Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 4267 – 4283

doi:10.3906/elk-1502-50

A switching inverse dynamics controller for parallel manipulators around drive singular configurations

Mustafa ÖZDEMİR[1,*], Sıtkı Kemal İDER[2]

[1]Department of Mechanical Engineering, Faculty of Engineering, Middle East Technical University, Ankara, Turkey
[2]Department of Mechanical Engineering, Faculty of Engineering, Çankaya University, Ankara, Turkey

Received: 08.02.2015 • **Accepted/Published Online:** 29.07.2015 • **Final Version:** 20.06.2016

Abstract: Despite many advantages, parallel manipulators are known to possess drive singularities where the control of one or more degrees of freedom is lost. Around these singular configurations, the required actuator forces grow unbounded. Previous efforts in the literature put forward singularity-consistent trajectory planning and singularity-robust modification of the dynamic equations as a solution to this problem. However, this previous method is applicable only for the open-loop operation of the manipulator, whereas initial configuration errors, external disturbances, and modeling errors should necessarily be taken into account in a closed-loop sense in real-life applications. With this aim, a switching inverse dynamics controller is proposed in this study for the trajectory tracking control of parallel manipulators as they pass through drive singular configurations. Simulations of the application of the developed controller result in good tracking performance, even in the presence of modeling errors, while the actuator efforts remain bounded and continuous in the neighborhood of the singularity.

Key words: Parallel manipulators, inverse dynamics control, drive singularities

1. Introduction

When compared to conventional robotic arms, parallel manipulators offer definite advantages due to their closed-loop architecture, such as faster and more precise positioning ability, higher load-carrying capacity, and higher end-effector accelerations. However, their closed-loop structure also results in their main weakness: drive singularities encountered during their inverse dynamics solution, around which the required actuator forces tend to infinity, and hence the desired trajectory becomes impossible to accomplish by the manipulator.

Gosselin and Angeles [1] performed a detailed analysis of the singularities of closed-loop kinematic chains, and showed that this type of singularities characteristically arises inside the workspace, as opposed to the singularities that are encountered during the inverse kinematics solutions. For this reason, the usable workspace of parallel manipulators is dramatically small. Therefore, although not controllable at drive singularities [2], it is needed to develop methods for moving parallel manipulators through these positions in a controllable manner.

Özgören [3] studied the trajectory tracking control of closed-loop kinematic chains, and proposed a control approach that allows a deviation from the desired trajectory around drive singularities. Jui and Sun [4] devised a method to overcome the problem of unbounded actuator forces at a drive singularity by relaxing the timing of the path.

Ider [5,6] showed that a parallel manipulator can pass through its drive singular configurations while the

*Correspondence: mozdemir@metu.edu.tr

required actuator forces remain finite if the trajectory is planned such that its dynamic equations of motion are consistent at these singularities. He further showed that the dynamic equations can be modified to avoid ill-conditioning in their neighborhood.

Briot and Arakelian [7] studied the dynamical interpretation of the necessary conditions for parallel manipulators to pass through drive singularities, and derived also the associated conditions for rigid-link flexible-joint [8] and flexible-link [9] cases.

However, the suggested technique of modification of the dynamic equations of motion to obtain a full-rank system following a singularity-consistent planning of the desired trajectory considered the operation of the manipulator in an open-loop sense, and thus does not take into account initial configuration errors, external disturbances, and modeling errors that are inevitable in real-life applications. With this aim, a switching inverse dynamics controller is proposed in this study for the trajectory tracking control problem of parallel manipulators passing through drive singularities. While the conventional inverse dynamics controller is used at configurations away from singular positions, a new inverse dynamics control law is developed to be switched in their neighborhoods, based on the "consistency conditions" and the "modified equations" of Ref. [5], in order to avoid the control inputs becoming unbounded. As a numerical example, a 3 degree of freedom RPR-RPR planar parallel manipulator is considered, and the derived control law is tested in the presence of initial positioning errors and modeling errors.

2. Preliminary formulations

Consider an n degree of freedom nonredundant parallel manipulator. Let m denote the degrees of freedom of the open-tree structure obtained by virtually disconnecting a sufficient number of unactuated joints of the robot. Without loss of generality, the joint variables vector can be arranged as $\boldsymbol{q}^T = \left[\begin{array}{cc} \boldsymbol{q}^{a^T} & \boldsymbol{q}^{u^T} \end{array} \right]$, where $\boldsymbol{q}^a = \left[\begin{array}{ccc} q_1 & \cdots & q_n \end{array} \right]^T$ is the vector of the joint variables of the actuated joints and $\boldsymbol{q}^u = \left[\begin{array}{ccc} q_{n+1} & \cdots & q_m \end{array} \right]^T$ is the vector of the joint variables of the unactuated joints. The $(m-n)$ velocity-level loop-closure constraints can be written as

$$\boldsymbol{\Phi}\dot{\boldsymbol{q}} = \mathbf{0}_{(m-n)\times 1}, \tag{1}$$

where $\boldsymbol{\Phi} = \boldsymbol{\Phi}(\boldsymbol{q})$ is the $(m-n)\times m$ constraint Jacobian matrix, and $\dot{\boldsymbol{q}}$ is the m-dimensional vector of the joint velocities.

The equations of motion of the parallel manipulator can be obtained in terms of the joint variables $\mathbf{q}$ and the generalized constraint forces as

$$\mathbf{M}\ddot{\boldsymbol{q}} + \boldsymbol{B} = \left[\begin{array}{c} \boldsymbol{\tau} \\ \mathbf{0}_{(m-n)\times 1} \end{array} \right] + \boldsymbol{\Phi}^T\boldsymbol{\lambda} \tag{2}$$

where $\mathbf{M}(\boldsymbol{q})$ is the $m\times m$ generalized inertia matrix of the open-tree structure, which is known to be positive-definite, i.e. nonsingular [10], $\boldsymbol{B}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is the m-dimensional vector combining the nonlinear terms, $\boldsymbol{\lambda}$ is the $(m-n)$-dimensional vector of constraint reaction forces, $\boldsymbol{\tau}$ is the n-dimensional vector of actuator forces, and $\ddot{\boldsymbol{q}}$ is the m-dimensional vector of the joint accelerations.

In order to eliminate the intermediate variables and to obtain the input–output relation, the matrices $\mathbf{M}$, $\boldsymbol{B}$, and $\boldsymbol{\Phi}$ are partitioned as follows:

$$\mathbf{M} = \left[\begin{array}{c} \mathbf{M}^a_{n\times m} \\ \mathbf{M}^u_{(m-n)\times m} \end{array} \right] \tag{3}$$

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}^a_{n\times 1} \\ \boldsymbol{B}^u_{(m-n)\times 1} \end{bmatrix} \tag{4}$$

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}^a_{(m-n)\times n} & \boldsymbol{\Phi}^u_{(m-n)\times(m-n)} \end{bmatrix}, \tag{5}$$

where the superscripts a and u denote the partitions associated with the actuated and unactuated joints, respectively, and the sizes of the matrices and the vectors are shown as subscripts. In terms of the above partitions, Eq. (2) can be divided into the following two equations:

$$\boldsymbol{\tau} = \mathbf{M}^a\ddot{\boldsymbol{q}} + \boldsymbol{B}^a - \left(\boldsymbol{\Phi}^a\right)^T\boldsymbol{\lambda} \tag{6}$$

$$\left(\boldsymbol{\Phi}^u\right)^T\boldsymbol{\lambda} = \mathbf{M}^u\ddot{\boldsymbol{q}} + \boldsymbol{B}^u \tag{7}$$

At a drive singularity, $\boldsymbol{\Phi}^u$ is singular. When $\boldsymbol{\Phi}^u$ is nonsingular, $\boldsymbol{\lambda}$ is obtained as

$$\boldsymbol{\lambda} = \left[\left(\boldsymbol{\Phi}^u\right)^T\right]^{-1}\left(\mathbf{M}^u\ddot{\boldsymbol{q}} + \boldsymbol{B}^u\right) \tag{8}$$

$\boldsymbol{\tau}$ can then be determined substituting this expression into Eq. (6):

$$\boldsymbol{\tau} = \mathbf{M}'\ddot{\boldsymbol{q}} + \boldsymbol{B}' \tag{9}$$

where the $n \times m$ matrix $\mathbf{M}'$ is

$$\mathbf{M}' = \mathbf{M}^a - \left(\boldsymbol{\Phi}^a\right)^T\left[\left(\boldsymbol{\Phi}^u\right)^T\right]^{-1}\mathbf{M}^u \tag{10}$$

and the n-dimensional vector $\boldsymbol{B}'$ is

$$\boldsymbol{B}' = \boldsymbol{B}^a - \left(\boldsymbol{\Phi}^a\right)^T\left[\left(\boldsymbol{\Phi}^u\right)^T\right]^{-1}\boldsymbol{B}^u \tag{11}$$

