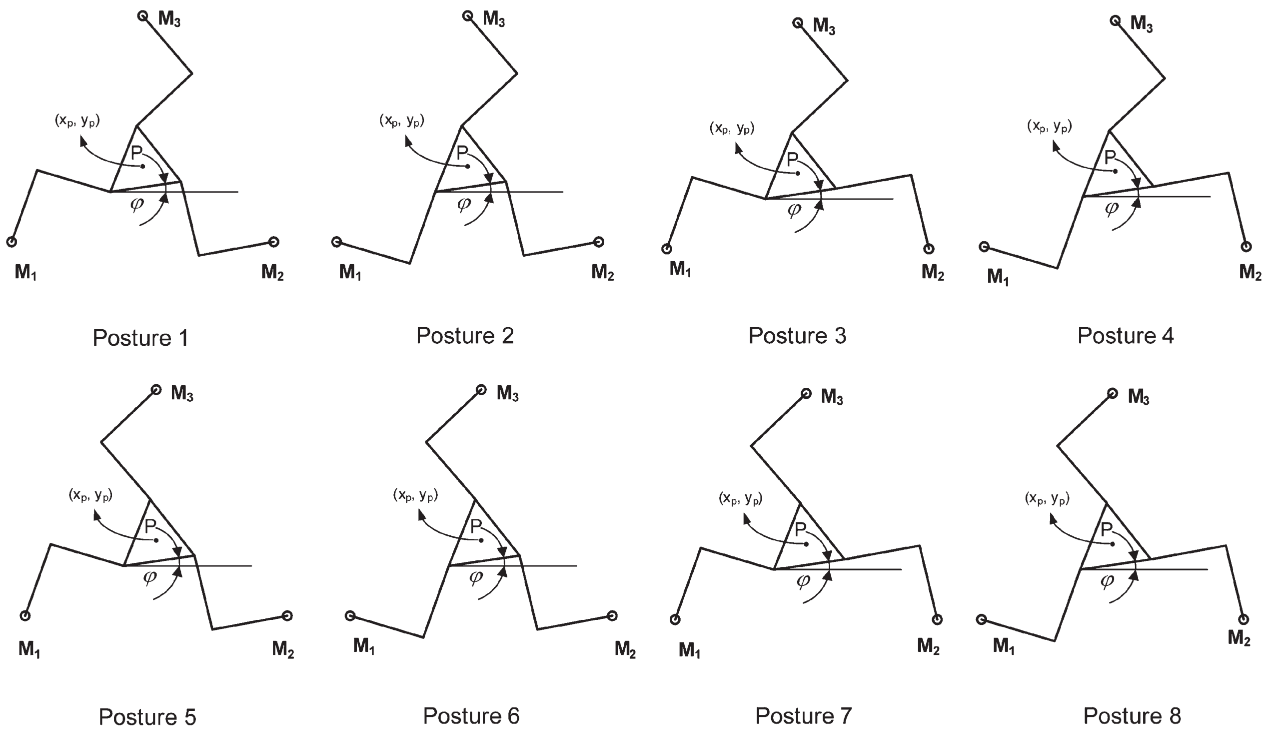

Fig. 12 Eight possible inverse kinematics solutions for the 3-RRR parallel manipulator

5.1 Kinematics description

The properties of the considered 3-RRR parallel manipulator are listed in Table 1.

Table 1 Manipulator properties

Link sizes		Position of motors $[x_p, y_p]^T$(m)	
$\overline{M_1D}=\overline{M_2E}=\overline{M_3F}=l_1$ (m)	0.6	M_1	$[0,0]^T$
$\overline{DA}=\overline{EB}=\overline{FC}=l_2$ (m)	0.6	M_2	$[1,0]^T$
$\overline{AP}=\overline{BP}=\overline{CP}=l_3$ (m)	$\frac{0.3}{\sqrt{3}}$	M_3	$\left[\frac{1}{2},\frac{\sqrt{3}}{2}\right]^T$

5.2 Proposed method

In this section, the direct kinematics problem for the 3-RRR parallel manipulator is solved using the proposed method. First, procedure 1 is applied to determine the basic region in the tool space and form the map tensor. In order to determine the acceptable solution during a trajectory-following application, procedure 2 and procedure 3 are used. Procedure 1 is performed offline, whereas procedure 2 and procedure 3 are designed for online control applications.

