Trajectory-tracking control of a planar 3-RRR parallel manipulator with singularity avoidance

Chaman Nasa*
Department of Engineering Design
Indian Institute of Technology Madras
Chennai, India

Sandipan Bandyopadhyay†
Department of Engineering Design
Indian Institute of Technology Madras
Chennai, India

Abstract

Avoiding singularities while tracking a commanded trajectory is an inherent challenge in any parallel manipulator. Off-line path planning and trajectory generation can be used where the singularity manifold is relatively well-known, and/or amenable to mathematical analysis. However, for manipulators such as the planar 3-RRR, the singularity curve is of degree 42 for a given orientation of the moving platform, and therefore not very easy to either analyse or avoid during path planning.

In this paper, an alternate strategy is presented, whereby the manipulator is enabled to trace a curve containing singularities by virtue of allowing its end-effector orientation to change. The orientation is used as a redundant degree-of-freedom in this context, and it helps to keep the manipulator away from singularities in the 3-dimensional task space, while executing a given trajectory on a curve. While the idea of confining a non-redundant serial manipulator on a lower-dimensional task space to avoid singularities is fairly dated and well-established, its use in parallel manipulators in new to the best of our knowledge. We present numerical simulations to show that with suitable choice of the control parameters, the manipulator is able to execute "singular" trajectories.

*chaman.nasa@gmail.com
†Corresponding author, sandipan@iitm.ac.in

1 Introduction

The planar 3-RRR is a parallel manipulator with three-degrees-of-freedom, which correspond to its ability to move the triangle $\boldsymbol{p}_1\boldsymbol{p}_2\boldsymbol{p}_3$ in any direction in the plane, and orient it arbitrarily about any point (see fig. (1))[1]. With these abilities, the manipulator has several practical applications, e.g., pick-and-place operations in the plane, tracing planar curves for profile-cutting etc.

However, like all parallel manipulators, the 3-RRR suffers from *gain-type* singularities inside its workspace (see, e.g., [1, 2, 3]). It is known that the *singularity curve*, defined as the locus of the centre $\boldsymbol{p}_c(x, y)$ when the manipulator is at a gain-type singularity for a given orientation of the moving platform (denoted by α), is of degree 42 in x, y [2]. Therefore, determination of this curve and avoiding it algorithmically while generating a desired path are challenging problems, which have not been solved satisfactorily to the best of our knowledge. In fact, no closed-form derivation of the singularity manifold is available in literature.

It seems attractive, therefore, to adopt a strategy, to avoid singularities at a different stage than kinematics, such that explicit knowledge of the singularity manifold is not required in tracing out a path. We treat singularity-avoidance as a *sub-task* at the stage of motion control. Philosophically, this is a straight-forward adaption of the local resolution of redundancy through singularity avoidance, as proposed by Nakamura [4] for serial manipulators. For the given task of tracking a curve on a plane (with arbitrary platform orientation), the 3-RRR can be considered to have one redundant degree-of-freedom, represented by the variable α. We modify this "free" variable α in such a manner that the singularity curve moves/stays away from the commanded trajectory. This is achieved through the maximisation of a non-negative *artificial potential* function, the zeros of which signify singularity. Details of the mathematical development is presented in section (4).

To test our algorithm, we command the manipulator to track a given path, such that with the initial orientation $\alpha(t)$, the path intersects the singularity manifold at multiple points. However, as the simulation shows, when the above-mentioned artificial potential is included in the control scheme, it modifies α in such a way that the singularity curves corresponding to these orientations have no intersection with the given path. These results are presented in section (5).

It may be noted here that the present work differs from the established approach of singularity-avoidance utilising kinematic redundancy, which is introduced by employing more actuators than the dimension of the original task-space. For instance, in [5], the authors introduce three degrees of redundancy in a planar 3-RRR by making the length of each actuated link of variable, by

[1]It is understood that the constraints of *reachable* and *orientable* workspaces apply.

means of three prismatic actuators. In our approach, however, the geometry and the actuation scheme of the manipulator remain unchanged. The redundancy appears due to the confinement of the primary manipulation task to a space of dimension lower than the actual degree-of-freedom of the manipulator. The approach is motivated by the observation that it is fairly common in practice to find such *lower dimensional tasks*. For instance, in a pick-and-place operation, the orientation of the payload may not be specified very strictly in most locations, except for the extremities of the path. Therefore, for the greater part of the trajectory, the manipulator can be treated as redundant.

2 Kinematic model of the 3-RRR

2.1 Geometry of the manipulator

The 3-RRR planar parallel manipulator is shown in figure 1. The "end-effector" of the manip-

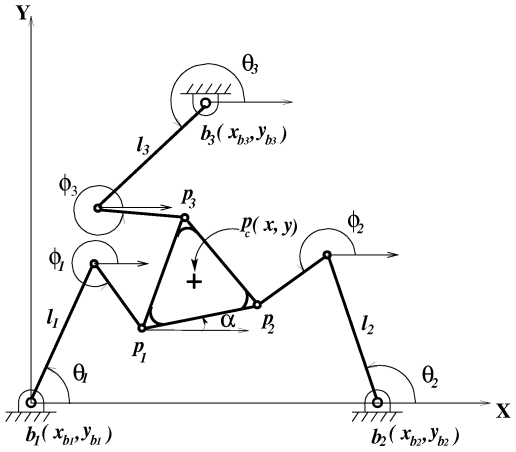


Figure 1: Schematic of the 3-RRR planar parallel manipulator

ulator is an equilateral triangle, $\boldsymbol{p}_1\boldsymbol{p}_2\boldsymbol{p}_3$, which is connected to a fixed base $\boldsymbol{b}_1\boldsymbol{b}_2\boldsymbol{b}_3$, which also forms an equilateral triangle of length b. The vertices $\boldsymbol{p}_i$ are connected to the corresponding base points of manipulator $\boldsymbol{b}_i$ by three identical fingers, where $i = 1, 2, 3$. Each finger comprises of two links, of length l and r respectively, connected by revolute joints. The actuators are located at $\boldsymbol{b}_i$ and therefore the links l_1, l_2, l_3 are called the *active* links and r_1, r_2, r_3 the *passive* links. The platform has three degrees-of-freedom. These are the two translations in the plane parallel to the fixed base and a rotation about the normal to this plane.

2.2 Forward kinematics

Though the degree-of-freedom of the 3-$\underline{\text{R}}$RR is three, a complete description of the manipulator requires the specification of two sets of variables: (a) the active joint coordinates, $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)^T$, and (b) the passive joint coordinates, $\boldsymbol{\phi} = (\phi_1, \phi_2, \phi_3, \alpha)^T$. Therefore, the configuration of the manipulator is defined by the variable $\boldsymbol{q} = (\boldsymbol{\theta}, \boldsymbol{\phi})$. Note that the orientation of the moving platform, α, is included in the passive variable $\boldsymbol{\phi}$. This is because like ϕ_1, ϕ_2, ϕ_3, it is also determined indirectly through the solution of the forward kinematics of the manipulator.