The task equations at velocity-level can be expressed as

$$\boldsymbol{\psi}\dot{\boldsymbol{q}} = \dot{\boldsymbol{x}}, \tag{12}$$

where $\boldsymbol{x} = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T$ denotes the task-space coordinates of the end-effector, and $\boldsymbol{\psi} = \boldsymbol{\psi}(\boldsymbol{q})$ is the $n \times m$ Jacobian matrix of the task equations. Combining the time derivatives of Eqs. (1) and (12), one obtains [5,6]

$$\boldsymbol{\Gamma}\ddot{\boldsymbol{q}} = -\dot{\boldsymbol{\Gamma}}\dot{\boldsymbol{q}} + \begin{bmatrix} \mathbf{0}_{(m-n)\times 1} \\ \ddot{\boldsymbol{x}} \end{bmatrix}, \tag{13}$$

where the $m \times m$ matrix $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}(\boldsymbol{q})$ is defined as

$$\boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{\Phi} \\ \boldsymbol{\psi} \end{bmatrix} \tag{14}$$

As long as $\boldsymbol{\Gamma}$ is nonsingular, one can solve Eq. (13) for $\ddot{\boldsymbol{q}}$. The singularity of the matrix $\boldsymbol{\Gamma}$, on the other hand, corresponds to the inverse kinematic singularities [5,6]. In the present study, the desired trajectory is

assumed to be such that no inverse kinematic singularity is encountered during its realization, i.e. $\mathbf{\Gamma}$ is always invertible. This assumption does not yield any loss in the practical applicability of the formulations given here, since inverse kinematic singularities are encountered in general at the boundaries of the workspace only [1]. Eq. (13) can then be solved for $\ddot{\boldsymbol{q}}$ as

$$\ddot{\boldsymbol{q}} = \mathbf{\Gamma}^{-1}\left(-\dot{\mathbf{\Gamma}}\dot{\boldsymbol{q}} + \begin{bmatrix} \mathbf{0}_{(m-n)\times 1} \\ \ddot{\boldsymbol{x}} \end{bmatrix}\right) \tag{15}$$

Substituting Eq. (15) into Eq. (9), one obtains

$$\boldsymbol{\tau} = \mathbf{M}^{*}\begin{bmatrix} \mathbf{0}_{(m-n)\times 1} \\ \ddot{\boldsymbol{x}} \end{bmatrix} + \boldsymbol{B}^{*}, \tag{16}$$

where the $n \times m$ matrix $\mathbf{M}^{*}$ is

$$\mathbf{M}^{*} = \mathbf{M}'\mathbf{\Gamma}^{-1} \tag{17}$$

and the n-dimensional vector $\boldsymbol{B}^{*}$ is

$$\boldsymbol{B}^{*} = \boldsymbol{B}' - \mathbf{M}'\mathbf{\Gamma}^{-1}\dot{\mathbf{\Gamma}}\dot{\boldsymbol{q}} \tag{18}$$

In Eq. (16), all of the elements in the first $m-n$ columns of the matrix $\mathbf{M}^{*}$ are multiplied by zero. This fact suggests utilizing a partitioning of the matrix $\mathbf{M}^{*}$ as follows:

$$\mathbf{M}^{*} = \begin{bmatrix} \mathbf{M}^{*^{G}}_{n\times(m-n)} & \mathbf{M}^{*^{P}}_{n\times n} \end{bmatrix} \tag{19}$$

Based on this partitioning, the relation between the input $\boldsymbol{\tau}$ and the output $\boldsymbol{x}$ can be obtained in the following form:

$$\boldsymbol{\tau} = \mathbf{M}^{*^{P}}\ddot{\boldsymbol{x}} + \boldsymbol{B}^{*} \tag{20}$$

An inverse dynamics control law can then be formulated via Eq. (20) to be used outside the neighborhood of drive singularities. In the control law, command accelerations $\boldsymbol{u}$ have to be specified. For example, using PD control, $\boldsymbol{u}$ can be generated as

$$\boldsymbol{u} = \ddot{\boldsymbol{x}}^{d} + \mathbf{C}_1\left(\dot{\boldsymbol{x}}^{d} - \dot{\boldsymbol{x}}\right) + \mathbf{C}_2\left(\boldsymbol{x}^{d} - \boldsymbol{x}\right), \tag{21}$$

where superscript d denotes desired values and $\mathbf{C}_1$ and $\mathbf{C}_2$ are constant feedback gain diagonal matrices. Then $\boldsymbol{\tau}$ can be computed through Eq. (20) as

$$\boldsymbol{\tau} = \mathbf{M}^{*^{P}}\boldsymbol{u} + \boldsymbol{B}^{*} \tag{22}$$

The invertibility of the $\mathbf{M}^{*^{P}}$ matrix appearing in the control law should also be investigated. Under the assumption of no inverse kinematic singularity, the $\mathbf{\Gamma}$ matrix, and hence the $\mathbf{\Phi}$ matrix, has always full rank. Thus, the constraint equations are always linearly independent. In this case, the existence and uniqueness of a forward dynamic solution, $\ddot{\boldsymbol{x}}$, of the constrained manipulator are guaranteed for a given $\boldsymbol{\tau}$ [11,12]. In this paper, to obtain the relation between the input $\boldsymbol{\tau}$ and the output $\ddot{\boldsymbol{x}}$, i.e. Eq. (20), the intermediate variables ($\boldsymbol{\lambda}$ and $\ddot{\boldsymbol{q}}$) are eliminated outside the neighborhoods of drive singularities ($\mathbf{\Phi}^{u}$ is nonsingular) under the assumption of no inverse kinematic singularity ($\mathbf{\Gamma}$ is nonsingular). Since singularity of the $\mathbf{M}^{*^{P}}$ matrix would violate the existence and uniqueness of $\ddot{\boldsymbol{x}}$ in Eq. (20) for a given $\boldsymbol{\tau}$, one can conclude that the $\mathbf{M}^{*^{P}}$

matrix is always nonsingular outside the neighborhoods of drive singularities under the assumption of no inverse kinematic singularity.

Hence, in the absence of modeling errors and disturbances, the application of the actuator forces given by Eq. (22) results in actual accelerations equal to the command accelerations, i.e.

$$\ddot{\boldsymbol{x}} = \boldsymbol{u} \tag{23}$$

As a result, Eqs. (21) and (23) lead to the following error equation:

$$\ddot{\boldsymbol{e}} + \mathbf{C}_1\dot{\boldsymbol{e}} + \mathbf{C}_2\boldsymbol{e} = \mathbf{0}_{n\times 1}, \tag{24}$$

where $\boldsymbol{e} = \boldsymbol{x}^d - \boldsymbol{x}$, and $\dot{\boldsymbol{x}}^d$ and $\ddot{\boldsymbol{x}}^d$ are the desired velocity and acceleration profiles of the end-effector, respectively.

The form of the error dynamics will depend on the selection of $\mathbf{C}_1$ and $\mathbf{C}_2$. Based on this error equation, the feedback gain matrices can be selected appropriately to ensure asymptotic stability.