Application of procedure 1

1. Select one of the postures. All possible postures for the 3-RRR manipulator are shown in Fig. 12. The selection is arbitrary. There are many parameters that may affect the selection, for example, obstacle avoidance for robot links.

2. Increment each of the three tool space dimensions x, y, and φ from the minimum to the maximum value in steps of size u. Use all combinations of these values as input to the inverse kinematics problem. Due to considering just one posture in step 1, there is just one inverse kinematic solution.
3. Solve the direct kinematics problem given the actuator's angles obtained in the previous step. Using Bezout's elimination method this problem leads to six solutions.
4. Number the six obtained solutions from one to six while considering imaginary answers.
5. Develop the map tensor detailed in procedure 1.

The map tensor is used to determine the basic regions in the tool space of the 3-RRR manipulator. The tool space is three-dimensional; therefore, it is difficult to graphically show all basic regions. For simplification, cross-sectional views representing constant values of φ are shown. For instance, cross-sections of $\varphi = -60^\circ$, $\varphi = 0^\circ$, and $\varphi = 60^\circ$ are shown in Figs 13 to 15. In these figures basic regions are filled with a specific colour representing their numbers. Also, in this figure the singular points are depicted by dark lines. These figures show that singular points fall on the borders of the basic regions but not all basic region borders contain a singularity point.

Multiple direct kinematics solutions for a path crossing basic regions

An illustrative example is supplied to show different direct kinematics solutions for a specific trajectory. In this example a circular path in the tool space is considered. This circular path is discretized into

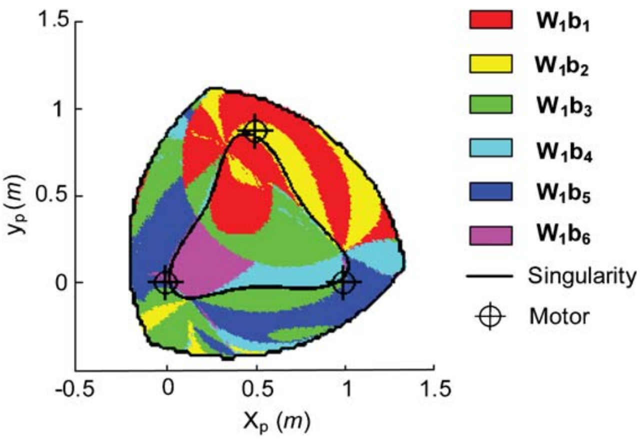

Fig. 13 Cross-section of basic regions and singular points for $\varphi = -60^\circ$ in the first posture of the 3-RRR parallel manipulator

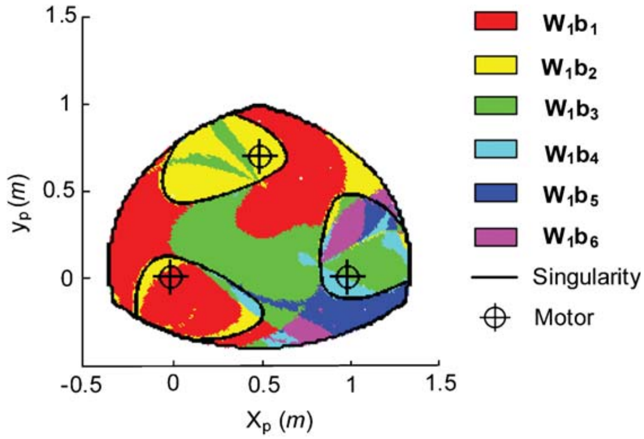

Fig. 14 Cross-section of basic regions and singular points for $\varphi = 0^\circ$ in the first posture of the 3-RRR parallel manipulator