2.2.1 Loop closure equations

The manipulator must move in a manner so as to ensure that the mechanical integrity of the system, as defined by the *loop-closure* equations, is maintained at all times. Considering the loops between fixed pivots $\boldsymbol{b}_1, \boldsymbol{b}_2$ and $\boldsymbol{b}_2, \boldsymbol{b}_3$ respectively, we can write the following equations which define the variable $\boldsymbol{\phi}$ implicitly:

$$\begin{aligned}
\eta_1 =& l_1 \cos\theta_1 + r_1 \cos\phi_1 + a\cos\alpha - r_2\cos\phi_2 \\
& - l_2\cos\theta_2 - x_{b2} = 0 \qquad (1)\\
\eta_2 =& l_1 \sin\theta_1 + r_1 \sin\phi_1 + a\sin\alpha - r_2\sin\phi_2 \\
& - l_2\sin\theta_2 = 0 \qquad (2)\\
\eta_3 =& x_{b2} + l_2\cos\theta_2 + r_2\cos\phi_2 + a\cos\left(\frac{2\pi}{3}+\alpha\right) \\
& - r_3\cos\phi_3 - l_3\cos\theta_3 - x_{b3} = 0 \qquad (3)\\
\eta_4 =& l_2\sin\theta_2 + r_2\sin\phi_2 + a\sin\left(\frac{2\pi}{3}+\alpha\right) \\
& - r_3\sin\phi_3 - l_3\sin\theta_3 - y_{b3} = 0 \qquad (4)
\end{aligned}$$

Put together, the loop-closure constraints can be written as:

$$\boldsymbol{\eta}(\boldsymbol{q}) = \boldsymbol{\eta}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbf{0}, \text{ where } \boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3, \eta_4)^T \qquad (5)$$

The forward kinematics problem of the manipulator refers to finding the solutions $\boldsymbol{\phi} = \boldsymbol{\phi}(\boldsymbol{\theta})$ of equation (5).

2.2.2 Solution of the loop-closure equations

Equations (1, 2) can be solved *linearly* for $\sin\phi_1$ and $\cos\phi_1$ to obtain:

$$\begin{aligned}\cos\phi_1 &= \frac{-l_1\cos\theta_1 - a\cos\alpha + r_2\cos\phi_2 + l_2\cos\theta_2 + x_{b2}}{r_1} \\ \sin\phi_1 &= \frac{-l_1\sin\theta_1 - a\sin\alpha + r_2\sin\phi_2 + l_2\sin\theta_2}{r_1}\end{aligned} \tag{6}$$

From these, ϕ_1 is obtained *uniquely:*

$$\phi_1 = \text{atan2}(\sin\phi_1, \cos\phi_1) \tag{7}$$

where $\text{atan2}(\sin(\cdot), \cos(\cdot))$ represents the two-argument arc-tangent function. Eliminating ϕ_1 by squaring and adding equations (6), we get an equation of the form $f_1(\boldsymbol{\theta}, \phi_2, \alpha) = 0$. Similarly, from equations (3, 4), we obtain ϕ_3 and another equation of the form $f_2(\boldsymbol{\theta}, \phi_2, \alpha) = 0$. Incidentally, f_1, f_2 are linear in $\cos\phi_2, \sin\phi_2$, and therefore applying a similar procedure as above, we solve for ϕ_2, and obtain the following equation containing only one unknown, α:

$$f(\boldsymbol{\theta}, \alpha) = 0 \tag{8}$$

To solve this trigonometric equation, it is first converted into a polynomial equation in the variable $t = \tan(\alpha/2)$:

$$a_0 t^6 + a_1 t^5 + a_2 t^4 + a_3 t^3 + a_4 t^2 + a_5 t + a_6 = 0 \tag{9}$$

The coefficients, $a_i,\ i = 0, \ldots, 6$, are obtained as closed-form expressions of the architecture parameters (a, b, l, r) and $\boldsymbol{\theta}$, symbolic simplification algorithms described in [6] implemented on the commercial software `Mathematica`. For a given set of numerical values of the inputs, equation (9) can be solved numerically for all the six values of t. Therefore all the corresponding values of α and ϕ_i can be obtained uniquely, leading to the 6 sets of solutions to the forward kinematics problem.

2.3 Velocity kinematics

In order to find out velocity of the target point $\boldsymbol{p}_c$, i.e., the centre of the moving triangle, we need to find out the passive joint rates in addition to the active joint rates. Differentiating equation (5) with respect to time, we get:

$$\frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\theta}}\dot{\boldsymbol{\theta}} + \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\phi}}\dot{\boldsymbol{\phi}} = \mathbf{0} \tag{10}$$

$$\Rightarrow \dot{\boldsymbol{\phi}} = -\frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\phi}}^{-1} \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}}, \quad \det\left(\frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\phi}}\right) \neq 0 \tag{11}$$

Writing $\boldsymbol{J}_{\boldsymbol{\eta\theta}} = \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\theta}}$, $\boldsymbol{J}_{\boldsymbol{\eta\phi}} = \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\phi}}$, we obtain:

$$\dot{\boldsymbol{\phi}} = \boldsymbol{J}_{\boldsymbol{\phi\theta}} \dot{\boldsymbol{\theta}}, \ \boldsymbol{J}_{\boldsymbol{\phi\theta}} = -\boldsymbol{J}_{\boldsymbol{\eta\phi}}^{-1} \boldsymbol{J}_{\boldsymbol{\eta\theta}}, \ \det\left(\boldsymbol{J}_{\boldsymbol{\eta\phi}}\right) \neq 0 \tag{12}$$

Equation (12) gives an expression of the passive joint rates in terms of the active joint rates. Further, it leads to the definition of an important linear relationship, namely, the mapping of the active joint rates $\dot{\boldsymbol{\theta}}$ into the rate of the configuration space variable:

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_{\boldsymbol{q\theta}} \dot{\boldsymbol{\theta}}, \text{ where,} \tag{13}$$

$$\boldsymbol{J}_{\boldsymbol{q\theta}} = \begin{pmatrix} \boldsymbol{I}_3 \\ \boldsymbol{J}_{\boldsymbol{\phi\theta}} \end{pmatrix} \tag{14}$$

Consider any point $\boldsymbol{p}$ on the manipulator. Since $\boldsymbol{p} = \boldsymbol{p}(\boldsymbol{q})$, its velocity, $\dot{\boldsymbol{p}}$, can always be written as:

$$\dot{\boldsymbol{p}} = \frac{\partial \boldsymbol{p}}{\partial \boldsymbol{q}} \dot{\boldsymbol{q}} = \boldsymbol{J}_{\boldsymbol{pq}} \dot{\boldsymbol{q}}, \text{ where } \boldsymbol{J}_{\boldsymbol{pq}} = \frac{\partial \boldsymbol{p}}{\partial \boldsymbol{q}} \tag{15}$$

All the Jacobian matrices appearing above can be computed at any configuration after the forward kinematics problem has been solved. Therefore, the linear velocity of any point of interest, can be obtained by the above process, if the joint velocity vector, $\dot{\boldsymbol{\theta}}$, is known. With the knowledge of the position and velocities, we now proceed to formulate the dynamic model of the manipulator.