At this point, it will be useful to discuss also the effects of the modeling errors and disturbances on the error dynamics. For this purpose, let the estimates of the actual $\mathbf{M}^{*^P}$ matrix and the actual $\boldsymbol{B}^*$ vector used in the dynamic model be denoted by $\mathbf{M}^{*^P}_{model}$ and $\boldsymbol{B}^*_{model}$, respectively, and let the external disturbances acting on the system be represented by an n-dimensional vector $\boldsymbol{D}(t)$. Hence, the application of the control torque and forces computed through the inverse dynamics control law results in the following equation:

$$\mathbf{M}^{*^P}_{model}\boldsymbol{u} + \boldsymbol{B}^*_{model} = \mathbf{M}^{*^P}\ddot{\boldsymbol{x}} + \boldsymbol{B}^* + \boldsymbol{D} \tag{25}$$

By rearranging Eq. (25), one can express the governing error dynamics in such a case as

$$\ddot{\boldsymbol{e}} + \mathbf{C}_1\dot{\boldsymbol{e}} + \mathbf{C}_2\boldsymbol{e} = \boldsymbol{\varepsilon}, \tag{26}$$

where the n-dimensional vector $\boldsymbol{\varepsilon}$ is given by

$$\boldsymbol{\varepsilon} = \left(\mathbf{M}^{*^P}_{model}\right)^{-1}\left(\Delta\mathbf{M}^{*^P}\ddot{\boldsymbol{x}} + \Delta\boldsymbol{B}^* + \boldsymbol{D}\right) \tag{27}$$

Here

$$\Delta\mathbf{M}^{*^P} = \mathbf{M}^{*^P} - \mathbf{M}^{*^P}_{model} \tag{28}$$

and

$$\Delta\boldsymbol{B}^* = \boldsymbol{B}^* - \boldsymbol{B}^*_{model} \tag{29}$$

Asymptotic stability of the system can be achieved by selecting the diagonal feedback gain matrices simply as

$$\mathbf{C}_1 = diag\left([2\zeta_1\omega_1, 2\zeta_2\omega_2, \cdots, 2\zeta_n\omega_n]\right) \tag{30}$$

and

$$\mathbf{C}_2 = diag\left(\left[\omega_1^2, \omega_2^2, \cdots, \omega_n^2\right]\right), \tag{31}$$

where ω_i and ζ_i $(i = 1, 2, \cdots, n)$ are all positive constants. Hence, by choosing ω_i large enough for all i, $\boldsymbol{e}$ can be reduced. This is because in a practical situation $\boldsymbol{\varepsilon}$ varies much more slowly, i.e. it is almost constant, with

respect to a sinusoidal function even for the smallest one of the ω_i's. Therefore, after the transient phase, the error converges to the value $\bar{\boldsymbol{e}}$ given below, which gets smaller as the elements of $\mathbf{C}_2$, i.e. ω_i's, are increased:

$$\bar{\boldsymbol{e}} = (\mathbf{C}_2)^{-1}\,\boldsymbol{\varepsilon} \tag{32}$$

It is also concluded in Ref. [13] that, in the presence of modeling errors, inverse dynamics control can be quite good if the feedback gains are selected large enough.

As a last note, before moving on to the next section, it should be recalled that when $|\boldsymbol{\Phi}^u| = 0$, drive singularities occur, in the vicinity of which the required actuator forces in general diverge to infinity [3,5], and hence the actuators unavoidably saturate during the realization of the conventional inverse dynamics control law. To remedy this, a new inverse dynamics control law is proposed in the next section.

3. Proposed control law

The "consistency conditions" that the desired trajectory should satisfy at drive singularities in order to be realizable by the manipulator and the "modified equations" to be used to replace the linearly dependent equations of motion in the neighborhood of these singularities to prevent ill-conditioned inverse dynamics solution were previously derived in Ref. [5] by considering the combined coefficient matrix of the actuator forces and the constraint forces. However, as should be apparent from Eq. (7), the rank deficiency of $\boldsymbol{\Phi}^u$ in this equation system is solely responsible for any drive singular configuration encountered. For this reason, by following derivation steps similar to those of Ref. [5], i.e. by differentiating Eq. (7) and expressing the linear dependency relations among the rows of the $(\boldsymbol{\Phi}^u)^T$ matrix at the singular configuration as $\Phi_{s_k j}^{u^T} = \sum_{p=1,(p\neq s_k)}^{m-n} \alpha_{kp}\Phi_{pj}^{u^T}$, $j = 1,\cdots,m-n$, $k = 1,\cdots,m-n-r$, the "modified equations" can be revised as given below, and these revised equations can be used to replace the linearly dependent rows of the matrix Eq. (7) in the neighborhood of drive singularities for avoidance of ill-conditioning.

$$\begin{aligned}
&\sum_{j=1}^{m-n}\left[\dot{\Phi}_{s_k j}^{u^T} - \sum_{p=1,(p\neq s_k)}^{m-n}\left(\alpha_{kp}\dot{\Phi}_{pj}^{u^T} + \dot{\alpha}_{kp}\Phi_{pj}^{u^T}\right)\right]\lambda_j = \\
&\sum_{i=1}^{m}\left\{\left(M_{s_k i}^u - \sum_{p=1,(p\neq s_k)}^{m-n}\alpha_{kp}M_{pi}^u\right)\dddot{\boldsymbol{q}}_i + \left[\dot{M}_{s_k i}^u - \sum_{p=1,(p\neq s_k)}^{m-n}\left(\alpha_{kp}\dot{M}_{pi}^u + \dot{\alpha}_{kp}M_{pi}^u\right)\right]\ddot{q}_i\right\} + \\
&\dot{B}_{s_k}^u - \sum_{p=1,(p\neq s_k)}^{m-n}\left(\alpha_{kp}\dot{B}_p^u + \dot{\alpha}_{kp}B_p^u\right), k = 1,\cdots,m-n-r
\end{aligned} \tag{33}$$

Here, similar to the notation of Ref. [5], r, $s_k\,(k = 1,\cdots,m-n-r)$, and α_{kp} are used to denote the rank of the $(\boldsymbol{\Phi}^u)^T$ matrix, its linearly dependent row indices, and the coefficients of the corresponding linear dependency relations, respectively, at the singular position.

After the replacements, as described above, the resulting system of equations is

$$\left(\hat{\boldsymbol{\Phi}}^u\right)^T\boldsymbol{\lambda} = \hat{\mathbf{N}}^u\dddot{\boldsymbol{q}} + \hat{\mathbf{M}}^u\ddot{\boldsymbol{q}} + \hat{\boldsymbol{B}}^u, \tag{34}$$

where $\dddot{\boldsymbol{q}}$ denotes the vector of the joint jerks, the s_kth rows $(k = 1,\cdots,m-n-r)$ of the $(m-n)\times m$ matrix $\hat{\mathbf{N}}^u(\boldsymbol{q})$ consist of the coefficients of $\dddot{q}_i$'s in Eqs. (33) while its remaining r rows are zero rows. Hence, the rank

of $\hat{\mathbf{N}}^u$ is $m-n-r$. $\left(\hat{\mathbf{\Phi}}^u\right)^T(\boldsymbol{q},\dot{\boldsymbol{q}})$ and $\hat{\mathbf{M}}^u(\boldsymbol{q},\dot{\boldsymbol{q}})$ are obtained by replacing the s_kth rows $(k=1,\cdots,m-n-r)$ of $(\mathbf{\Phi}^u)^T$ and $\mathbf{M}^u$ with the rows consisting of the coefficients of λ_j's and $\ddot{q}_i$'s in Eqs. (33), respectively, and $\hat{\boldsymbol{B}}^u(\boldsymbol{q},\dot{\boldsymbol{q}})$ is obtained by replacing the s_kth elements $(k=1,\cdots,m-n-r)$ of $\boldsymbol{B}^u$ with the remaining terms in Eqs. (33).

Eq. (34) can be employed to determine $\boldsymbol{\lambda}$ in the neighborhood of drive singularities, as

$$\boldsymbol{\lambda}=\left[\left(\hat{\mathbf{\Phi}}^u\right)^T\right]^{-1}\left(\hat{\mathbf{N}}^u\ddot{\boldsymbol{q}}+\hat{\mathbf{M}}^u\ddot{\boldsymbol{q}}+\hat{\boldsymbol{B}}^u\right), \tag{35}$$

where $\hat{\mathbf{\Phi}}^u$ has in general full rank when the system is in motion. Then $\boldsymbol{\tau}$ can be determined by substituting Eq. (35) into Eq. (6):

$$\boldsymbol{\tau}=\hat{\mathbf{N}}'\ddot{\boldsymbol{q}}+\hat{\mathbf{M}}'\ddot{\boldsymbol{q}}+\hat{\boldsymbol{B}}', \tag{36}$$

where the $n\times m$ matrices $\hat{\mathbf{N}}'$ and $\hat{\mathbf{M}}'$ are

$$\hat{\mathbf{N}}'=-\left(\mathbf{\Phi}^a\right)^T\left[\left(\hat{\mathbf{\Phi}}^u\right)^T\right]^{-1}\hat{\mathbf{N}}^u \tag{37}$$

$$\hat{\mathbf{M}}'=\mathbf{M}^a-\left(\mathbf{\Phi}^a\right)^T\left[\left(\hat{\mathbf{\Phi}}^u\right)^T\right]^{-1}\hat{\mathbf{M}}^u \tag{38}$$

and the n-dimensional vector $\hat{\boldsymbol{B}}'$ is

$$\hat{\boldsymbol{B}}'=\boldsymbol{B}^a-\left(\mathbf{\Phi}^a\right)^T\left[\left(\hat{\mathbf{\Phi}}^u\right)^T\right]^{-1}\hat{\boldsymbol{B}}^u \tag{39}$$

The rank of $\hat{\mathbf{N}}'$ is equal to the rank of $\hat{\mathbf{N}}^u$, which is $m-n-r$.