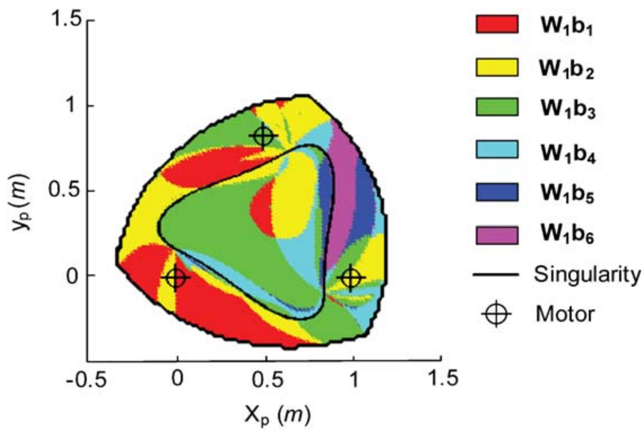

Fig. 15 Cross-section of basic regions and singular points for $\varphi = 60^\circ$ in the first posture of the 3-RRR manipulator

1000 points, see Fig. 16. Using inverse kinematics, the coordinates of these points are transformed into the joint space. As was shown in Fig. 12, there exist eight inverse kinematics solutions (postures) for each point in the tool space. One may select any posture. For example, posture 1 is selected and all points are transformed to the joint space in that posture. This results in a continuous path in the joint space. Next, all points obtained in the joint space are mapped back to the tool space. This step is performed to show how different direct kinematics solutions for a path in the joint space result in different continuous paths in the tool space. As previously mentioned there exist up to six direct kinematics solutions (assembly modes) for each point in the joint space. This mapping back of all points from the joint space to the tool space makes multiple continuous paths in the tool space, see Fig. 17. Different colours

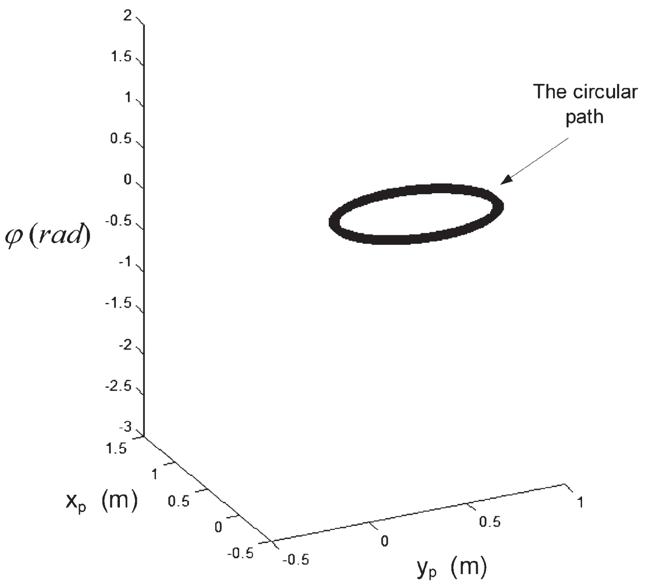

Fig. 16 A discretized circular path in the tool space

along these paths represent different direct kinematics solution (assembly mode) numbers. The change in colour along these paths occurs as a path crosses basic regions. The original circular trajectory can be seen by inspecting Fig. 17.

In trajectory-following applications, because of changes in the number of direct kinematics solutions (assembly modes) jumping from one path to another may occur. Therefore, one must find a way to ensure that the solution stays on the desired path. The method presented in this article offers a way to do this (see section 4.2).

Application of procedure 2 and procedure 3

As stated earlier, in a trajectory-following application, multiple solutions occur in all iterations of the direct kinematics solution. Using procedure 2 and procedure 3, it is possible to identify the acceptable solution (x_p, y_p, and φ of the moving platform). An illustrative example is shown in Fig. 18. In this example a circular trajectory is considered, as the number of points, in the tool space. These points are mapped to the joint space using inverse kinematics. Next, in order to check the validity of the method proposed in this paper, direct kinematics is used to map back the points from the joint space to the tool space.

