

Crossing Type 2 Singularities of Parallel Robots without Pre-planned Trajectory with a Virtual-constraint-based Controller

Rafael Balderas Hill^{1,2}, Damien Six^{1,2}, Abdelhamid Chriette^{1,2}, Sébastien Briot² and Philippe Martinet^{1,2}

Abstract—The presence of Type 2 singularities in parallel robots severely affects their performances, mainly because the platform motion control is partially lost. It also leads to a size reduction of the operational workspace. Moreover, the dynamic model of the parallel mechanism degenerates and locally, the robot becomes underactuated in the singularity. It has been proven that it is possible to cross Type 2 singularities by respecting a dynamic criterion. Nevertheless, the controllers designed up to now require a pre-planned optimized trajectory including this criterion, and as a result, this strategy can only be used by qualified users. In order to avoid this drawback and to cross these types of singularities even if the trajectory is not pre-planned, this paper proposes a controller based on virtual constraints. Furthermore, the controller is integrated in a multi-control architecture in order to switch between a classical computed torque control far from the singularity and the virtual-constraint-based control law near to the singularity locus. Experimental results on a five-bar mechanism validated the automatic Type 2 singularity crossing.

I. INTRODUCTION

Parallel robots have several advantages compared to serial robots, such as higher stiffness, higher payload, lowest inertia and higher speed and acceleration. However, due to their geometric and kinematic complexity, singularity issues appear, which lead to the division of their workspace into several aspects corresponding to different assembly modes [1][2]. For a general overview on the classification of the singularities in parallel robots the reader can refer to [3][4]. The literature review shows that some designers have overcome the problems of workspace reachability by designing mechanisms without singularities [5], but this usually leads to robots with small workspace or parallel robots with low practicability. One can also use redundancy [6] but the use of additional actuators is costly. Another approach was presented in [7] in which it has been shown that it is possible to pass through Type 2 singularities by planning a trajectory respecting a physical criterion defined from the analysis of the degeneracy conditions of the dynamic model. This last approach is promising since it has shown its efficiency to cross the Type 2 singularities and to increase the reachable workspace thanks to a modification of the robot trajectory. However, the trajectory planning approach should be set more robust by combining a dedicated controller with it. Thus, a recent solution [8] proposed the implementation of a Multi-Model Controller in order to track the trajectory that

respects the physical criterion, and by switching between two different dynamic models. Nevertheless, the main issue with the current approach is that the definition of the optimal trajectory for crossing is required to be done by a well-skilled user, having some advanced knowledge in parallel robot dynamics. This is a brake towards automatizing the singularity crossing. The aim of this paper is to propose an advanced feedback control law dedicated to ensure un-planned singularity crossing allowing for an unskilled user to design a trajectory without specific knowledge in parallel robot dynamics. In order to solve this problem, we need to find a strategy which is able to control the underactuation of the end-effector of the parallel robot which occurs locally in Type 2 singularity [3].

In control of underactuated systems, several approaches exist [9], however, one of the most powerful tools is the enforcement of virtual constraints through feedback control. Virtual constraints are geometrical relations defined between controlled and uncontrolled variables that can be enforced through control action. Virtual constraints strategy has been applied to several applications, e.g. humanoid walking in [10], inverted pendulums in [11], aerial systems in [12]. Moreover, rigorous mathematical formulations have been presented in [11][13]. Thus, the use of virtual constraints for solving the problem of Type 2 singularity crossing seems to be a promising solution for dealing with the underactuated behavior of the robot at singularity given that at this specific singular posture, controlled and uncontrolled variables can be identified.

Thus, in this paper, we develop a controller based on virtual constraint approach in order to cross the singularity without pre-planned trajectory. Our main contributions are listed below:

- Separation of the controlled and free dynamics, locally, in the singularity locus;
- Feedback control law based on virtual constraints that respects the dynamic criterion/free dynamics, even if the trajectory is not pre-planned, avoiding the degeneracy of the dynamic model;
- Multi-control architecture in order to continuously perform a trajectory tracking far from the singularity and in the neighborhood of the singularity.

II. PHYSICAL BACKGROUND

A. Dynamic modeling of parallel robots

Let us briefly recall the dynamic modeling of parallel robots presented in [14]. We will assume a parallel robot

¹ Ecole Centrale de Nantes, 44321 Nantes, France

² Laboratoire des Sciences du Numérique de Nantes (LS2N), UMR 6004, 44321 Nantes, France {Rafael.Balderas-Hill, Damien.Six, Abdelhamid.Chriette, Sebastien.Briot, Philippe.Martinet}@ls2n.fr

composed of n degrees of freedom. The actuation is given by n active joints. The position and velocity of the parallel robot can be described by using:

- \mathbf{q} and $\dot{\mathbf{q}}$ representing the n -dimensional vectors of active joint variables and of active joint velocities, respectively.
- \mathbf{x} and $\dot{\mathbf{x}}$ representing the n -dimensional vectors of platform pose and of its time derivatives, respectively.