3 Dynamic model of the 3-$\underline{\text{R}}$RR

The dynamic model is an essential part of the control scheme as it is used for feedback linearisation of the equation of motion in the control scheme. We use the Lagrangian framework to develop the equation of motion. As in the case of kinematics, dynamics also requires a full description of the system including the motion of the passive links. This is evident in the expression for the total kinetic energy of the manipulator:

$$T = \sum_{i=1}^{7} T_i, \ T_i = \frac{1}{2} \boldsymbol{v}_{\boldsymbol{p}_{c_i}}^T m_i \boldsymbol{v}_{\boldsymbol{p}_{c_i}} + \frac{1}{2} \boldsymbol{\omega}_i^T \boldsymbol{I}_{ci} \boldsymbol{\omega}_i, \tag{16}$$

where, m_i is the mass, $\boldsymbol{v}_{\boldsymbol{p}_{c_i}}$ is the velocity of the mass centre $\boldsymbol{p}_{\boldsymbol{c}_i}$, $\boldsymbol{\omega}_i$ is the angular velocity, and, $\boldsymbol{I}_{ci}$ is the moment of inertia of the ith link about its centre with reference to the body-fixed

frame. Using equation (15), we can write $\boldsymbol{v}_{\boldsymbol{p}_{c_i}} = \boldsymbol{J}_{\boldsymbol{pq}_i}\dot{\boldsymbol{q}}$, and similarly, $\boldsymbol{\omega}_i = \boldsymbol{J}_{\boldsymbol{\omega}i}\dot{\boldsymbol{q}}$. Therefore, we rewrite T_i as:

$$T_i = \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{J}_{\boldsymbol{pq}i}^T m_i \boldsymbol{J}_{\boldsymbol{pq}i}\dot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{J}_{\boldsymbol{\omega}i}^T \boldsymbol{I}_{ci}\boldsymbol{J}_{\boldsymbol{\omega}i}\dot{\boldsymbol{q}}, \tag{17}$$

From the last two equations, the mass matrix of the system can be obtained through a series of matrix multiplications and additions. Further, the matrix of centripetal and Corioli's force, $\boldsymbol{C}$, can be computed from the mass matrix (see, for e.g., [7]):

$$\boldsymbol{C}_{ij} = \frac{1}{2}\sum_{k=1}^{n}\left(\frac{\partial \boldsymbol{M}_{ij}}{\partial \boldsymbol{q}_k} + \frac{\partial \boldsymbol{M}_{ik}}{\partial \boldsymbol{q}_j} - \frac{\partial \boldsymbol{M}_{kj}}{\partial \boldsymbol{q}_i}\right)\dot{\boldsymbol{q}}_k \tag{18}$$

With these elements, we can now put together the Lagrangian equation of motion for the manipulator as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \boldsymbol{\tau} + \boldsymbol{\tau}_c \tag{19}$$

Note that the customary gravitational potential term is omitted on the left side of the equation, as the manipulator is assumed to be in a horizontal plane in this paper. Further, friction is neglected as well as any other disturbing forces. The vectors $\boldsymbol{\tau}$, $\boldsymbol{\tau}_c$ represent the applied torque, and the torques arising out of the kinematic constraints respectively. It is well-known that $\boldsymbol{\tau}_c$ can be written as $\boldsymbol{J}_{\boldsymbol{\eta q}}^T\boldsymbol{\lambda}$ (see, e.g., [7]), where:

$$\boldsymbol{J}_{\boldsymbol{\eta q}} = \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{q}} \tag{20}$$

Therefore, the equation of motion of the manipulator can be written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \boldsymbol{\tau} + \boldsymbol{J}_{\boldsymbol{\eta q}}^T\boldsymbol{\lambda}, \tag{21}$$

where, $\boldsymbol{\lambda}$ is a vector of Lagrange multipliers. It is possible to explicitly evaluate $\boldsymbol{\lambda}$ from equation (21) by obtaining $\ddot{\boldsymbol{q}}$ from it and substituting it in the second derivative of the constraint equation $\boldsymbol{\eta}(\boldsymbol{q}) = \boldsymbol{0}$ (see, e.g., [7]). However, it is also possible to eliminate it without evaluation, as shown below (see, e.g., [8], [9]). Note that:

$$\frac{d\boldsymbol{\eta}}{dt} = \boldsymbol{0} \Rightarrow \boldsymbol{J}_{\boldsymbol{\eta q}}\dot{\boldsymbol{q}} = \boldsymbol{0} \tag{22}$$

Also, differentiating equation (13), we get:

$$\ddot{\boldsymbol{q}} = \boldsymbol{J}_{\boldsymbol{q\theta}}\ddot{\boldsymbol{\theta}} + \dot{\boldsymbol{J}}_{\boldsymbol{q\theta}}\dot{\boldsymbol{\theta}} \tag{23}$$

Substituting $\dot{\boldsymbol{q}}$, $\ddot{\boldsymbol{q}}$ from equation (22, 23) into equation (21), we get:

$$\boldsymbol{M}\boldsymbol{J}_{\boldsymbol{q\theta}}\ddot{\boldsymbol{\theta}} + \left(\boldsymbol{M}\dot{\boldsymbol{J}}_{\boldsymbol{q\theta}} + \boldsymbol{C}\boldsymbol{J}_{\boldsymbol{q\theta}}\right)\dot{\boldsymbol{\theta}} = \boldsymbol{\tau} + \boldsymbol{J}_{\boldsymbol{\eta q}}^{T}\boldsymbol{\lambda} \tag{24}$$

Pre-multiplying equation (24) by $\boldsymbol{J}_{\boldsymbol{q\theta}}^{T}$ we get:

$$\begin{aligned}\boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\boldsymbol{M}\boldsymbol{J}_{\boldsymbol{q\theta}}\ddot{\boldsymbol{\theta}} + \boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\left(\boldsymbol{M}\dot{\boldsymbol{J}}_{\boldsymbol{q\theta}} + \boldsymbol{C}\boldsymbol{J}_{\boldsymbol{q\theta}}\right)\dot{\boldsymbol{\theta}} \\ - \boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\boldsymbol{\tau} - \boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\boldsymbol{J}_{\boldsymbol{\eta q}}^{T}\boldsymbol{\lambda} = \boldsymbol{0}\end{aligned} \tag{25}$$