In this figure, an example is shown where the previous point is near a basic region border. Using procedure 3, the estimated current point is calculated. Then, the one solution that is located in the same basic region as the basic region of the estimated current point is selected. This solution is the acceptable current point solution.

6 CONCLUSIONS

A novel method has been presented which determines the acceptable solution among the multiple solutions of a direct kinematics problem for non-cuspidal parallel manipulators. The direct kinematics problem in parallel manipulators usually leads to multiple solutions. When a control application or dynamic simulation is considered, only one of the

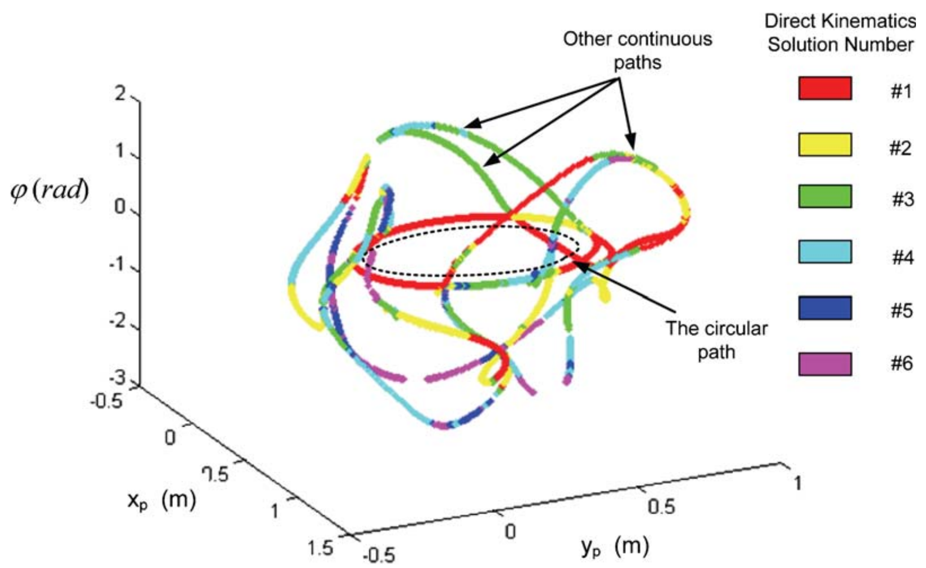

Fig. 17 The mapping back of all points from the joint space to the tool space. Different colours represent different direct kinematics solutions