The input-output kinematic constraint that relates $\dot{\mathbf{x}}$ with the active joint velocities $\dot{\mathbf{q}}$ is:

$$\mathbf{A}\dot{\mathbf{x}} + \mathbf{B}\dot{\mathbf{q}} = \mathbf{0} \quad (1)$$

where \mathbf{A} and \mathbf{B} are the $(n \times n)$ parallel and serial kinematic Jacobian matrices, respectively [1]. Then, by using the Lagrange formalism, the dynamic model of the robot can be written as follows:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{ta} - \mathbf{B}^T \boldsymbol{\lambda} \quad (2)$$

$$\mathbf{w}_p = \mathbf{A}^T \boldsymbol{\lambda} \quad (3)$$

where:

- $\boldsymbol{\tau}$ is the n -dimensional vector of the robot input efforts;
- $\boldsymbol{\lambda}$ is the n -dimensional vector of Lagrange multipliers;
- $\boldsymbol{\tau}_{ta}$ and \mathbf{w}_p are the n -dimensional vectors related to the Lagrangian L of the system:

$$\boldsymbol{\tau}_{ta} = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{q}}} \right)^T - \left(\frac{\partial L}{\partial \mathbf{q}} \right)^T \quad (4)$$

$$\mathbf{w}_p = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{x}}} \right)^T - \left(\frac{\partial L}{\partial \mathbf{x}} \right)^T \quad (5)$$

B. General solution for the inverse dynamic model and degeneracy conditions of the dynamic model

This section seeks to give the general solution for the inverse dynamic model and to recall its degeneracy conditions due to the loss of rank of the kinematic Jacobian matrix \mathbf{A} . Based on the classification of the singularities in [3], the parallel robots can encounter three singular configurations: **Type 1 singularities** occur when the mechanism loses its ability to move along one given direction. In such singular configurations the kinematic matrix \mathbf{B} becomes rank deficient.

Type 2 singularities occur when the mechanism gains some uncontrollable motions. In such singularities the kinematic matrix \mathbf{A} becomes rank deficient.

Type 3 singularities are configurations when matrices \mathbf{A} and \mathbf{B} are simultaneously rank deficient.

The general solution for the inverse dynamic model of a parallel robot assuming the matrix \mathbf{A} can be inverted, meaning that the lagrange multipliers $\boldsymbol{\lambda}$ from (3) can be substituted into (2), is given by:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{ta} - \mathbf{B}^T \mathbf{A}^{-T} \mathbf{w}_p \quad (6)$$

The solution of the dynamic model under the form (6) assumes that the kinematic matrix \mathbf{A} is a full rank matrix. Furthermore, for a prescribed trajectory in the Cartesian space, the values of the vectors $\ddot{\mathbf{q}}$, $\dot{\mathbf{q}}$ and \mathbf{q} can be calculated

from the inverse kinematics. Thus, considering that the parallel robot is not in a Type 1 singularity, the terms $\boldsymbol{\tau}_{ta}$ and \mathbf{w}_p can be computed. Nevertheless, for a trajectory passing through a Type 2 singularity, the matrix \mathbf{A} is rank deficient [3]. In such singular configurations, the parallel mechanism gains one or more uncontrollable degrees of freedom making the system locally underactuated. An example of a Type 2 singularity is shown in Fig. 1 on a five-bar mechanism.

The degeneracy studied in the present research work will be limited to one uncontrollable degree of freedom at the singularity locus. The case of parallel robots with more degrees of degeneracy rarely appears.

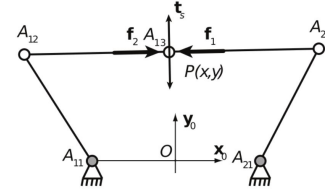


Fig. 1. A five-bar mechanism gains an uncontrollable motion perpendicular to $\vec{A}_{12}A_{13}$ and $\vec{A}_{22}A_{13}$. The vector \mathbf{t}_s represents the direction of the uncontrollable motion, which makes the robot underactuated. \mathbf{f}_1 and \mathbf{f}_2 are two forces applied by the actuators on the platform through the passive legs.

III. MODELING OF THE FREE AND CONTROLLED DYNAMICS

A. Separation of the free and controlled dynamics in the singularity locus

Locally in the singularity locus it is possible to separate the controlled and free dynamics. In order to do that let us express the term \mathbf{w}_p in (3), (5) under the following form given in [14]:

$$\mathbf{w}_p = \mathbf{M}_r \ddot{\mathbf{x}} + \mathbf{c}_r = \mathbf{A}^T \boldsymbol{\lambda} \quad (7)$$

where \mathbf{M}_r is an $(n \times n)$ definite positive matrix of inertia depending on the platform coordinates [14], and \mathbf{c}_r is an n -dimensional vector of Coriolis, centrifugal and gravitational effects and its value depends on the platform coordinates \mathbf{x} and their time derivatives $\dot{\mathbf{x}}$. Knowing that the matrix \mathbf{A} is rank deficient in Type 2 singularity, there exists a unit vector \mathbf{t}_s in its kernel, such that:

$$\mathbf{A} \mathbf{t}_s = \mathbf{0} \quad (8)$$

We take advantage of this property in order to build an orthonormal matrix \mathbf{V} described as follows:

$$\mathbf{V} = \begin{bmatrix} \mathbf{t}_s^T \\ \text{-----} \\ \mathbf{V}_c \end{bmatrix} \quad (9)$$

whose first row is the transpose of \mathbf{t}_s and \mathbf{V}_c is an $((n-1) \times n)$ matrix grouping the other rows of \mathbf{V} in order to fill the orthonormal basis. Then, multiplying (7) by the orthonormal matrix \mathbf{V} , and decomposing the resulting equations, we obtain:

$$\mathbf{t}_s^T \mathbf{w}_p = \mathbf{t}_s^T (\mathbf{M}_r \ddot{\mathbf{x}} + \mathbf{c}_r) = 0 \quad (10)$$

$$\mathbf{V}_c \mathbf{w}_p = \mathbf{A}_s^T \boldsymbol{\lambda} \quad (11)$$

where \mathbf{A}_s is an $((n-1) \times n)$ reduced kinematic Jacobian matrix, as follows:

$$\mathbf{A}_s^T = \mathbf{V}_c \mathbf{A}^T \quad (12)$$

Expression (10) represents the local free dynamics of the robot in singularity, and equation (11) represents the controlled dynamics, given that the control inputs appear by substituting $\boldsymbol{\lambda}$, computed from expression (2), into (11):

$$\mathbf{V}_c \mathbf{w}_p = \mathbf{A}_s^T (\mathbf{B}^{-T} (\boldsymbol{\tau} - \boldsymbol{\tau}_{ta})) \quad (13)$$

Expression (13) is valid since matrix \mathbf{B} is not singular in a Type 2 singularity as disclosed in the singularity conditions of Section II-B.

B. Controlled and uncontrolled variables in the singularity locus

Based on the separation of the free and controlled dynamics, let us identify the controlled and uncontrolled variables in the singularity locus by performing a change of variable as follows:

$$\boldsymbol{\xi} = \mathbf{V} \mathbf{M}_r \mathbf{x} \quad (14)$$

where $\mathbf{V} \mathbf{M}_r$ is an invertible matrix considered locally constant since the underactuated behavior occurs restrictively only in the singularity locus. Then, the platform dynamics can be expressed with the new variable $\boldsymbol{\xi}$ by computing the following expression:

$$\ddot{\boldsymbol{\xi}} = \mathbf{V} \mathbf{M}_r \ddot{\mathbf{x}} \quad (15)$$

where $\ddot{\boldsymbol{\xi}}$ is the acceleration of the variable change. By introducing the change of variable (14) into equations (10) and (11), the free and controlled dynamics in the coordinate $\boldsymbol{\xi}$ become:

$$\ddot{\xi}_1 + \mathbf{t}_s^T \mathbf{c}_r = 0 \quad (16)$$

$$\ddot{\boldsymbol{\xi}}_c + \mathbf{V}_c \mathbf{c}_r = \mathbf{A}_s^T \boldsymbol{\lambda} \quad (17)$$

where ξ_1 is the first term of $\boldsymbol{\xi}$ and is the uncontrolled variable and $\boldsymbol{\xi}_c = [\xi_2, \xi_3, \dots, \xi_n]^T$ is the $(n-1)$ -dimensional vector of controlled variables, grouping all the remaining components of $\boldsymbol{\xi}$.

IV. DYNAMIC MODEL IN THE SINGULARITY LOCUS

The inverse dynamic model can be expressed as a function of the acceleration of the controlled variables only. According to [14], equation (4) can be obtained under the form:

$$\boldsymbol{\tau}_{ta} = \mathbf{M}_a \ddot{\mathbf{q}} + \mathbf{c}_a \quad (18)$$

with \mathbf{M}_a as an $(n \times n)$ matrix and \mathbf{c}_a as an n -dimensional vector. Let us now substitute (18) into (2) and compute \mathbf{w}_p in the coordinate $\boldsymbol{\xi}$, then the dynamic model can be expressed:

$$\boldsymbol{\tau} = \mathbf{M}_a \ddot{\mathbf{q}} + \mathbf{c}_a - \mathbf{B}^T \boldsymbol{\lambda} \quad (19)$$

$$\mathbf{w}_p = \ddot{\boldsymbol{\xi}} + \mathbf{V}_c \mathbf{c}_r \quad (20)$$

The expression (20) can be split into the free and controlled dynamics by the equations (16) and (17). In order to find an exact solution for the inverse dynamic model under the form (19) and (20), it is possible to multiply (19) by $\mathbf{A}_s^T \mathbf{B}^{-T}$:

$$\mathbf{A}_s^T \mathbf{B}^{-T} \boldsymbol{\tau} = \mathbf{A}_s^T \mathbf{B}^{-T} (\mathbf{M}_a \ddot{\mathbf{q}} + \mathbf{c}_a) - \mathbf{A}_s^T \boldsymbol{\lambda} \quad (21)$$