It is interesting to note that $\boldsymbol{J}_{\boldsymbol{\eta q}}\boldsymbol{J}_{\boldsymbol{q\theta}} = \boldsymbol{0}$, since from equation (22), $\boldsymbol{J}_{\boldsymbol{\eta q}}\dot{\boldsymbol{q}} = \boldsymbol{0} \Rightarrow \boldsymbol{J}_{\boldsymbol{\eta q}}\boldsymbol{J}_{\boldsymbol{q\theta}}\dot{\boldsymbol{\theta}} = \boldsymbol{0}$, where $\dot{\boldsymbol{\theta}}$ is arbitrary. Therefore, $\boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\boldsymbol{J}_{\boldsymbol{\eta q}}^{T}\boldsymbol{\lambda} = \boldsymbol{0}$ for any finite $\boldsymbol{\lambda}$. This last step, therefore, effectively eliminates the constraint force term, and creates an *equivalent unconstrained dynamic system* as follows:

$$\boldsymbol{M}_{\boldsymbol{\theta}}\ddot{\boldsymbol{\theta}} + \boldsymbol{C}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} = \boldsymbol{\tau}_{\boldsymbol{\theta}}, \tag{26}$$

where, $\boldsymbol{M}_{\boldsymbol{\theta}} = \boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\boldsymbol{M}\boldsymbol{J}_{\boldsymbol{q\theta}}$ is the mass matrix of the manipulator *as reflected on the active coordinates alone*, $\boldsymbol{C}_{\boldsymbol{\theta}} = \boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\left(\boldsymbol{M}\dot{\boldsymbol{J}}_{\boldsymbol{q\theta}} + \boldsymbol{C}\boldsymbol{J}_{\boldsymbol{q\theta}}\right)$ is the corresponding centripetal and Corioli's term, and $\boldsymbol{\tau}_{\boldsymbol{\theta}} = \boldsymbol{J}_{\boldsymbol{q\theta}}^{T}\boldsymbol{\tau}$ is the vector of joint torques.

Expressed in this way, the dynamic model of the parallel manipulator is not different from that of an *equivalent* serial manipulator with the same actuators. It also has a compact form, i.e., it has only 3 *ordinary differential equations* as opposed to 7 in the original system (21). However, there is a very serious distinction, which is apparently buried in the matrix manipulations above. The model (26) is only valid when the mapping $\dot{\boldsymbol{q}} = \boldsymbol{J}_{\boldsymbol{q\theta}}\dot{\boldsymbol{\theta}}$ is well-defined, i.e., when $\boldsymbol{J}_{\boldsymbol{\phi\theta}}$ exists, i.e., when $\boldsymbol{J}_{\boldsymbol{\eta\phi}}$ is *non-singular*. Physically, this means that the model is valid only when the manipulator is not at a *gain-type* singularity. Further, integration of this model only updates the active variable, $\boldsymbol{\theta}$, while the matrices $\boldsymbol{M}_{\boldsymbol{\theta}}, \boldsymbol{C}_{\boldsymbol{\theta}}$ etc. are dependent on $\boldsymbol{\theta}$ as well as $\boldsymbol{\phi}$. Therefore, to complete the update of the model, every integration step has to be followed by a step of forward kinematics, so as to update the passive variables $\boldsymbol{\phi}$. From an implementation point of view, this also introduces another complexity: while computing the passive variables at any time step, one has to ensure that the *branch of solution* chosen (among the possible 6) is the same as the original, i.e., at the initial time. This *branch-following* in forward kinematics is essential in successfully implementing the equivalent model.

4 Formulation of the control scheme

In this section, we develop the control scheme that enables the robot to pass through a singularity in a given trajectory. Before we present the mathematical equations, a brief description of the scheme may be in order.

4.1 Overview of the scheme for singularity avoidance

The scheme for singularity avoidance used in this paper can be thought of as a special case of resolution of kinematic redundancy by the introduction of a secondary task [4, 10]. The "redundancy" in this case is artificial, i.e., the manipulator is *not* redundant inherently; the redundancy is made to appear by constraining the commanded task to a subspace of the manipulator's workspace, whose dimension in lower than the degree-of-freedom of the manipulator. In particular, the workspace of this manipulator has 3 dimensions, which can be represented by the variables $\boldsymbol{x} = (x, y, \alpha)^T$. However, the task we consider is to track a *planar trajectory*, i.e., the position variables x, y are specified as $x_d(t), y_d(t)$, $t \in [t_i, t_f]$, while the orientation variable, $\alpha(t)$, is not specified at all. Therefore, it may be assumed that the orientation of the moving triangle is a *redundant degree-of-freedom* in this context. Following [11], this redundancy may be utilised to avoid singularity. We define an *artificial potential* to represent the *distance* from singularities. The orientation is determined such that at each instance, the manipulator *tries to* move in a way so as to maximise this distance. This is the *secondary* task, as defined by [4, 10], and therefore, as per the formulation, it does not disturb the primary task, i.e., tracking of the trajectory, while trying to avoid singularity.

4.2 Mathematical formulation

In the following, we present the details of the control scheme as per the above outline. First, we consider the partitioning of the manipulation task into two subtasks, namely, (a) trajectory tracking, and (b) singularity avoidance.

4.2.1 Task partitioning and trajectory tracking

The overall task of the manipulator is to execute a commanded trajectory, while avoiding singularities. An implicit requirement in this is that the second task does not hamper the first in any manner, i.e., in trying to avoiding singularities, the manipulator should not deviate from

the trajectory. This is the basis of the "task-priority based control" [4, 10], with high priority being associated with the trajectory tracking.

Let us define the first *task* or *manipulation variable* as:

$$\boldsymbol{r}_1 = (x(t), y(t))^T \tag{27}$$

Let the desired trajectories in these variables be defined as $\boldsymbol{r}_{1d} = (x_d(t), y_d(t))^T$.
Let the *tracking error* be defined as:

$$\boldsymbol{e}(t) = \boldsymbol{r}_1 - \boldsymbol{r}_{1d} = (x(t) - x_d(t), y(t) - y_d(t))^T \tag{28}$$

Differentiating $\boldsymbol{r}_1$ with respect to time twice, we get:

$$\dot{\boldsymbol{r}}_1 = \boldsymbol{J}_1\dot{\boldsymbol{\theta}}, \quad \boldsymbol{J}_1 = \frac{\partial \boldsymbol{r}_1}{\partial \boldsymbol{\theta}} \tag{29}$$

$$\ddot{\boldsymbol{r}}_1 = \boldsymbol{J}_1\ddot{\boldsymbol{\theta}} + \dot{\boldsymbol{J}}_1\dot{\boldsymbol{\theta}} \tag{30}$$