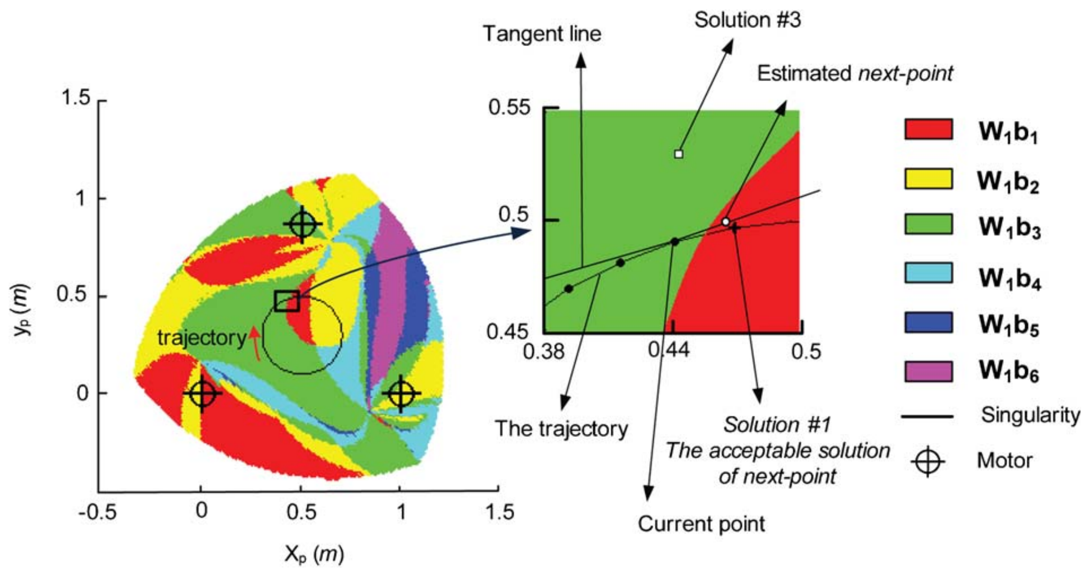

Fig. 18 Utilizing procedure 2 and procedure 3 to determine the acceptable direct kinematics solutions

solutions is acceptable. This paper shows how to implement the concept of basic regions to tackle the problem of determining the acceptable solution. It is shown that when direct kinematics solution of a stream of vectors in the joint space is considered, in all iterations basic regions theory could be used to predict the acceptable solution. The method has been developed using three proposed procedures. The first procedure provides a new extended concept of basic regions and shows a way of determining the basic regions for general non-cuspidal parallel manipulators with multiple inverse and multiple direct kinematics solutions. The second procedure, determines the basic region that the end effector is located in, as well as its distance to the basic region borders. Finally, procedure 3 is suggested which utilizes procedures 1 and 2 to determine which direct kinematics solution is the acceptable one. For illustration, the proposed method has been applied to a 3-RRR parallel manipulator. Basic regions are shown for three different orientations of the moving platform. Next, a circular trajectory in the tool space is used to validate procedure 3. Initially the trajectory is mapped to joint space and then mapped back to the tool space using procedure 3. The obtained trajectory accurately reproduces the original trajectory. This shows that the method could be successfully used to identify the correct solution of the direct kinematics problem. It should be noted that the attractive part of the method is not its computational efficiency but its accuracy and reliability. In future studies, one could consider combining other methods such as soft computing tools to reduce the computation time.

REFERENCES

1 **Merlet, J. P.** *Parallel robots. Solid mechanics and its applications,* second edition, 2006 (Springer, The Netherlands).
2 **Su, Y., Sun, D., Ren, L.,** and **Mills, J. K.** Integration of saturated PI synchronous control and PD feedback for control of parallel manipulators. *IEEE Trans. Robot.*, 2006, **22**, 202–207.
3 **Gosselin, C.** *Kinematics analysis, optimization and programming of parallel robotic manipulators.* PhD Thesis, McGill University, Montreal, Canada, 1988.
4 **Nanua, P., Waldron, K. J.,** and **Murthy, V.** Direct kinematic solution of a Stewart platform. *IEEE Trans. Robot. Autom.*, 1990, **6**(4), 438–443.
5 **Innocenti, C.** and **Parenti-Castelli, V.** Direct position analysis of the Stewart platform mechanism. *Mech. Mach. Theory*, 1990, **25**(6), 611–621.
6 **Dasgupta, B.** and **Mruthyunjay, T. S.** The Stewart platform manipulator: a review. *Mech. Mach. Theory*, 2000, **35**, 15–40.
7 **Innocenti, C.** Forward kinematics in polynomial form of the general Stewart platform 1. *Trans. ASME, J. Mech. Des.*, 2001, **123**, 254–260.
8 **Kamali, K.** and **Akbarzadeh, A.** Fast estimation of direct kinematics of 3-RRR parallel manipulator using Takagi–Sugeno model. The Second Joint Congress on Fuzzy and Intelligent Systems, Malek-e-Ashtar University, Iran, October 2008.
9 **Mahmoodinia, P., Kamali, K.,** and **Akbarzadeh, A.** A novel approach for direct kinematics solution of 3-RRR parallel manipulator following a trajectory. The 16th Annual International Conference on