Eq. (13) can be differentiated to obtain [5,6]

$$\mathbf{\Gamma}\dddot{\boldsymbol{q}}=-\left(2\dot{\mathbf{\Gamma}}\ddot{\boldsymbol{q}}+\ddot{\mathbf{\Gamma}}\dot{\boldsymbol{q}}\right)+\begin{bmatrix}\mathbf{0}_{(m-n)\times 1}\\ \dddot{\boldsymbol{x}}\end{bmatrix} \tag{40}$$

Since $\ddot{\mathbf{\Gamma}}$ also depends on $\ddot{\boldsymbol{q}}$, the terms $-\left(2\dot{\mathbf{\Gamma}}\ddot{\boldsymbol{q}}+\ddot{\mathbf{\Gamma}}\dot{\boldsymbol{q}}\right)$ in Eq. (40) can be rearranged as

$$-\left(2\dot{\mathbf{\Gamma}}\ddot{\boldsymbol{q}}+\ddot{\mathbf{\Gamma}}\dot{\boldsymbol{q}}\right)=\mathbf{P}\ddot{\boldsymbol{q}}+\boldsymbol{R}, \tag{41}$$

where $\mathbf{P}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is an $m\times m$-dimensional matrix and $\boldsymbol{R}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is an m-dimensional vector. By using such a rearrangement in terms of $\mathbf{P}$ and $\boldsymbol{R}$, it is seen that the terms involving $\ddot{\boldsymbol{q}}$ in Eq. (40) are linear in $\ddot{\boldsymbol{q}}$. Solving Eq. (40) for $\dddot{\boldsymbol{q}}$ and substituting Eqs. (15) and (41) yields

$$\dddot{\boldsymbol{q}}=\mathbf{\Gamma}^{-1}\left[\mathbf{P}\mathbf{\Gamma}^{-1}\left(-\dot{\mathbf{\Gamma}}\dot{\boldsymbol{q}}+\begin{bmatrix}\mathbf{0}_{(m-n)\times 1}\\ \ddot{\boldsymbol{x}}\end{bmatrix}\right)+\boldsymbol{R}+\begin{bmatrix}\mathbf{0}_{(m-n)\times 1}\\ \dddot{\boldsymbol{x}}\end{bmatrix}\right] \tag{42}$$

Before proceeding, it is worth recalling that the $\mathbf{\Gamma}$ matrix can be inverted above due to the assumption of no inverse kinematic singularity. Substitution of Eqs. (15) and (42) into Eq. (36) gives

$$\boldsymbol{\tau} = \hat{\mathbf{N}}^* \begin{bmatrix} \mathbf{0}_{(m-n)\times 1} \\ \dddot{\boldsymbol{x}} \end{bmatrix} + \hat{\mathbf{M}}^* \begin{bmatrix} \mathbf{0}_{(m-n)\times 1} \\ \ddot{\boldsymbol{x}} \end{bmatrix} + \hat{\boldsymbol{B}}^*, \tag{43}$$

where the $n \times m$ matrices $\hat{\mathbf{N}}^*$ and $\hat{\mathbf{M}}^*$ are

$$\hat{\mathbf{N}}^* = \hat{\mathbf{N}}' \mathbf{\Gamma}^{-1} \tag{44}$$

$$\hat{\mathbf{M}}^* = \left(\hat{\mathbf{N}}' \mathbf{\Gamma}^{-1} \mathbf{P} + \hat{\mathbf{M}}' \right) \mathbf{\Gamma}^{-1} \tag{45}$$

and the n-dimensional vector $\hat{\boldsymbol{B}}^*$ is

$$\hat{\boldsymbol{B}}^* = \hat{\boldsymbol{B}}' + \hat{\mathbf{N}}' \mathbf{\Gamma}^{-1} \left(\boldsymbol{R} - \mathbf{P} \mathbf{\Gamma}^{-1} \dot{\mathbf{\Gamma}} \dot{\boldsymbol{q}} \right) - \hat{\mathbf{M}}' \mathbf{\Gamma}^{-1} \dot{\mathbf{\Gamma}} \dot{\boldsymbol{q}} \tag{46}$$

In Eq. (43), all of the elements in the first $m-n$ columns of the matrices $\hat{\mathbf{N}}^*$ and $\hat{\mathbf{M}}^*$ are multiplied by zero. This fact leads to a partitioning of the matrices $\hat{\mathbf{N}}^*$ and $\hat{\mathbf{M}}^*$ as

$$\hat{\mathbf{N}}^* = \begin{bmatrix} \hat{\mathbf{N}}^{*G}_{n\times(m-n)} & \hat{\mathbf{N}}^{*P}_{n\times n} \end{bmatrix} \tag{47}$$

$$\hat{\mathbf{M}}^* = \begin{bmatrix} \hat{\mathbf{M}}^{*G}_{n\times(m-n)} & \hat{\mathbf{M}}^{*P}_{n\times n} \end{bmatrix} \tag{48}$$

Based on this partitioning, the relation between the input $\boldsymbol{\tau}$ and the output $\boldsymbol{x}$ can be obtained in the following form:

$$\boldsymbol{\tau} = \hat{\mathbf{N}}^{*P} \dddot{\boldsymbol{x}} + \hat{\mathbf{M}}^{*P} \ddot{\boldsymbol{x}} + \hat{\boldsymbol{B}}^* \tag{49}$$

An inverse dynamics control law can then be formulated using Eq. (49), provided that the desired trajectory is consistent. Since the rank of $\hat{\mathbf{N}}^{*P}$ is less than n, in the control law, both command accelerations $\boldsymbol{u}$ and command jerks $\dot{\boldsymbol{u}}$ have to be specified. $\boldsymbol{u}$ is given in Eq. (21) and $\dot{\boldsymbol{u}}$ is obtained by differentiating Eq. (21) as

$$\dot{\boldsymbol{u}} = \dddot{\boldsymbol{x}}^d + \mathbf{C}_1 \left(\ddot{\boldsymbol{x}}^d - \ddot{\boldsymbol{x}} \right) + \mathbf{C}_2 \left(\dot{\boldsymbol{x}}^d - \dot{\boldsymbol{x}} \right), \tag{50}$$

where $\dddot{\boldsymbol{x}}^d$ is the desired jerk profile of the end-effector. Then $\boldsymbol{\tau}$ can be computed as below:

$$\boldsymbol{\tau} = \hat{\mathbf{N}}^{*P} \dot{\boldsymbol{u}} + \hat{\mathbf{M}}^{*P} \boldsymbol{u} + \hat{\boldsymbol{B}}^* \tag{51}$$

In the absence of modeling errors and disturbances, substitution of Eq. (51) into Eq. (49) results in

$$\hat{\mathbf{N}}^{*P} \dot{\boldsymbol{\eta}} + \hat{\mathbf{M}}^{*P} \boldsymbol{\eta} = \mathbf{0}_{n\times 1}, \tag{52}$$

where $\boldsymbol{\eta} = \boldsymbol{u} - \ddot{\boldsymbol{x}}$. Since Eq. (23) is valid outside the neighborhood of the drive singularity in the absence of modeling errors and disturbances,

$$\boldsymbol{\eta}(t_e) = \mathbf{0}_{n\times 1}, \tag{53}$$

where t_e is the entrance time to the singularity neighborhood. Since the rank of the matrix $\hat{\mathbf{N}}^{*^P}$ is less than n (namely, $m-n-r$), among the n scalar equations of Eq. (52), $n-(m-n-r)=2n-m+r$ of them can be converted to algebraic equations through elementary row operations. Then, using these algebraic equations, $2n-m+r$ of $\eta_1,\cdots,\eta_n$ can be obtained in terms of the remaining ones and these results can be used to reduce the system of Eq. (52) by $2n-m+r$ unknowns and $2n-m+r$ equations. The reduced system is a linear first-order system of $m-n-r$ differential equations in the $m-n-r$ unknowns $\kappa_1(t),\ldots,\kappa_{m-n-r}(t)$, which can be expressed as the initial value problem given below:

$$\dot{\boldsymbol{\kappa}}(t)=\boldsymbol{\Lambda}(t)\,\boldsymbol{\kappa}(t) \tag{54}$$

$$\boldsymbol{\kappa}(t_e)=\mathbf{0}_{(m-n-r)\times 1} \tag{55}$$

Since the $\boldsymbol{\Lambda}(t)$ matrix is a continuous function of t on the closed time interval Υ, and t_e is a given point in Υ, then, according to the existence and uniqueness theorem for linear first-order systems [14], the initial value problem has a unique solution on Υ, and obviously it is the trivial solution, i.e. $\boldsymbol{\kappa}(t)=\mathbf{0}_{(m-n-r)\times 1}$. Hence, in light of the relations between $\boldsymbol{\kappa}$ and $\boldsymbol{\eta}$ described above, $\boldsymbol{\eta}(t)=\mathbf{0}_{n\times 1}$. Therefore, Eqs. (23) and (24) are still valid inside the neighborhood of drive singularities, i.e. the application of the computed forces from Eq. (51) linearizes and decouples the system in the absence of modeling errors and disturbances.