From the second equation, the *joint acceleration for the first manipulation task* is obtained as:

$$\ddot{\boldsymbol{\theta}}_t = \boldsymbol{J}_1^{\#}(\ddot{\boldsymbol{r}}_1 - \dot{\boldsymbol{J}}_1\dot{\boldsymbol{\theta}}), \tag{31}$$

where, the subscript 't' has been added to indicate the 'trajectory-tracking' task. Now, let us assume a control scheme, such that the error has the following dynamics:

$$\ddot{\boldsymbol{e}}(t) + \boldsymbol{K}_v\dot{\boldsymbol{e}}(t) + \boldsymbol{K}_p\boldsymbol{e}(t) = \boldsymbol{0}, \tag{32}$$

where, $\boldsymbol{K}_p$ represents the proportional gain and $\boldsymbol{K}_v$ represents the velocity gain. Therefore, we get the acceleration in the first task as:

$$\ddot{\boldsymbol{r}}_1 = \ddot{\boldsymbol{r}}_{1d} - \boldsymbol{K}_v\dot{\boldsymbol{e}}(t) - \boldsymbol{K}_p\boldsymbol{e}(t) \tag{33}$$

Substituting equation (33) in equation (31) we can rewrite the joint acceleration for the first task as:

$$\ddot{\boldsymbol{\theta}}_t = \boldsymbol{J}_1^{\#}\boldsymbol{h}_1, \ \boldsymbol{h}_1 = \ddot{\boldsymbol{r}}_{1d} - \boldsymbol{K}_v\dot{\boldsymbol{e}}(t) - \boldsymbol{K}_p\boldsymbol{e}(t) - \dot{\boldsymbol{J}}_1\dot{\boldsymbol{\theta}} \tag{34}$$

The significance of this value of the joint acceleration is that with this acceleration, the the trajectory tracking error is brought down to zero asymptotically (see equation (28)). However, this part is oblivious of the singularities.

4.2.2 Singularity condition and the artificial potential

As discussed in sections 2 and 3, the formulation of the velocity kinematics as well as dynamics breaks down at a gain-type singularity, which is defined by the vanishing of the following determinant:

$$\det(\boldsymbol{J}_{\boldsymbol{\eta\phi}}) = 0 \tag{35}$$

This equation can be simplified to the following form [3]:

$$\begin{aligned}\det(\boldsymbol{J}_{\boldsymbol{\eta\phi}}) = & ar^3\{\sin(\phi_1-\alpha)\sin(\phi_2-\phi_3) \\ & + \sin(\phi_1-\phi_2)\sin\left(\alpha+\frac{2\pi}{3}-\phi_3\right)\}\end{aligned} \tag{36}$$

As we can safely assume that the passive link lengths are positive for a generic 3-RRR, the singularity condition can be defined independent of the *physical scale* of the manipulator:

$$S(\boldsymbol{\phi}) = 0,$$

where,

$$\begin{aligned}S(\boldsymbol{\phi}) \triangleq & \sin(\phi_1-\alpha)\sin(\phi_2-\phi_3) \\ & + \sin(\phi_1-\phi_2)\sin\left(\alpha+\frac{2\pi}{3}-\phi_3\right)\end{aligned} \tag{37}$$

Consider now the following function [2]:

$$\begin{aligned}P(\boldsymbol{\theta}) &= \frac{1}{ar^3}\sqrt{\det\left(\boldsymbol{J}_{\boldsymbol{\eta\phi}}^T\boldsymbol{J}_{\boldsymbol{\eta\phi}}\right)} \\ &= |S(\boldsymbol{\phi})|\end{aligned} \tag{38}$$

Obviously, $P \geq 0$; i.e., it vanishes at a singularity, and is positive elsewhere. We consider P as a distance from singularity, or equivalently, an artificial potential effective on the manipulator [3]. The gradient of the artificial potential gives the torque due to the second task:

$$\boldsymbol{\tau}_s = k\frac{\partial P(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}},\ k \in \mathbb{R}^+,\ P(\boldsymbol{\theta}) \neq 0 \tag{39}$$

[2] As we can see, $S(\boldsymbol{\phi})$ is an explicit function of the passive variables alone. However, since $\boldsymbol{\phi} = \boldsymbol{\phi}(\boldsymbol{\theta})$, $P(\boldsymbol{\theta})$ is actually an implicitly defined function of $\boldsymbol{\theta}$.

[3] This definition of the distance function is not mathematically rigorous; it is not proven that P increases monotonically in *all* directions as we move away from a singularity. Nonetheless, it seems to work, just as the "measure of manipulability" worked in [4, 10].

where the positive constant k has the dimension of torque (i.e., it is measured in N-m in this paper), and has the role of a *gain* for the second manipulation task. It is *expected* that the effect of this torque would be to take the manipulator away from a singularity. Substituting $\boldsymbol{\tau}_s$ in equation (26), we get:

$$\boldsymbol{M}_{\boldsymbol{\theta}}\ddot{\boldsymbol{\theta}} + \boldsymbol{C}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} = k\frac{\partial P}{\partial \boldsymbol{\theta}} \tag{40}$$

Therefore, the corresponding acceleration is given by:

$$\ddot{\boldsymbol{\theta}}_s = \boldsymbol{M}_{\boldsymbol{\theta}}^{-1}\left(k\frac{\partial P}{\partial \boldsymbol{\theta}} - \boldsymbol{C}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}\right), \tag{41}$$

where, the subscript 's' has been added to indicate the 'singularity avoidance' task. While this acceleration ensures that the manipulator moves away from singularities, it is oblivious of the task of trajectory tracking.

4.2.3 Combined joint acceleration

It is quite obvious from the above discussion that the execution of the combined task requires a combination of the joint accelerations arising out of the two individual subtasks. However, to make sure that the first task is not affected by the second, the following combination is used which ensures the assumed task priority [4, 10]:

$$\ddot{\boldsymbol{\theta}} = \ddot{\boldsymbol{\theta}}_t + \left(\boldsymbol{I}_3 - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1\right)\ddot{\boldsymbol{\theta}}_s \tag{42}$$

Note that any disturbance to the first task due to the second term is nullified by the feedback loop implemented on that task. However, the reverse is not true. Therefore, the singularity avoidance term, $\left(\boldsymbol{I}_3 - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1\right)\ddot{\boldsymbol{\theta}}_s$, does not affect the trajectory tracking *significantly*. It ends up modifying only α suitably, such that singularity is avoided.

Another way to visualise the above scheme is the following. The singularity manifold of the manipulator can be obtained as $g(x, y, \alpha) = 0$. For a given $\alpha = \alpha^*$, the same can be expressed as $m_{\alpha^*}(x, y) = 0$. Consider the given curve to be traced be given as $n(x, y) = 0$. Therefore, it suffices to say that for a point (x, y) on the path, if we can find α^* such that there is no intersection between the curves $m_{\alpha^*}(x, y) = 0$, $n(x, y) = 0$, then the manipulator is at a nonsingular configuration at (x, y, α^*). In practise, it is difficult to find an α^* for each point on the curve m. However, the control scheme described above does that *dynamically*, i.e., as the manipulator traces the commanded curve, it tries to find α^* such that $m_{\alpha^*}(x, y) = 0$ tries to move away from the current position $(x(t), y(t))$. In doing so, it may cause the curve to cut

the commanded path elsewhere, but that is not of any consequence as the process is *local* and *dynamic*.

The theoretical developments described above is illustrated through a series of numerical simulations, as detailed in the following section.