Then, $\mathbf{A}_s^T \boldsymbol{\lambda}$ from (21) can be substituted by using the expression of the controlled dynamics (17), and equation (21) becomes:

$$\mathbf{A}_s^T \mathbf{B}^{-T} \boldsymbol{\tau} = \mathbf{A}_s^T \mathbf{B}^{-T} (\mathbf{M}_a \ddot{\mathbf{q}} + \mathbf{c}_a) - \ddot{\boldsymbol{\xi}}_c - \mathbf{V}_c \mathbf{c}_r \quad (22)$$

In order to express the dynamic model in the Cartesian space, $\ddot{\mathbf{q}}$ is mapped into the Cartesian space by using the second order inverse kinematic model, obtained from the differentiation with respect to time of equation (1):

$$\ddot{\mathbf{q}} = \mathbf{J}_{inv} \ddot{\mathbf{x}} + \mathbf{b} \quad (23)$$

where $\mathbf{J}_{inv} = -\mathbf{B}^{-1} \mathbf{A}$ and $\mathbf{b} = -\mathbf{B}^{-1} (\dot{\mathbf{A}} \dot{\mathbf{x}} + \dot{\mathbf{B}} \dot{\mathbf{q}})$.

We can perform the change of variable from $\ddot{\mathbf{x}}$ into $\ddot{\boldsymbol{\xi}}$ by using expression (15), as follows:

$$\ddot{\mathbf{x}} = \mathbf{M}_r^{-1} \mathbf{V}^T \ddot{\boldsymbol{\xi}} \quad (24)$$

Then, to separate the controlled and uncontrolled variables from (24), the following computations are performed:

$$\mathbf{V}^T \ddot{\boldsymbol{\xi}} = \mathbf{t}_s \ddot{\xi}_1 + \mathbf{V}_c^T \ddot{\boldsymbol{\xi}}_c \quad (25)$$

By introducing expression (25) into equation (24), and substituting into the second order inverse kinematic model (23), the model (22) becomes:

$$\mathbf{A}_s^T \mathbf{B}^{-T} \boldsymbol{\tau} = \mathbf{A}_s^T \mathbf{B}^{-T} (\mathbf{M}_a \mathbf{J}_{inv} \mathbf{M}_r^{-1} (\mathbf{t}_s \ddot{\xi}_1 + \mathbf{V}_c^T \ddot{\boldsymbol{\xi}}_c) + \mathbf{M}_a \mathbf{b} + \mathbf{c}_a) - \ddot{\boldsymbol{\xi}}_c - \mathbf{V}_c \mathbf{c}_r \quad (26)$$

The expression of the free dynamics (16) can be rewritten as follows:

$$\ddot{\xi}_1 = -\mathbf{t}_s^T \mathbf{c}_r \quad (27)$$

Therefore, by substituting (27) into (26) and grouping the controlled dynamics, the following expression is obtained:

$$\mathbf{J}_s^T \boldsymbol{\tau} = \mathbf{M}_s \ddot{\boldsymbol{\xi}}_c + \mathbf{c}_s \quad (28)$$

with the matrix $\mathbf{J}_s = -\mathbf{B}^{-1} \mathbf{A}_s$, a generalized inertia matrix reduced to the controllable degrees of freedom $\mathbf{M}_s = \mathbf{J}_s^T \mathbf{M}_a \mathbf{J}_{inv} \mathbf{M}_r^{-1} \mathbf{V}_c^T + \mathbf{I}_{(n-1)}$ and the term that contains the Coriolis, centrifugal, gravity and friction effects $\mathbf{c}_s = \mathbf{J}_s^T (\mathbf{M}_a \mathbf{b} + \mathbf{c}_a - \mathbf{M}_a \mathbf{J}_{inv} \mathbf{M}_r^{-1} \mathbf{c}_r) + \mathbf{V}_c \mathbf{c}_r$.

From the properties of the Moore-Penrose pseudo-inverse, the inverse dynamic model that minimizes the input torques at the singularity locus is expressed by:

$$\boldsymbol{\tau} = \mathbf{J}_s^{T+} (\mathbf{M}_s \ddot{\boldsymbol{\xi}}_c + \mathbf{c}_s) \quad (29)$$

Since the uncontrolled variable has been removed from expression (29), the matrix \mathbf{J}_s^T is full row rank and the obtained inverse dynamic model does not degenerate in the singularity locus.