A block diagram of the implementation of the proposed switching inverse dynamics control law around drive singular configurations is shown in Figure 1. The developed method can also be summarized as follows:

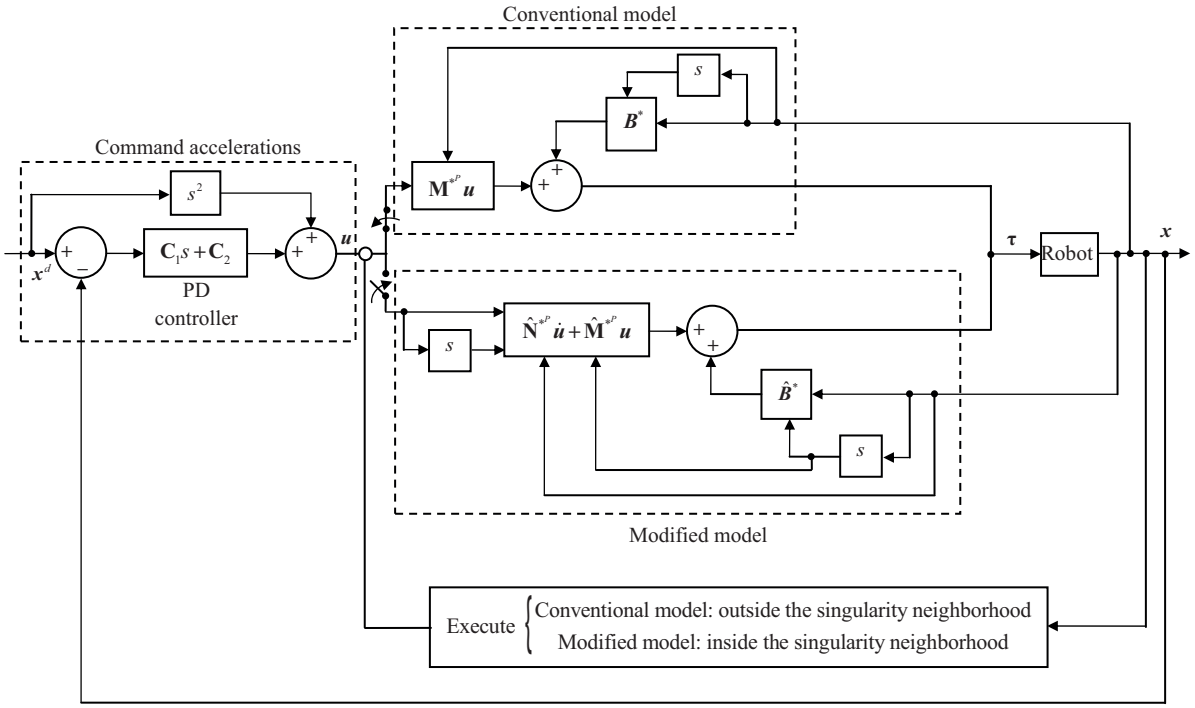

Figure 1. Block diagram of the proposed switching inverse dynamics controller around drive singular configurations.

$$\boldsymbol{\tau}=\begin{cases}\mathbf{M}^{*^P}\boldsymbol{u}+\boldsymbol{B}^*, & \text{outside the neighborhood of singularity}\\ \hat{\mathbf{N}}^{*^P}\dot{\boldsymbol{u}}+\hat{\mathbf{M}}^{*^P}\boldsymbol{u}+\hat{\boldsymbol{B}}^*, & \text{inside the neighborhood of singularity}\end{cases} \tag{56}$$

Before concluding this section, it should be mentioned that, being different from its conventional counterpart, acceleration information is also required in this new control law in addition to the position and velocity measurements. However, as can be seen from Eq. (56), these accelerations are needed only in the neighborhood of the singular position, and can be computed through the following procedure, instead of their direct measurement. Rearranging Eq. (2) and differentiating Eq. (1), one can write

$$\mathbf{M}\ddot{\boldsymbol{q}} - \boldsymbol{\Phi}^T\boldsymbol{\lambda} = \begin{bmatrix} \boldsymbol{\tau} \\ \mathbf{0}_{(m-n)\times 1} \end{bmatrix} - \boldsymbol{B} \tag{57}$$

$$\boldsymbol{\Phi}\ddot{\boldsymbol{q}} = -\dot{\boldsymbol{\Phi}}\dot{\boldsymbol{q}} \tag{58}$$

In the above equations, $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$ are the measured quantities, and $\boldsymbol{\tau}$ is vector of the computed torques and forces at one previous time step. Hence, Eqs. (57) and (58) constitute a linear system of $2m-n$ scalar equations in $2m-n$ unknowns, $\ddot{\boldsymbol{q}}$ and $\boldsymbol{\lambda}$. As mentioned previously, under the assumption of no inverse kinematic singularity, $\ddot{\boldsymbol{q}}$ can be uniquely determined from this system [11,12]. Once $\ddot{\boldsymbol{q}}$ is solved, $\ddot{\boldsymbol{x}}$ can be determined using Eq. (13). It should be noted that the accuracy of the computed accelerations as described here will depend on the accuracy of the dynamic model at hand. A similar computational approach was also used in Ref. [15] for the accelerations required in the inverse dynamics control of flexible joint parallel manipulators.

4. Case study

A 3 degree of freedom RPR-RPR planar parallel manipulator is considered in this section, as shown in Figure 2. Here R denotes a revolute joint, P a prismatic joint, and underlines denote the actuated joints. This manipulator was previously considered in Ref. [6], where its inverse dynamics, singular configurations, "consistency conditions", and "modified equations" were given. The numerical values of the kinematic and dynamic parameters of the manipulator, and the singularity-consistent motion used in the numerical example of Ref. [6] are also used in the simulations of the present study as the corresponding numerical values of the manipulator parameters given in Table 1, and as the reference (desired) motion given below for $0 \le t \le 1$ s, respectively:

$$x_P^d(t) = 0.800 + f(t)\cos\left(\frac{10}{9}\pi\right) \tag{59}$$

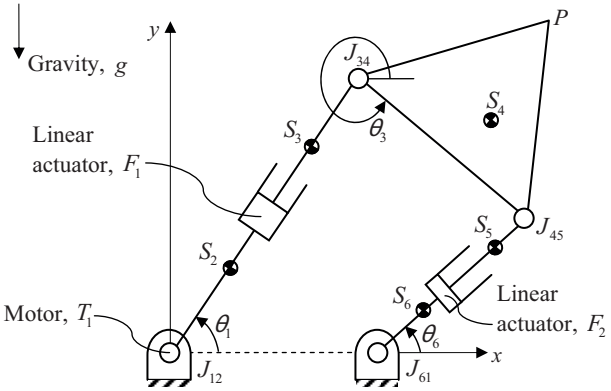

Figure 2. A 3 degree of freedom RPR-RPR planar parallel manipulator.

Table 1. Numerical values of the manipulator parameters (taken from Ref. [6]).