5 Results of numerical simulations

The geometric parameters of the manipulator used in the numerical simulations are as follows:

$$x_{b1} = y_{b1} = 0,\ x_{b2} = b,\ y_{b2} = 0,\ x_{b3} = b/2, y_{b3} = b/\sqrt{3}$$

where the base length, b, is taken as 0.5[4]. The length of the side of the moving platform is taken as $a = b/4$. The length of the active links are $l_1 = l_2 = l_3 = 1/4$ and the lengths of the passive links $r_1 = r_2 = r_3 = 1/6$. The mass and inertia parameters of the links are given as follows: The mass of the active links: $m_1 = m_2 = m_3 = 0.4680$ Kg, mass of the passive links: $m_4 = m_5 = m_6 = 0.3120$ Kg, and the mass of the moving triangle: $m_7 = 0.2340$ Kg. Moments of inertia of the links about the vertical axis passing through respective centres of mass: $I_{c1} = I_{c2} = I_{c3} = 0.0024$ Kg-m^2, $I_{c4} = I_{c5} = I_{c6} = 0.0007$ Kg-m^2, and $I_{c7} = 0.0003$ Kg-m^2. The links are assumed to have uniform mass distribution, and therefore the mass centres coincide with the respective geometric centres.

The numerical simulations pertain to the tracking of a circle in the $\boldsymbol{XY}$ plane. The centre of the circle is at $(0.30, 0.20)$ and its radius is 0.07. It is chosen so that it cuts the singularity curve at two points for the fixed orientation $\alpha = 0$, as seen in figure 2. The contour curves corresponding to $S(\boldsymbol{\phi}) = 0$ delineate the singular loci of the centre of the platform, $\boldsymbol{p}(x, y)$, corresponding to the orientation $\alpha = 0$. The manipulator starts at the point marked by a cross in figure 3. The corresponding configuration of the manipulator is shown in figure 3. The desired trajectory $\boldsymbol{r}_{1d} = (x_d(t), y_d(t))^T$ is planned as a pair of cubic polynomials in time so that the circle is traced clockwise with a mean speed of 0.2m/s, with zero initial and final speeds. The tracked path is shown in figure 4. The errors in tracking, in the $\boldsymbol{X}$ and $\boldsymbol{Y}$ directions respectively, are plotted in figure 5. The control gains used in this case were:

$$\boldsymbol{K}_p = k_p \boldsymbol{I}_2, \quad \boldsymbol{K}_v = 2\sqrt{k_p}\boldsymbol{I}_2$$

where $\boldsymbol{I}_2$ is the 2×2 identity matrix, and $k_p = 1000$. The value of k was 115, and it was chosen through numerical experiments. It can be seen from figures 4, 5 that the tracking performance

[4]All the linear entities are expressed in metres and all angles in radians, unless specified otherwise.

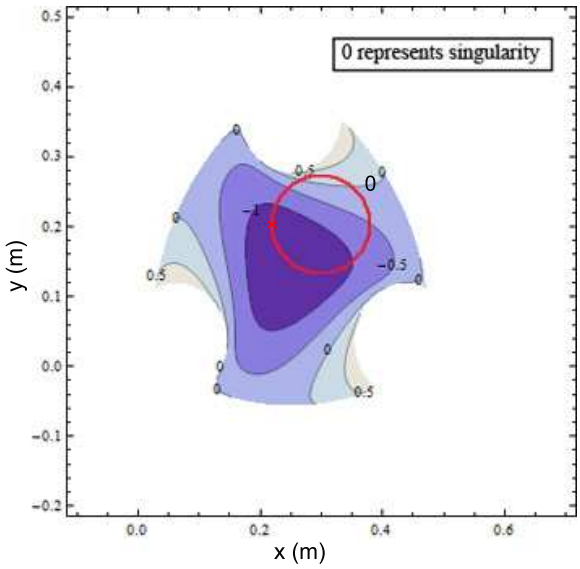


Figure 2: The desired path superposed on the contours of the singularity function, $S(\boldsymbol{\phi})$ for the orientation $\alpha = 0$

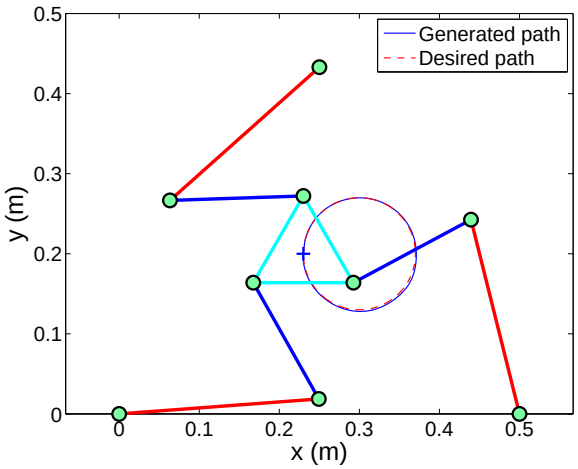


Figure 3: Initial configuration of the 3-RRR: $x = 0.2300,\ y = 0.2000,\ \alpha = 0$

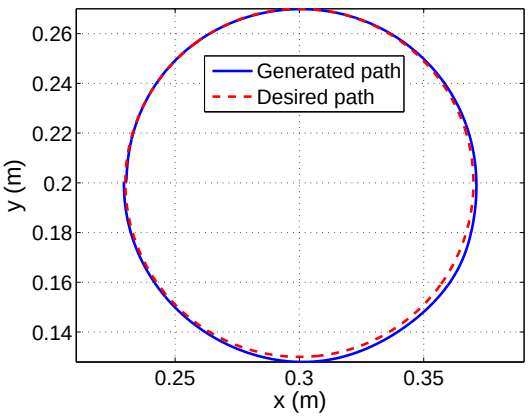


Figure 4: The path tracked vs. the desired path

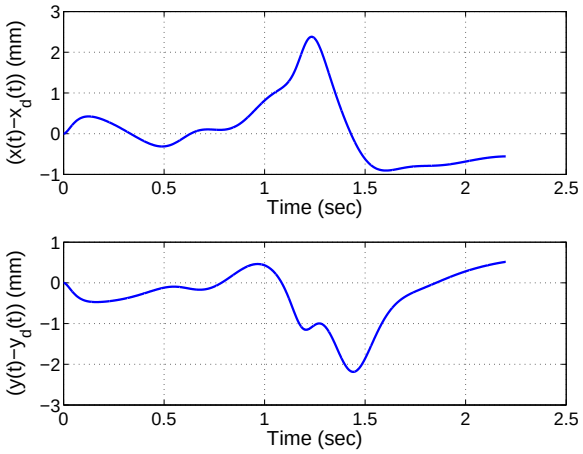


Figure 5: Errors in path-tracking

is quite satisfactory even with a linear control law. It can attributed greatly to the fact that the nonlinearities in the plant are assumed to be known *exactly*, such that they could be cancelled off completely.