V. MODELING OF THE VIRTUAL CONSTRAINTS AND CONTROL LAW IN THE SINGULARITY

In sections III-B and IV, we showed that locally in a Type 2 singularity, the dynamic model of a parallel robot can be separated in a free dynamics, associated to the local uncontrolled direction, and a controlled dynamics in the other directions. Virtual-constraint-based controllers have been shown to be efficient in the control of underactuated systems. In this section, we will apply this tool to maintain the controllable variables on a trajectory defined by the virtual constraints while the uncontrolled dynamics of the system remains free. The controller developed naturally respects the controlled and uncontrolled dynamics of the robot, avoiding model degeneracy issues obtained with a classic dynamic controller as Computed Torque Control. In [13], the authors have defined the theoretical framework in the modeling of virtual constraints. They defined virtual constraints as geometrical relations among the generalized coordinates that virtually constrain the system through the application of control inputs for keeping those constraints invariant. Recalling that the controlled and uncontrolled variables at the singularity locus are

- ξ_1 : the uncontrolled variable;
- $\xi_c = [\xi_2, \xi_3, \dots, \xi_n]^T$: the controlled variables.

It is thus possible to define a relation between the controlled and uncontrolled variables in order to define the virtual constraint to be enforced around the Type 2 singularity as follows:

$$\xi_c = \mathbf{h}(\xi_1) \quad (30)$$

In order to enforce the constraints, outputs are defined as

$$\mathbf{y} = \xi_c - \mathbf{h}(\xi_1) \quad (31)$$

The controller will drive the outputs \mathbf{y} towards zero through feedback linearization. To perform this linearization, the outputs are differentiated with respect to time twice

$$\ddot{\mathbf{y}} = \ddot{\xi}_c - (\ddot{\xi}_1 \mathbf{h}'(\xi_1) + \dot{\xi}_1^2 \mathbf{h}''(\xi_1)) \quad (32)$$

where \mathbf{h}' and \mathbf{h}'' represent the first and second derivatives with respect to the passive variable ξ_1 of the function $\mathbf{h}(\xi_1)$, respectively. The uncontrolled dynamics, known by equation (27) are introduced in equation (32).

$$\ddot{\mathbf{y}} = \ddot{\xi}_c - (-\mathbf{t}_s^T \mathbf{c}_r \mathbf{h}'(\xi_1) + \dot{\xi}_1^2 \mathbf{h}''(\xi_1)) \quad (33)$$

The objective of the controller is to obtain a convergence of the output \mathbf{y} towards $\mathbf{0}$ as a second order ordinary differential equation. To obtain it, an auxiliary input \mathbf{v} corresponding to the acceleration of the controlled variables ξ_c is introduced, and defined to ensure the convergence of the virtual constraints

$$\mathbf{v} = (-\mathbf{t}_s^T \mathbf{c}_r \mathbf{h}'(\xi_1) + \dot{\xi}_1^2 \mathbf{h}''(\xi_1)) - \mathbf{K}_p \mathbf{y} - \mathbf{K}_d \dot{\mathbf{y}} \quad (34)$$

where \mathbf{K}_p and \mathbf{K}_d are positive definite matrices of proportional and derivative gains to be tuned. Imposing torques such that $\ddot{\xi}_c = \mathbf{v}$ will lead to the closed-loop equation

$$\ddot{\mathbf{y}} + \mathbf{K}_d \dot{\mathbf{y}} + \mathbf{K}_p \mathbf{y} = \mathbf{0} \quad (35)$$

ensuring the convergence of \mathbf{y} towards zero. The auxiliary input is obtained by an input-output linearization through the dynamic model of the controlled variables (29).

$$\boldsymbol{\tau} = \mathbf{J}_s^{T+} (\mathbf{M}_s \mathbf{v} + \mathbf{c}_s) \quad (36)$$

The designed virtual-constraint-based control law (Fig. 2) only applies for the situation when the robot becomes underactuated in the singularity locus. Nevertheless, the goal is to continuously and automatically track a trajectory in the workspace far from the singularity and in the singularity locus. Thus, the present paper proposes the use of a multi-control architecture that switches between two control laws. In [8] it has been shown that in order to track a given trajectory out of the singularity locus, a classical computed torque control can be implemented. The second control law is the virtual-constraint-based controller for tracking in the singularity locus.

In order to design a Multi-Controller scheme, a supervisory control architecture is implemented (for a global overview on Supervisory Control the reader can refer to [15]). Fig. 3 shows the Multi-Controller scheme integrating the control law based on virtual constraints \mathbf{u}_1 and the classical computed torque control \mathbf{u}_2 .

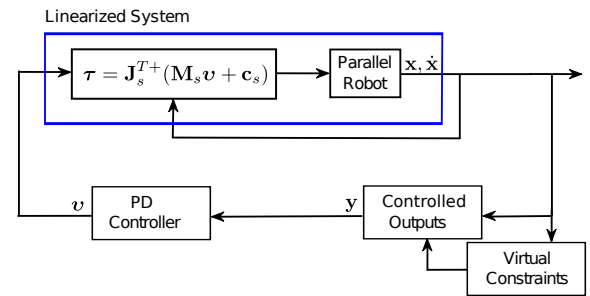


Fig. 2. Virtual-constraint-based Control scheme is depicted with the vector of control inputs $\boldsymbol{\tau}$ representing the input torques in the singularity locus. The box in blue highlights the linearized system, and the PD controller ensures the convergence of the virtual constraints.