Link lengths and angular dimensions			Masses (kg)		Centroidal moments of inertia (kg.m^2)		Mass center distances (m)	
(m)	$\lvert J_{12}J_{61}\rvert$	1.0	m_2	2.0	I_2	0.05	$\lvert J_{12}S_2\rvert$	0.15
	$\lvert J_{34}J_{45}\rvert$	0.4	m_3	1.5	I_3	0.03	$\lvert J_{34}S_3\rvert$	0.15
	$\lvert J_{34}P\rvert$	0.2	m_4	1.0	I_4	0.02	$\lvert J_{34}S_4\rvert$	0.20
(°)	$\angle PJ_{34}J_{45}$	0	m_5	1.5	I_5	0.03	$\lvert J_{45}S_5\rvert$	0.15
	$\angle S_4J_{34}J_{45}$	0	m_6	2.0	I_6	0.05	$\lvert J_{61}S_6\rvert$	0.15

$$y_P^d(t) = 0.916 + f(t)\sin\left(\frac{10}{9}\pi\right) \tag{60}$$

$$\theta_3^d(t) = 320^\circ, \tag{61}$$

where

$$f(t) = 20.7303t^2 - 87.8009t^3 + 146.5637t^4 - 103.6459t^5 + 25.6528t^6 \tag{62}$$

The parallel manipulator is assumed to be initially mispositioned with $(x_P)_0 = 0.75$ m, $(y_P)_0 = 0.900$ m and $(\theta_3)_0 = 318^\circ$. The control torque and forces are to be computed using Eq. (22) unless the manipulator is near a drive singularity. However, a drive singularity arises when $\theta_6 = \theta_3 - \pi$ [6], and Eq. (51) is to be switched within the $\pm$ δ-neighborhood of this singular configuration (i.e. for $|\theta_3 - \theta_6 - \pi| < \delta$ [6]), where δ is a positive constant used to define the switching condition between the two control modes.

To validate the performance of the proposed control law, a series of simulations is performed in the MATLAB®/Simulink® environment, and the results are summarized in Table 2 [16]. In the first of these simulations (i.e. Case 1 of Table 2), a PD controller is used to generate the command accelerations, and δ is selected to be 0.5°. The constant feedback gain diagonal matrices are chosen to be in binomial form, i.e. $\mathbf{C}_1 = 2\omega_0\mathbf{I}_3$ and $\mathbf{C}_2 = \omega_0^2\mathbf{I}_3$, where $\mathbf{I}_n$ denotes the $n \times n$ identity matrix and ω_0 is a positive constant chosen as 30 rad/s. The errors in the end-effector states, the motor torque, and the actuator forces are shown in Figures 3–6. Negligibly small steady-state errors in the end-effector states are obtained both before the entrance of the neighborhood and towards the end of the task. The maximum task violations occurred after entering the singularity neighborhood are 0.0002 mm in x_P and y_P, and 0.0001° in θ_3. In addition, the jumps in the control torque and forces at the entrance and exit of the neighborhood are negligible, and hence not visible in the associated figures. Note that, at the end of the considered motion, the desired end-effector velocities are zero, but the desired acceleration of point P is not zero. For this reason, the actuator efforts are not close to zero at the end of the shown simulation.

Moreover, the tracking performance is observed to be quite good even in a doubled switch region (i.e. $\delta = 1^\circ$) such that the steady-state errors in the end-effector states both before the entrance of the neighborhood and towards the end of the task are again negligible, and the maximum task violations that occurred after entering the singularity neighborhood are still pretty small: 0.0026 mm and 0.0031 mm in x_P and in y_P, respectively, and 0.0013° in θ_3 (Case 2 of Table 2).

To see the effects of modeling errors, the closed-loop control system is also simulated using 5% smaller values for the mass and inertia values in the model while all the other simulation parameters are kept the same as in the first simulation. This corresponds to Case 3 of Table 2. The errors in the end-effector states towards

Table 2. Summary of the simulation results.

Case	Mode error	Action	ω_0 (rad/s)	δ (°)	Steady-state errors (as absolute values)						Maximum errors (as absolute values)					
					before the entrance of the singularity neighborhood			towards the end of the task			in the singularity neighborhood			after the singularity neighborhood		
					x_P (mm)	y_P (mm)	θ_3 (°)	x_P (mm)	y_P (mm)	θ_3 (°)	x_P (mm)	y_P (mm)	θ_3 (°)	x_P (mm)	y_P (mm)	θ_3 (°)
1	No	PD	30	0.5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001	0.0000	0.0002	0.0002	0.0001
2	No	PD	30	1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0016	0.0019	0.0008	0.0026	0.0031	0.0013
3	5%	PD	30	0.5	0.4538	0.4414	0.0157	1.1108	1.0543	0.0039	0.4734	0.4408	0.0157	1.1108	1.0543	0.0152
4	5%	PD	50	0.5	0.1862	0.1519	0.0055	0.4820	0.4086	0.0015	0.1912	0.1518	0.0055	0.4820	0.4086	0.0051
5	5%	PID	30	0.5	0.0928	0.0301	0.0004	0.2755	0.0972	0.0004	0.0921	0.0298	0.0006	0.2755	0.0972	0.0033
6	5%	PID	50	0.5	0.0139	0.0039	0.0002	0.0626	0.0223	0.0001	0.0137	0.0038	0.0003	0.0626	0.0223	0.0008

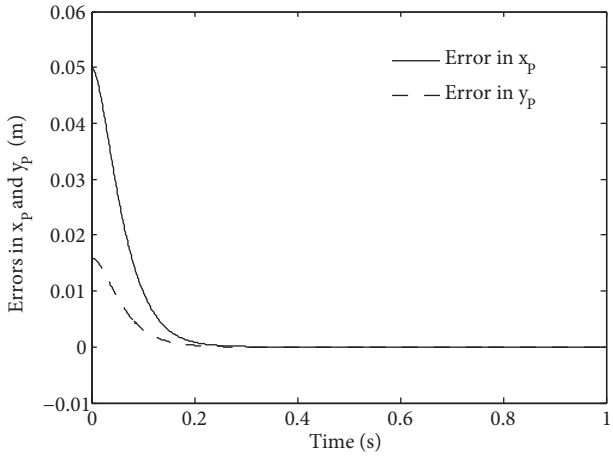

Figure 3. Errors x_P and y_P in Case 1, where no modeling error exists, command accelerations are generated through a PD controller with an ω_0 of 30 rad/s, and switching is set to be in the $\pm$ 0.5° -neighborhood of the drive singularity.

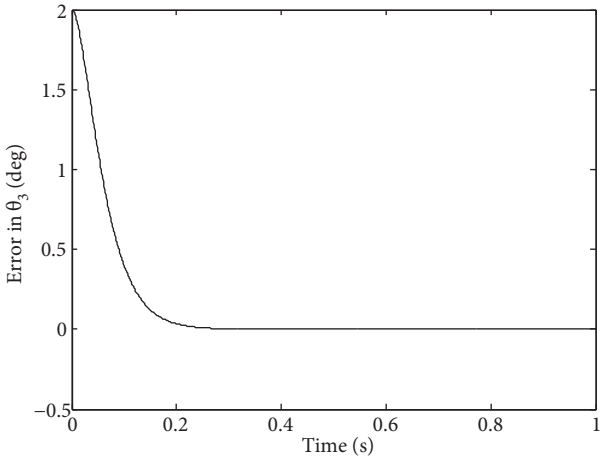

Figure 4. Error in θ_3 in Case 1, where no modeling error exists, command accelerations are generated through a PD controller with an ω_0 of 30 rad/s, and switching is set to be in the $\pm$ 0.5° -neighborhood of the drive singularity.

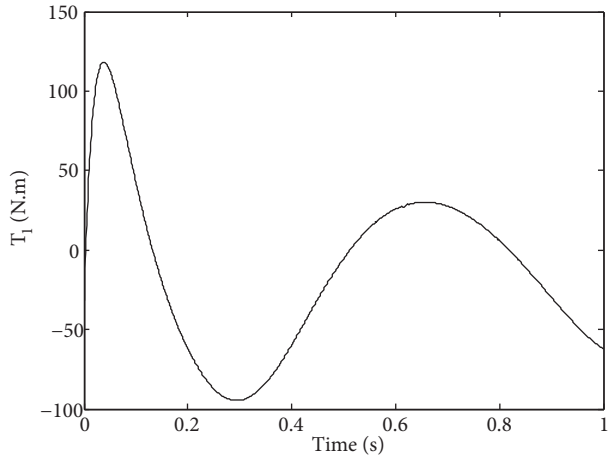

Figure 5. Motor torque, T_1, in Case 1, where no modeling error exists, command accelerations are generated through a PD controller with an ω_0 of 30 rad/s, and switching is set to be in the $\pm$ 0.5° -neighborhood of the drive singularity.