The effect of the singularities in the desired path can be seen in figure 6. The orientation α goes through significant change in order to keep the manipulator away from singularity. In

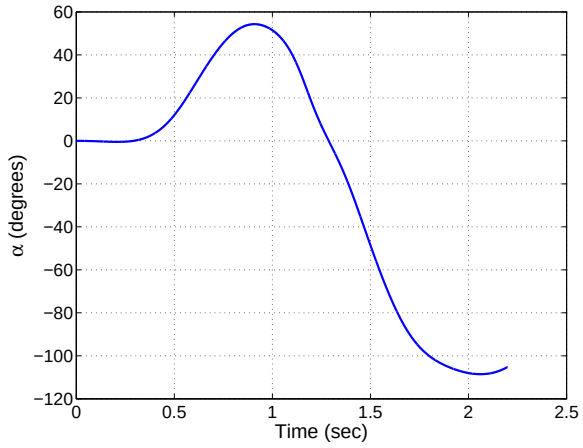


Figure 6: Variation of α along the trajectory

doing so, the manipulator manages to keep $S(\boldsymbol{\phi})$ away from zero, as can be seen in figure 7. For the sake of comparison, the variation of $S(\boldsymbol{\phi})$ along the trajectory with $\alpha = 0$ is also given, which is seen to cross the zero line at two places, in conformity with the observations made on figure 2. The variation of the control torques for the three joints respectively are shown in figure 8. It shows the total torque as well as the contribution from the singularity avoidance term, $\boldsymbol{\tau}_s$, alone. It is interesting to note that the task of following the path, which originally contained singularities, is accomplished with finite control efforts at the active joints.

We now present a series of figures, namely 9 and 10 to substantiate the claims made in

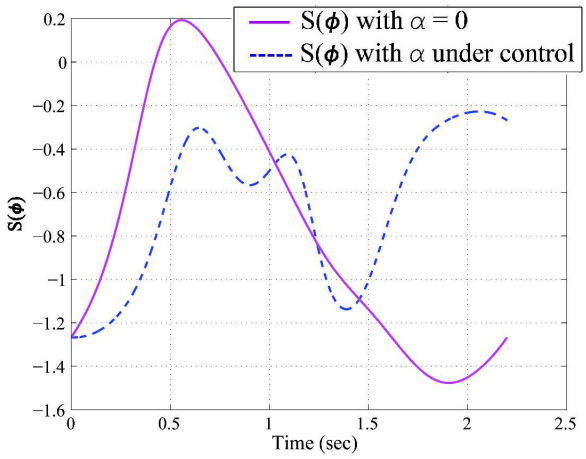


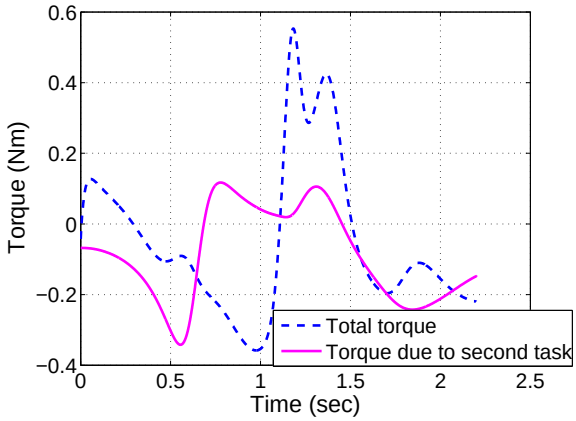
Figure 7: Variation of $S(\boldsymbol{\phi})$ along the trajectory

the second paragraph of section 4.2.3. These figures show the desired path, the location of $\boldsymbol{p}$ on it, and the contours of $S(\boldsymbol{\phi})$ at the corresponding orientations α. As can be seen from these, as the target point moves on the circle, the zero-valued contour-curves of the singularity function move away from it, thus reinforcing the intuitive understanding of the control scheme as explained earlier. The above results establish the feasibility of the control scheme proposed for trajectory-tracking with singularity avoidance.

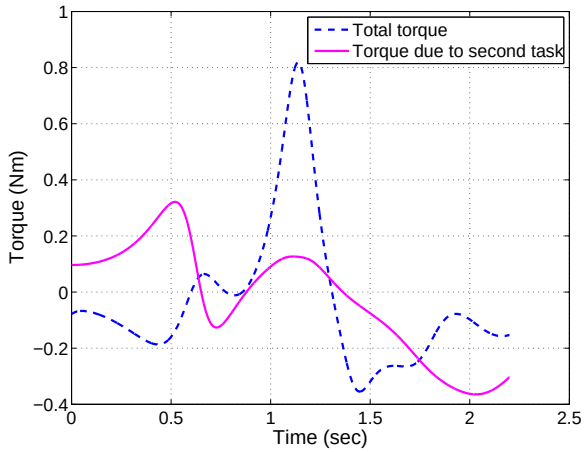
6 Conclusion

In this paper, we have presented a task-priority-based control scheme for trajectory-tracking control of a planar parallel manipulator, with singularity avoidance. It is proposed that by allowing the platform orientation to change suitably under the control scheme, the manipulator may be able to track planar trajectories which would have been singular if the original orientation were to be held fixed. The numerical simulations demonstrate the feasibility of the scheme, albeit with the caveat that in addition to the PD control gains, a new "gain" parameter, k, needs to be tuned for the success of the scheme.

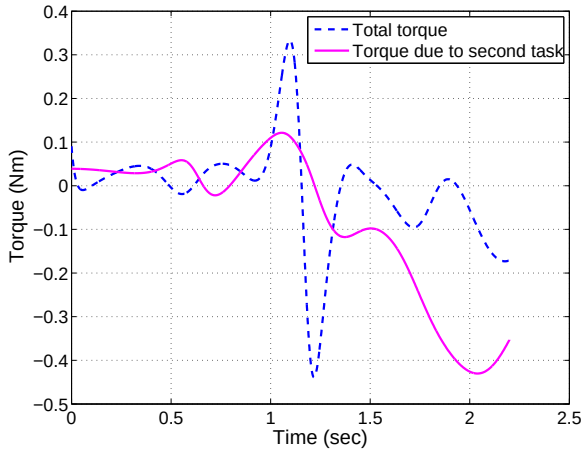
Although the results presented here pertain solely to the planar 3-RRR manipulator, the formulation is fairly general. As such, it should be possible, in theory, to apply it to *any* parallel or hybrid manipulator, whenever the manipulator is required to execute a task in a space of at least one dimension lower than its degree-of-freedom.