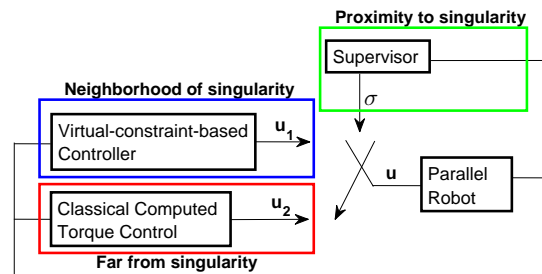


Fig. 3. Multi-Controller scheme is depicted with the designed virtual-constraint-based control law near to the singularity and the classical computed torque control far from the singularity. The supervisor block decides when to switch the control laws based on the proximity to the singularity locus.

VI. CASE STUDY

A. Five-bar mechanism: Benchmark in Type 2 singularity crossing

The proposed control approach for crossing Type 2 singularities was validated in a five-bar mechanism from Meca-

demec, as shown in Fig. 4. The actuation is provided by $\mathbf{q} = [q_{11}, q_{21}]^T$. The vector of passive joints is given by $\mathbf{q}_d = [q_{12}, q_{22}, q_{13}]^T$. The vector of moving platform pose is given by $\mathbf{x} = [x, y]^T$. a is the distance between the actuated joints. All link lengths l are identical.

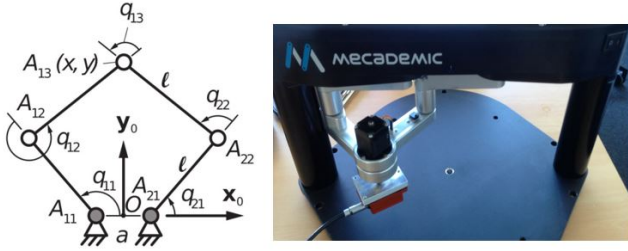


Fig. 4. On the left the five-bar mechanism parametrization is presented. On the right the real robot is shown with an Inertial Measurement Unit sensor attached in the end-effector. The five-bar mechanism is a planar parallel robot with two actuated joints at A_{11} and A_{21} , and three passive joints located at A_{12} , A_{22} and A_{13} .

The dynamic model of the five-bar mechanism from Mecademic with identified parameters was given in [16]. The dynamic model has the following form:

$$\boldsymbol{\tau} = \mathbf{ZZ}\ddot{\mathbf{q}} + \mathbf{B}^T \boldsymbol{\lambda} + \mathbf{f} \quad (37)$$

$$\mathbf{w}_p = \mathbf{A}^T \boldsymbol{\lambda} = m\ddot{\mathbf{x}} \quad (38)$$

where \mathbf{A} and \mathbf{B} are computed from the input-output kinematic constraint relation (1). The matrix \mathbf{ZZ} is positive diagonal resulted from the identified dynamic parameters, and \mathbf{f} groups the active joint friction terms.

B. Virtual-constraint-based control implementation

In the case of the five-bar mechanism, the general expression of the matrix \mathbf{A}^T in a Type 2 singular configuration as in Fig. 1 is given by

$$\mathbf{A}^T = \begin{bmatrix} \cos \beta & -\cos \beta \\ \sin \beta & -\sin \beta \end{bmatrix} \quad (39)$$

with β the orientation of the alignment of the passive legs

$$\beta = q_{11} + q_{12} + q_{13} = q_{21} + q_{22} \quad (40)$$

In order to obtain the separation between the free and the controlled dynamics, the matrix \mathbf{V} of equation (9) is defined by

$$\mathbf{V} = \begin{bmatrix} -\sin \beta & \cos \beta \\ \cos \beta & \sin \beta \end{bmatrix} \quad (41)$$

and the equation (38) is pre-multiplied by the matrix \mathbf{V}

$$m\mathbf{V}\ddot{\mathbf{x}} = \begin{bmatrix} 0 & 0 \\ 1 & -1 \end{bmatrix} \boldsymbol{\lambda} \quad (42)$$

The variable $\boldsymbol{\xi} = [\xi_1 \ \xi_c]^T$ is defined by

$$\boldsymbol{\xi} = m\mathbf{V}\mathbf{x} \quad (43)$$

Giving the uncontrolled dynamics

$$\ddot{\xi}_1 = 0 \quad (44)$$

and the inverse dynamic model (29) associated to the controllable variables for the five-bar mechanism at singular configuration

$$\boldsymbol{\tau} = \mathbf{J}_s^{T+} (\mathbf{M}_s \ddot{\boldsymbol{\xi}}_c + \mathbf{c}_s) \quad (45)$$

with $\mathbf{J}_s = -\mathbf{B}^{-1}\mathbf{A}_s$, $\mathbf{M}_s = 1 + \mathbf{J}_s^T \mathbf{ZZ} \mathbf{J}_s / m$, $\mathbf{c}_s = \mathbf{J}_s^T \mathbf{ZZ} \mathbf{b} - \mathbf{A}_s^T \mathbf{B}^{-T} \mathbf{f}$, $\xi_c = mx$ and $\mathbf{A}_s^T = [1 \ -1]$. This inverse dynamic model will be used to perform the input-output linearization in the control loop (see section V). The enforced virtual constraint is simply $\xi_c = 0$. Then the auxiliary control law v applied on the controlled acceleration $\ddot{\xi}_c$ is given by