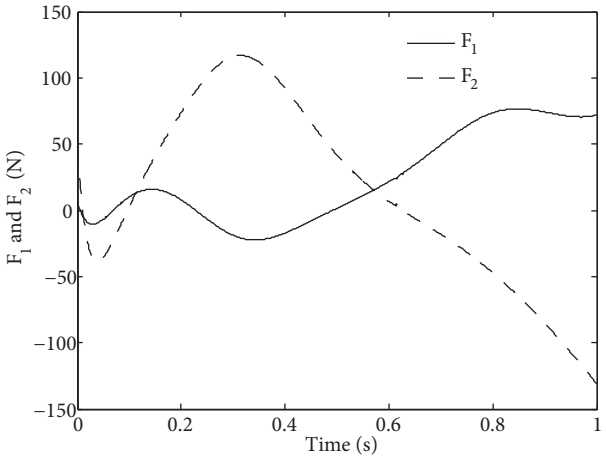

Figure 6. Linear actuator forces, F_1 and F_2, in Case 1, where no modeling error exists, command accelerations are generated through a PD controller with an ω_0 of 30 rad/s, and switching is set to be in the $\pm$ 0.5° -neighborhood of the drive singularity.

the entrance of the neighborhood and towards the end of the task take almost constant values so that steady-state can be assumed to be reached (Figures 7 and 8). Furthermore, the steady-state errors at the entrance of the neighborhood are quite small (0.4538 mm and 0.4414 mm in x_P and y_P, respectively, and 0.0157° in θ_3). The steady-state errors towards the end of the task are also small: 1.1108 mm and 1.0543 mm in x_P and y_P, respectively, and 0.0039° in θ_3. The jumps in the control torque and forces at the entrance and exit of the neighborhood are small enough not to cause large position errors (Figures 9 and 10). The tracking performance in the presence of modeling errors can be improved with increasing ω_0. As an example of this, with ω_0 increased from 30 rad/s to 50 rad/s only, 57% to 66% reductions are achieved in all the steady-state errors and in all the maximum task violations occurred after entering the singularity neighborhood, as can be seen in Case 4 of

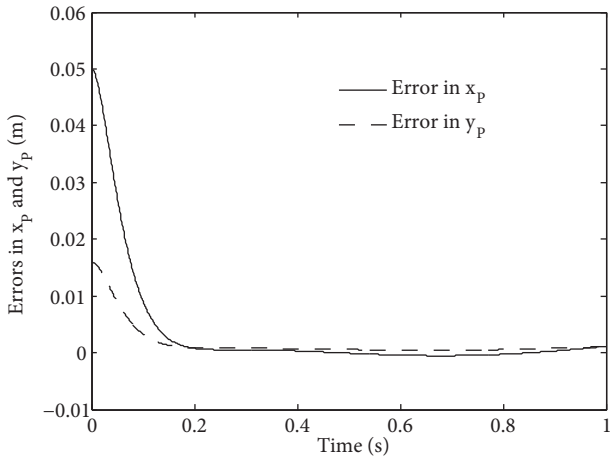

Figure 7. Errors in x_P and y_P in Case 3, i.e. in the presence of 5% modeling error while the rest is the same as in Case 1.

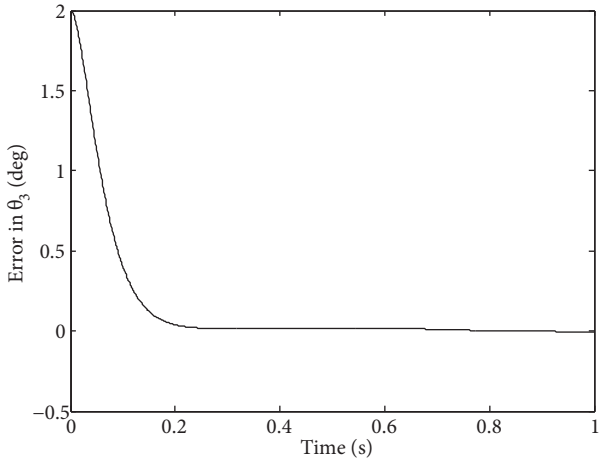

Figure 8. Error in θ_3 in Case 3, i.e. in the presence of 5% modeling error while the rest is the same as in Case 1.

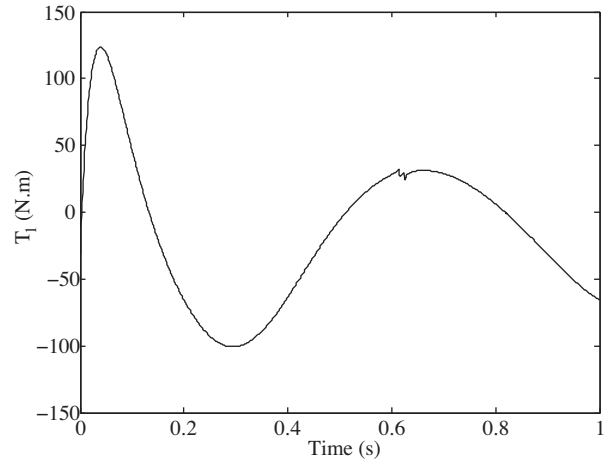

Figure 9. Motor torque, T_1, in Case 3, i.e. in the presence of 5% modeling error while the rest is the same as in Case 1.

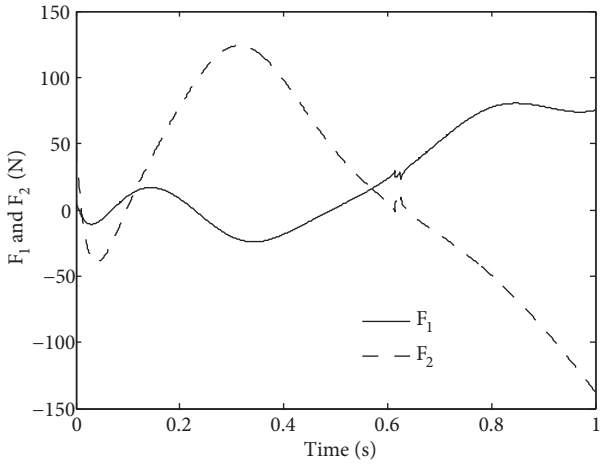

Figure 10. Linear actuator forces, F_1 and F_2, in Case 3, i.e. in the presence of 5% modeling error while the rest is the same as in Case 1.

Table 2: all the errors in x_P and y_P are in the order of one-tenth of a millimeter, and all the errors in θ_3 are in the order of one-thousandth of a degree. Another way of decreasing the resulting errors in the presence of modeling errors is to add an integral action to the controller, i.e. the command accelerations are generated as

$$\boldsymbol{u} = \ddot{\boldsymbol{x}}^d + \mathbf{K}_1 \left(\dot{\boldsymbol{x}}^d - \dot{\boldsymbol{x}} \right) + \mathbf{K}_2 \left(\boldsymbol{x}^d - \boldsymbol{x} \right) + \mathbf{K}_3 \int \left(\boldsymbol{x}^d - \boldsymbol{x} \right) dt, \tag{63}$$

where the constant feedback gain diagonal matrices can be again chosen to be in binomial form, i.e. $\mathbf{K}_1 = 3\omega_0 \mathbf{I}_3$, $\mathbf{K}_2 = 3\omega_0^2 \mathbf{I}_3$, and $\mathbf{K}_3 = \omega_0^3 \mathbf{I}_3$. Then, with the same ω_0 of 30 rad/s, the addition of the integral control yields 75% to 97% reductions in all the steady-state errors and in all the maximum task violations encountered after entering the singularity neighborhood, as shown in Case 5 of Table 2: all the errors in x_P and y_P are at most in the order of one-tenth of a millimeter, and all the errors in θ_3 are at most in the order of one-thousandth of a degree. Moreover, further improvements can be achieved by increasing ω_0, while using a PID controller. When

compared to a PD controller with an ω_0 of 30 rad/s, 94% to 99% reductions are accomplished in all the steady-state errors and in all the maximum task violations encountered after entering the singularity neighborhood by using a PID controller with an ω_0 of 50 rad/s, as given in Case 6 of Table 2: all the errors in x_P and y_P are at most in the order of one-hundredth of a millimeter, and all the errors in θ_3 are in the order of one ten-thousandth of a degree.