(a) Variation of τ_1

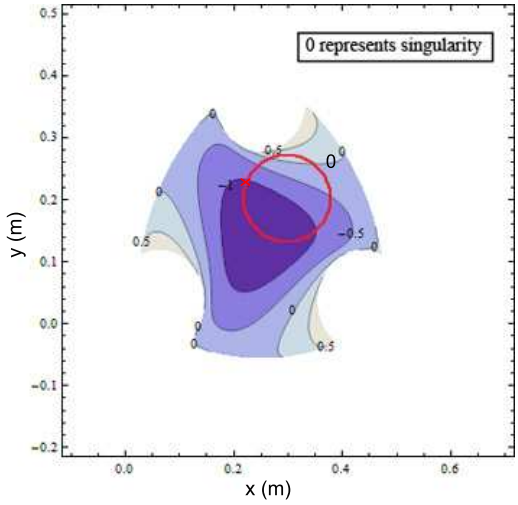


(b) Variation of τ_2

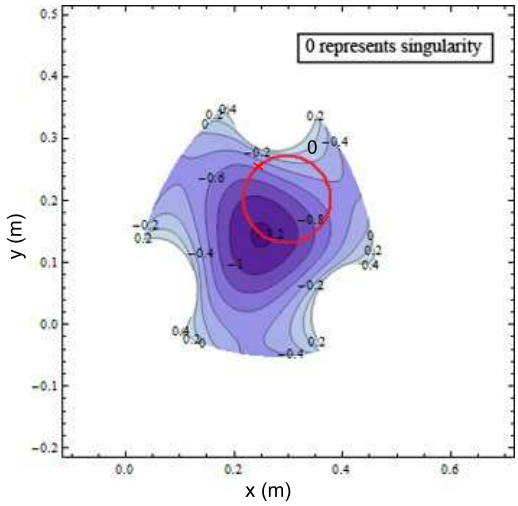


(c) Variation of τ_3

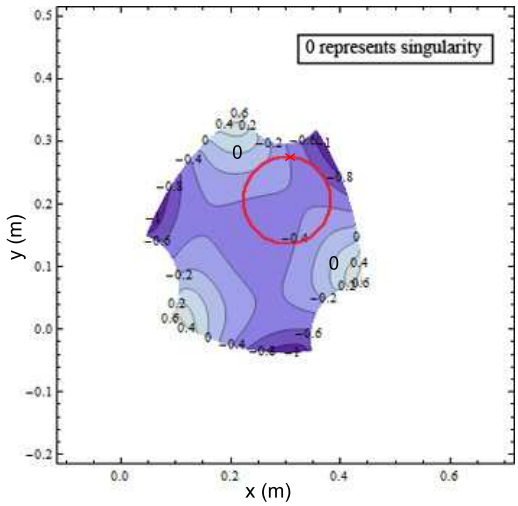
Figure 8: Variation of the control torques along the trajectory



(a) x = 0.2325, y = 0.2300, $\alpha = -0.0033$

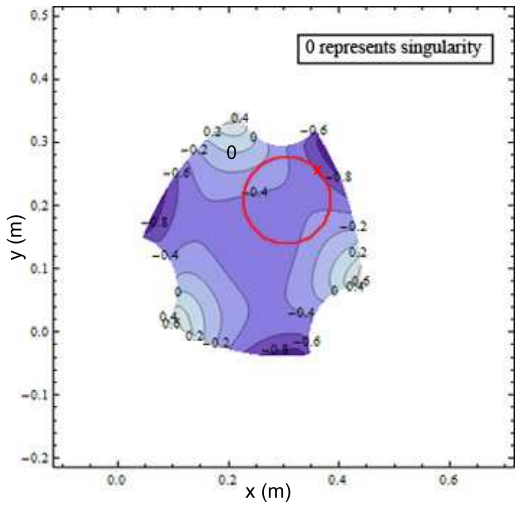


(b) x = 0.2611, y = 0.2560, $\alpha = 0.3196$

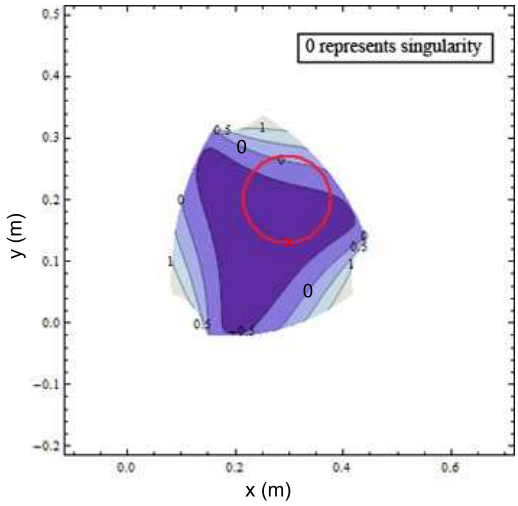


(c) x = 0.2992, y = 0.2670, $\alpha = 0.7244$

Figure 9: Evolution of the contours of the singularity function along the trajectory



(a) x = 0.3700, y = 0.2420, $\alpha = 0.7050$



(b) x = 0.2852, y = 0.1350, $\alpha = -1.0207$

Figure 10: Evolution of the contours of the singularity function along the trajectory (contd.)

References

[1] C. Gosselin and J. Angeles, "Singularity analysis of closed loop kinematic chains," *IEEE Transactions on Robotics and Automation*, vol. 6, pp. 281–290, June 1990.

[2] I.A.Bonev and C.Gosselin, "Singularity loci of planar parallel manipulators with revolute joints," in *Proceedings of 2nd Workshop on Computational Kinematics*, pp. 291–299, May 2001.

[3] S. Bandyopadhyay and A. Ghosal, "Analysis of configuration space singularities of closed-loop mechanisms and parallel manipulators," *Mechanism and Machine Theory*, vol. 39, pp. 519–544, May 2004.

[4] H. H. Yoshihiko Nakamura and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *International Journal of Robotics Research*, vol. 6, no. 2, pp. 3–15, 1987.

[5] S.-H. Cha, T. A.Lasky, and S. A.Velinsky, "Singularity avoidance for the 3-RRR mechanism using kinematics redundancy," in *Proceedings of the 2007 ICRA, Roma, Italy*, pp. 1195–1200, April 2007.

[6] S. Bandyopadhyay and A. Ghosal, "Geometric characterization and parametric representation of the singularity manifold of a 6-6 Stewart platform manipulator," *Mechanism and Machine Theory*, vol. 41, pp. 1377–1400, Nov. 2006.

[7] A. Ghosal, *Robotics: Fundamental Concepts and Analysis*. New Delhi: Oxford University Press, 2006.

[8] W. wei Shang, S. Cong, and S. long Jiang, "Dynamic model based nonlinear tracking control of a planar parallel manipulator," *Nonlinear Dynamics*, vol. 60, no. 4, pp. 597–606, 2009.

[9] C. Nasa and S. Bandyopadhyay, "Trajectory-tracking control of a planar 3-$\underline{\text{R}}$RR parallel manipulator with singularity avoidance," in *International Conference on Multi Body Dynamics, Vijaywada, India*, pp. 35–52, February 2011.

[10] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Addision-Wesley Publishing Company, Inc., 1991.

[11] H. H. Yoshihiko Nakamura and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *International Journal of Robotics Research*, vol. 6, no. 2, pp. 3–15, 1987.