Mechanical Engineering, Shahid Bahonar University of Kerman, Iran, May 2008.
10 **Merlet, J.-P.** Direct kinematics of parallel manipulators. *IEEE Trans. Robot. Autom.*, 1993, **9**(6), 842–846.
11 **Inocenti, C.** and **Parenti-Castelli, V.** Singularity-free evolution from one configuration to another in serial and fully-parallel manipulators. *Trans. ASME, J. Mech. Des.*, 1998, **120**(1), 73–79.
12 **Siciliano, B.** The Tricept robot: inverse kinematics, manipulability analysis and closed-loop direct kinematics algorithm. *Robotica*, 1999, **17**(4), 437–445.
13 **Merlet, J.-P.** Solving the forward kinematics of a Gough-type parallel manipulator with interval analysis. *Int. J. Robot. Res.*, 2004, **23**(3), 221–236.
14 **Hunt, K. H.** and **Primrose, E. J. F.** Assembly configurations of some in-parallel-actuated manipulators. *Mech. Mach. Theory*, 1993, **28**(1), 31–42.
15 **Wenger, P.** and **Chablat, D.** Uniqueness domains in the workspace of parallel manipulators. In SYROCO'97, Proceedings of the 5th IFAC Symposium on *Robots and control*, Nantes, France, 3–5 September 1997, vol. 2, pp. 431–436.
16 **Wenger, P.** and **Chablat, D.** Workspace and assembly-modes in fully-parallel manipulators: a descriptive study. *Advances on robot kinematics,* Kluwer Academic Publishers, 1998, pp. 117–126.
17 **Chablat, D.** and **Wenger, P.** Working modes and aspects in fully parallel manipulators. In Proceedings of the 1998 IEEE International Conference on *Robotics and automation*, Leuven, Belgium, May 1998, vol. 3, pp. 1964–1969.
18 **Wenger, P.** A classification of manipulator geometries based on singularity avoidance ability. ICAR'93 Tokyo, Japan, 8–9 November 1993, pp. 649–654.
19 **Ma, O.** *Mechanical analysis of parallel manipulators with simulation, design, and control applications.* PhD Thesis, Department of Mechanics, McGill University, Montreal, 1991.
20 **Mathews, J. H.** *Numerical methods for mathematics, science, and engineering*, 1992 (Prentice Hall, London).
21 **Li, T. Y.** Numerical solution of multivariate polynomial systems by homotopy continuation methods. *Acta Numer.*, 1997, 399–436.
22 **Kamali, K.** and **Akbarzadeh, A.** Applications of workspace categorization for parallel manipulators in identification of desired direct kinematics solution. The 17th Annual International Conference on Mechanical Engineering, May 2009, University of Tehran, Iran.
23 **Mourrain, B.** and **Pavone, J. P.** Subdivision methods for solving polynomial equations, technical report, INRIA Sophia-Antipolis, France, 2005.
24 **Khan, W. A., Krovi, V. N., Saha, S. K.,** and **Angeles, J.** Recursive kinematics and inverse dynamics for a planar 3R parallel manipulator. *Trans. ASME, Series G: J. Dyn. Syst. Meas. Control*, 2005, **127**, 529–536.