5. Discussion and conclusions

Due to the highly nonlinear and coupled nature of parallel manipulators, control strategies that are not based on a dynamic context cannot guarantee high performance in every application [17,18] and can even become unstable in high speed operations [17]. Therefore, owing to its linearizing and decoupling feedback loop, inverse dynamics control is widely used for this aim; and it is also extended to handle the control of parallel manipulators with their joint flexibilities also being taken into account, some examples of which were cited in Ref. [19]. However, this conventional technique suffers from being ill-conditioned around drive singular configurations, and hence the actuators will unavoidably saturate in their neighborhood. This will dramatically deteriorate the performance of the manipulator by resulting in poor tracking. Thus, the use of the conventional inverse dynamics controller is severely restricted in practice, due to these singularities [20], and the usable workspace of parallel manipulators is dramatically decreased.

In this paper, a switching inverse dynamics controller is proposed for the trajectory tracking control of parallel manipulators considering their drive singularities. For this purpose, while the conventional inverse dynamics controller is employed outside the neighborhood of singularities, a new inverse dynamics control law is developed to be switched around them. It should be recalled here that the desired trajectory is to be such that the dynamic equations of motion of the manipulator are consistent at the singular configuration.

The performance of the proposed controller is tested by applying it to a 3 degree of freedom RPR-RPR planar parallel manipulator in the presence of initial positioning errors and modeling errors. In all the cases considered, including the cases wherein the selected size of the singularity neighborhood is doubled or modeling errors exist, good tracking performance is obtained, while the motor torque and actuator forces are kept finite and continuous at and near the singularity.

Similar to its conventional counterpart, the performance of this newly suggested control law also depends on the accuracy of the dynamic model of the manipulator. However, a precise tracking of the desired trajectory is still possible even in the presence of modeling errors and disturbances by increasing the feedback gains and/or adding an integral action to the controller. Another alternative for handling the model uncertainties might be to combine the inverse dynamics control law with the learning algorithms [21].

It should also be emphasized that, although acceleration information is needed for determining the control forces in the neighborhood of the singularity, the necessity of their measurement is eliminated by the use of the dynamic equations. Hence, position and velocity measurements of the manipulator joints are sufficient to realize the proposed switching control law.

Acknowledgments

The first author (Mustafa Özdemir) is thankful to the Scientific and Technological Research Council of Turkey (TÜBİTAK) for the support provided through the National Scholarship Program for MS Students. This article is based on his MS thesis under the supervision of the second author (Sıtkı Kemal İDER).

Nomenclature

Symbols

$\boldsymbol{B}$	vector that includes the generalized Coriolis, centrifugal, and gravity forces
$\mathbf{C}_1, \mathbf{C}_2$	constant feedback gain diagonal matrices
$\boldsymbol{e}$	vector that includes the errors in the end-effector states
m	degree of freedom of the open-tree structure
$\mathbf{M}$	generalized inertia matrix
n	degree of freedom of the parallel robot
$\hat{\mathbf{N}}^u$	coefficient matrix of $\dddot{\boldsymbol{q}}$ in the modified dynamic equations
$\boldsymbol{q}$	joint variable vector
$\mathbf{P}$	coefficient matrix of $\ddot{\boldsymbol{q}}$ in the combined jerk-level kinematic relations
$\boldsymbol{R}$	vector of the third order velocity terms appearing in the combined jerk-level kinematic relations
t	time
$\boldsymbol{u}$	command accelerations
$\boldsymbol{x}$	vector of the Cartesian end-effector variables
$\boldsymbol{\Gamma}$	coefficient matrix of $\ddot{\boldsymbol{q}}$ in the combined acceleration-level kinematic relations
$\boldsymbol{\lambda}$	vector of Lagrange multipliers
$\boldsymbol{\tau}$	vector of the actuator forces
$\boldsymbol{\Phi}$	coefficient matrix of $\dot{\boldsymbol{q}}$ in the velocity-level loop-closure constraint equations
$\boldsymbol{\psi}$	coefficient matrix of $\dot{\boldsymbol{q}}$ in the velocity-level task equations

Superscripts and modifiers

a	partitions associated with the actuated joints
$*$	modifications associated with the coordinate transformation by the direct kinematic solution to express the equations of motion in terms of the vector $\begin{bmatrix} \mathbf{0}_{1\times(m-n)} & \ddot{\boldsymbol{x}} \end{bmatrix}^T$ (and also its time derivative in the modified dynamic equations) after $\boldsymbol{\lambda}$ is eliminated
d	desired values
$\cdot$	time derivative
G	partitions associated with the zero elements of the vector $\begin{bmatrix} \mathbf{0}_{1\times(m-n)} & \ddot{\boldsymbol{x}} \end{bmatrix}^T$ (and also its time derivative in the modified dynamic equations)
$\widehat{\ }$	modifications associated with replacing the linearly dependent dynamic equations by the "modified equations"
-1	inverse
P	partitions associated with the Cartesian end-effector variables
$\prime$	modifications associated with eliminating $\boldsymbol{\lambda}$ from the equations of motion
T	transpose
u	partitions associated with the unactuated joints

References

[1] Gosselin C, Angeles J. Singularity analysis of closed-loop kinematic chains. IEEE T Robotic Autom 1990; 6: 281-290.

[2] Choudhury P, Ghosal A. Singularity and controllability analysis of parallel manipulators and closed-loop mechanisms. Mech Mach Theory 2000; 35: 1455-1479.

[3] Özgören MK. Motion control of constrained systems considering their actuation-related singular configurations. P I Mech Eng I-J Sys 2001; 215: 113-123.

[4] Jui CKK, Sun Q. Path tracking of parallel manipulators in the presence of force singularity. J Dyn Syst-T ASME 2005; 127: 550-563.

[5] Ider SK. Inverse dynamics of parallel manipulators in the presence of drive singularities. Mech Mach Theory 2005; 40: 33-44.

[6] Ider SK. Singularity robust inverse dynamics of planar 2-RPR parallel manipulators. P I Mech Eng C-J Mec 2004; 218: 721-730.

[7] Briot S, Arakelian V. Optimal force generation in parallel manipulators for passing through the singular positions. Int J Robot Res 2008; 27: 967-983.

[8] Briot S, Arakelian V. On the dynamic properties of rigid-link flexible-joint parallel manipulators in the presence of type 2 singularities. J Mech Robot 2010; 2: 021004-1-021004-6.

[9] Briot S, Arakelian V. On the dynamic properties of flexible parallel manipulators in the presence of type 2 singularities. J Mech Robot 2011; 3: 031009-1-031009-8.

[10] Ghorbel F, Srinivasan B, Spong MW. On the positive definiteness and uniform boundedness of the inertia matrix of robot manipulators. In: 32nd IEEE Conference on Decision and Control; 15–17 December 1993; San Antonio, TX, USA: IEEE. pp. 1103-1108.

[11] Udwadia FE, Schutte AD. Equations of motion for general constrained systems in Lagrangian mechanics. Acta Mech 2010; 213: 111-129.

[12] De Jalón JG, Gutiérrez-López MD. Multibody dynamics with redundant constraints and singular mass matrix: existence, uniqueness, and determination of solutions for accelerations and constraint forces. Multibody Syst Dyn 2013; 30: 311-341.

[13] Lewis FL, Dawson DM, Abdallah CT. Robot Manipulator Control: Theory and Practice. 2nd ed., Revised and Expanded. Boca Raton, FL, USA: CRC Press, 2006.

[14] Greenberg MD. Advanced Engineering Mathematics. 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 1998.

[15] Ider SK, Korkmaz O. Trajectory tracking control of parallel robots in the presence of joint drive flexibility. J Sound Vib 2009; 319: 77-90.

[16] Özdemir M. Inverse dynamics control of parallel manipulators around singular configurations. MS, Middle East Technical University, Ankara, Turkey, 2008.

[17] Shang W, Cong S. Nonlinear computed torque control for a high-speed planar parallel manipulator. Mechatronics 2009; 19: 987-992.

[18] Zubizarreta A, Marcos M, Cabanes I, Pinto C. A procedure to evaluate Extended Computed Torque Control configurations in the Stewart-Gough platform. Robot Auton Syst 2011; 59: 770-781.

[19] Kılıçaslan S. Tracking control of elastic joint parallel robots via state-dependent Riccati equation. Turk J Electr Eng & Comp Sci 2015; 23: 522-538.

[20] Omran A, Elshabasy M. A note on the inverse dynamic control of parallel manipulators. P I Mech Eng C-J Mec 2010; 224: 25-32.

[21] Wang J, Wu J, Wang L, You Z. Dynamic feed-forward control of a parallel kinematic machine. Mechatronics 2009; 19: 313-324.