$$v = -K_p \xi_c - K_d \dot{\xi}_c \quad (46)$$

C. Multi-Control architecture for integrating the control laws

The performance index considered in the supervisor block from Fig. 3, for the control laws transition, is the proximity to the singularity locus. We use the approach of computing the condition number of the matrix \mathbf{A} , which is the kinematic matrix that degenerates in the singularity locus. Any other criterion characterizing the proximity to singularity could be used. Thus, the multi-control law can be synthesized by:

$$\mathbf{u} = \sigma \mathbf{u}_1 + (1 - \sigma) \mathbf{u}_2 \quad (47)$$

where σ is equal to 0 far from the singularity and equal to 1 in the singularity locus:

- $\sigma = 1$, if $(\frac{1}{\text{cond}(\mathbf{A})}) \geq 2\epsilon$
- $\sigma = (\frac{1 - \text{cond}(\mathbf{A})\epsilon}{\text{cond}(\mathbf{A})\epsilon})$, if $\epsilon < (\frac{1}{\text{cond}(\mathbf{A})}) < 2\epsilon$
- $\sigma = 0$, if $(\frac{1}{\text{cond}(\mathbf{A})}) \leq \epsilon$

ϵ is a threshold to be tuned experimentally.

D. Trajectory generation

In order to validate the theoretical formulations, the controller is tested by defining sections of trajectory crossing the singularity using fifth-degree order polynomials which can fix the position at the sections extremities with null initial and final velocities and accelerations. As the uncontrolled dynamic is taken into account in the controller, it is not necessary to design those sections such that the dynamic criterion established in previous studies [7] is respected. The trajectory will be automatically modified to respect the free dynamics when crossing the singularity and the classic CTC controller catches up with the initial trajectory after the crossing. The final trajectory consists of multiple sections crossing the singularity locus switching between the two different assembly modes of the five-bar mechanism. The sections of trajectories are defined between the points given by $A = [0, 0.15]$, $B = [0, 0.035]$, $C = [-0.025, 0.025]$, $D = [-0.025, 0.125]$, $E = [0.025, 0.125]$, $F = [0.025, 0.025]$. The sequence of the trajectory is defined as the Fig. 5.

E. Experimental results

This section presents the results of testing the controller for the trajectory defined in section VI-D. Fig. 6 shows the input torques computed from the multi-control law along the trajectory with multiple crossings from one assembly mode

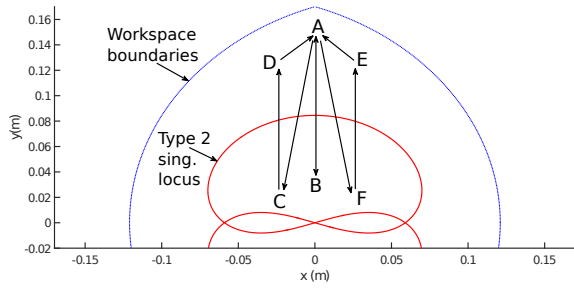


Fig. 5. Multiple crossing trajectory with the sequence: $A \rightarrow B \rightarrow A \rightarrow B \rightarrow A \rightarrow C \rightarrow D \rightarrow A \rightarrow F \rightarrow E \rightarrow A$.

to the other. The five-bar mechanism crosses the singularity without torque discontinuities. In order to validate the virtual-constraint-based control law in the multi-control architecture, an *IMU* was attached in the platform of the robot. Fig. 7 shows the tracking error of the magnitude of the acceleration of the end-effector. The error is expressed as a function of the desired trajectory. Since the controller automatically corrects the trajectory when crossing the singularity with respect to the controlled and uncontrolled dynamics, the perturbation of the trajectory tracking when crossing is expected. Furthermore, model errors may have an influence over the perturbations observed. Nevertheless, the overall error in acceleration remains converging towards zero, and varying in a range of 10^{-1} m/s^2 . The robustness in experimentation of the proposed control architecture was observed by testing the trajectory at least 10 times. In all the tests the robot successfully crosses the singularities automatically.

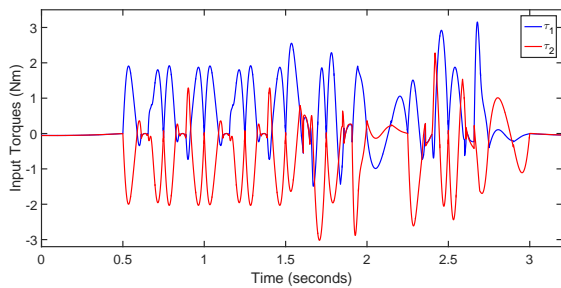


Fig. 6. Input torques for the first and the second actuator.