APPENDIX 1

Notation

F	general kinematics relation
g_j	the jth inverse kinematics function which maps the tool space to the joint space
$\mathbf{J}$	direct-kinematics matrix of parallel manipulators
JS_n	n-dimensional space containing joint space
$\mathbf{K}$	inverse-kinematics matrix of parallel manipulators
l	solution number
$m_{\alpha_1\alpha_1\cdots\alpha_n}$	elements of the map tensor
M	the map tensor
n	degrees-of-freedom of manipulator
N	set of the near nodes
OS_m	m-dimensional space containing tool space of the moving platform
$\boldsymbol{p}$	three-dimensional position vector of the end effector
$\boldsymbol{q}$	vector of all joint coordinates
$\dot{\boldsymbol{q}}$	joint space velocity vector
$\boldsymbol{q}^{\mathrm{a}}$	vector of active joint coordinates
$\boldsymbol{q}^{\mathrm{a}}_{\mathrm{current}}$	vector of current values of active joints
$\boldsymbol{q}^{\mathrm{a}}_{\mathrm{previous}}$	vector of previous values of active joints
$\boldsymbol{q}^{\mathrm{a}}\ (\alpha_1\alpha_2\cdots\alpha_n)$	the inverse kinematics solution for node $\boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}$ in a specific posture
$\boldsymbol{q}^{\mathrm{u}}$	vector of passive joint coordinates
$q_{i\ \max(j)}$	maximum value that the ith active joint in jth posture of the manipulator could reach
$q_{i\ \min(j)}$	minimum value that the ith active joint in jth posture of the manipulator could reach
q_j	the jth solution of the inverse kinematics problem
Q_j	the reachable joint space for jth posture of the manipulator
Q_jb_i	ith basic component in Q_j
r	the number of basic regions that exist in a specific posture
s_{dir}	the number of direct kinematics solutions
s_{inv}	the number of inverse kinematics solutions
u	number of steps into which each tool space dimension is divided
W_j	the reachable tool space for jth posture of the manipulator
W_jb_i	the ith basic region in W_j

$x_{i\ \max}$	the maximum reachable tool space of the manipulator in ith dimension
$x_{i\ \min}$	the minimum reachable tool space of the manipulator in ith dimension
$x_i^{\alpha_1\alpha_2\cdots\alpha_j\cdots\alpha_n}$	the ith component of the vector $\boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}$
x_{p}	the value of x-axis coordinate of end effector of 3-RRR manipulator
$\boldsymbol{X}$	tool space coordinate vector
$\boldsymbol{X}_{\mathrm{current}}$	vector of current (acceptable direct kinematics solution) position of end effector
$(\boldsymbol{X}_{\mathrm{current}})_{\mathrm{approximate}}$	estimated value for vector of current position of end effector
$\boldsymbol{X}_{\mathrm{previous}}$	vector of previous position of end effector
$\boldsymbol{X}^{\alpha_1\alpha_2\cdots\alpha_n}$	vector of the coordinates of the nodes
$\boldsymbol{X}'^{\alpha_1\alpha_2\cdots\alpha_n}_{(l)}$	the lth direct kinematics solution
y_{p}	the value of y-axis coordinate of end effector of 3-RRR manipulator
α_i	the index of ith dimension
φ	the orientation of moving platform of 3-RRR manipulator
$\boldsymbol{\Theta}$	three-dimensional orientation vector of the moving platform

APPENDIX 2

Definitions

Postures
Different inverse kinematics solutions for parallel manipulators result in different configurations. These different configurations are called postures.

Assembly mode
Multiple direct kinematics solutions for parallel manipulators result in different configurations. These different configurations are called assembly modes.

Cuspidal manipulators
Cuspidal manipulators defined for serial manipulators are those which can change posture without passing any singularity.

Characteristic surfaces
Characteristic surfaces are characterized which divide the tool space into basic regions.

Basic regions
Given a specific posture of a non-cuspidal fully parallel manipulator, the basic regions are defined as sub-regions in the tool space in which there is no trajectory which connects two direct kinematics solutions without passing the borders of the region.

Basic components
The mapping of each basic region from the tool space into the joint space is its corresponding basic component.

Number of basic regions
Number of basic regions in W_j is equal to, or less than, the number of direct kinematics solutions ($r \leqslant s_{\mathrm{dir}}$). Moreover, a basic region may consist of more than one closed area.

Overlap of the basic components
In contrast to basic regions, basic components can overlap each other. The number of overlapped basic components in a specific point in the joint space is equal to the number of direct kinematics solutions.

Theorem 1
For non-cuspidal parallel manipulators, if the end effector does not pass any singularity, the posture of the manipulator cannot change.

Change in assembly modes
It is shown that assembly modes can change without crossing singularities. Assembly modes can change just by crossing basic region borders.

Theorem 2
The restriction of g_i to any basic region is a bijection. In other words, there is only one direct solution in each basic region.