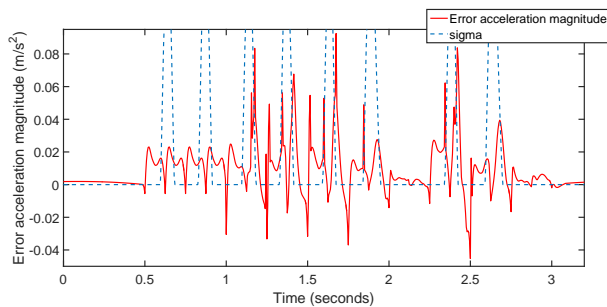


Fig. 7. Tracking error of the magnitude of the acceleration for the multi-control law.

VII. CONCLUSIONS

This paper proposes a new solution to cross Type 2 singularities on a parallel robot. An analysis of the dynamic

equations in the singularity locus provided the separation of the uncontrolled and controlled dynamics. Then, a specific controller using an approach based on virtual constraints was developed to deal with the local underactuation in the singularity. A multi-control law was then implemented to toggle between the local controller near the singularity locus and a classic computed torque control in the rest of the workspace. This controller allowed the crossing of singularities without the pre-planning of an optimal trajectory based on the dynamic criterion identified in previous studies. Experimental implementations led to the successful crossing of singularities in several configurations on a five-bar mechanism. Future work on this subject includes the theoretical proof of robustness of the multi-control law implemented and the design of alternative constraints to minimize the divergence from the initial trajectory when performing the singularity crossing.

REFERENCES

- [1] Merlet, J-P: Parallel Robots. Springer, 2006. ISBN 978-1-4020-4133-4.
- [2] Chablat, D., Wenger Ph.: Working modes and aspects in fully-parallel manipulator. In Proceedings of International Conference on Robotics and Automation, Bruxelles, Belgium, pp. 1964-1969, May 1998.
- [3] Gosselin, C. and Angeles, J.: Singularity analysis of closed-loop kinematic chains. IEEE Transactions On Robotics And Automation, Vol. 6, No. 3, pp. 281-290, June 1990.
- [4] Conconi, M. and Carricato, M.: A new assessment of singularities of parallel kinematic chains. IEEE Transactions on Robotics, Vol. 25, No.4, pp. 757-770, August 2009.
- [5] G. Gogu: Structural synthesis of fully-isotropic parallel robots via theory of linear transformations and evolutionary morphology, European Journal of Mechanics, A/Solids, Vol. 23, pp. 1021-1039, 2004.
- [6] Kurtz,R., and Hayward, V.: Multiple-goal kinematic optimization of a parallel spherical mechanism with actuator redundancy. IEEE on Robotics and Automation, Vol. 8, No. 5 pp. 644-651, October 1992.
- [7] S. Briot and V. Arakelian: Optimal Force Generation of Parallel Manipulators for Passing through the Singular Positions, The International Journal of Robotics Research, Vol. 27, No. 8, pp. 967-983, 2008.
- [8] Pagis, G., Bouton, N., Briot, S. and Martinet, P.: Enlarging parallel robot workspace through Type-2 singularity crossing. Control Engineering Practice, pp.1-11, June 2009.
- [9] Shiriaev, A., Perram, J., and Canudas-de-Wit, C.: Constructive tool for orbital stabilization of underactuated nonlinear systems: Virtual constraints approach. IEEE Transactions on Automatic Control, Vol. 50, No. 8, pp. 1164-1176, August 2005.
- [10] Westervelt, E., Grizzle, J., Chevallereau, C., Choi, J. and Morris B.: Feedback Control of Dynamic Bipedal Robot Locomotion, Series: Automation and Control Engineering. CRC Press, June 2007. ISBN: 9781420053722.
- [11] Canudas-de-Wit, C., Espiau, B., and Urrea, C.: Orbital stabilization of underactuated mechanical systems. 15th World Congress IFAC, Barcelona, Spain, July 2002.
- [12] Ahmed, M.S., Hably, A. and Bacha, S.: Kite generator system periodic motion planning via virtual constraints. 39th Annual Conference of the IEEE Industrial Electronics Society, Vienne, Austria, pp.1-6, November 2013.
- [13] Shiriaev, A., Freidovich, L. and Gusev, S.: Transverse linearization for controlled mechanical systems with several passive degrees of freedom. IEEE Transactions on Automatic Control, Vol. 55, No. 4, pp. 893-906, April 2010.
- [14] S. Briot and W. Khalil, Dynamics of Parallel Robots: From Rigid Bodies to Flexible Elements. Springer, 2015. ISBN: 978-3-319-19788-3.
- [15] Liberzon, D.: Switching in Systems and Control. Birkhuser Boston, 2003. ISBN: 978-1-4612-6574-0.
- [16] S. Briot, N. Bouton and P. Bigras: Controlling Parallel Robots during Singular Assembly Mode Changing. Proceedings of the 4th Joint International Conference on Multibody System Dynamics. Montreal, Canada, May 2016.