Design of a Controller for Enlarging Parallel Robots Workspace through Type 2 Singularity Crossing

Georges Pagis[1,2,3], Nicolas Bouton[2], Sébastien Briot[1] and Philippe Martinet[1,3]

Abstract— In order to increase the workspace size of parallel robots (largely reduced by the presence of singularities) several solutions have been proposed. One promising solution consists in the definition of optimal trajectories that ensure the non degeneracy of the dynamic model in the singularity and therefore are able to cross the Type 2 singularities. Those works are based on the computation of the optimal trajectories and assume that the robot can perfectly track the desired trajectory. Nevertheless, this assumption cannot be verified in reality due to modelling errors which largely impact the control law used to follow the desired trajectory. Therefore, if the optimal trajectory is not perfectly tracked, the dynamic model can degenerate near the Type 2 singularities and the robot might stay blocked.

In order to solve that problem, this paper proposes a multi-model approach that allows parallel robots to cross the Type 2 singularities without any torque discontinuity. The main idea is to shift near singularities from the full robot dynamic model to another simplified one that can never degenerate. The proposed control law is then coupled with an optimal trajectory planning methodology that makes the singularity crossing more robust to modelling errors. The proposed approach is validated experimentally on a prototype of Five-bar planar parallel mechanism.

I. INTRODUCTION

Parallel manipulators have many advantages in terms of acceleration capacities and payload-to-weight ratio, but one of their main drawbacks concerns the presence of singularities [1], [2], [3] which divide the workspace into different aspects, each aspect corresponding to one (or more) assembly mode [4]. For a global overview of the singularity problem, the reader is referred to [2].

Type 2 (or parallel) singularities [3] are probably the most constraining singularities of parallel manipulators. In those singularities, one (or more) manipulator's degree of freedom becomes uncontrollable. Moreover, Type 2 singularities divide the manipulator's workspace in different aspects, resulting in the decrease of the reachable workspace size. Therefore, several approaches have been envisaged in the literature in order to increase the workspace size such as:

- The design of parallel robots without singularities [5], [6].
- The use of redundancy [9], [8] or of mechanisms with variable actuation modes [1], [10].
- Planning assembly mode changing trajectories by either by-passing a cusp point [11] or directly going through

a Type 2 singularity [12], [13].

In [12], a physical criterion, obtained through the analysis of the dynamic model, is presented. Based on the mentioned criterion, it is possible to plane a singularity crossing trajectory which (i) avoids the degeneracy of the dynamic model (i.e. obtaining infinite joint reactions during the motion) and (ii) allows the robot to cross the singularity.

This last solution is promising since it can considerably increase the workspace size of any parallel mechanism. However, in previous works, it has been considered that the controller allowed the mechanism to perfectly track the desired trajectory. This is obviously impossible due to modelling uncertainties. In order to fill this gap, the aim of this paper is to propose an advanced control law dedicated to the Type 2 singularity crossing.

To correctly track trajectories, several control approaches can be used [14], but the most effective is probably the Computed Torque Control (*CTC*) law [14], [15], [16]. More precisely, this control law permits to compute the actuators torque of the robot based on its dynamic model and the desired trajectory to follow. Obviously, *CTC* is sensitive to the robot dynamic model that (i) must be well identified [21], [22] and (ii) must not degenerate near singularities, even if the trajectory does not perfectly respect the physical criterion mentioned above.

Therefore, in order to avoid the dynamic model degeneracy near the singularity, a multi-model *CTC* (e.g. see [15], [16]) is proposed in the present paper. This controller is combined with the definition of new singularity-crossing criteria developed for increasing the robustness of the control law along the desired trajectory.

II. TRAJECTORY GENERATION FOR CROSSING A TYPE 2 SINGULARITY

A. Dynamic modelling of parallel mechanisms

This section will briefly recall the dynamic equations of a parallel manipulator composed of m links, n degrees of freedom (*dof*) and driven by n actuators. The position and the speed of the manipulator can be fully described using:

- $\mathbf{q} = [q_1, q_2, ..., q_n]^T$ and $\dot{\mathbf{q}} = [\dot{q}_1, \dot{q}_2, ..., \dot{q}_n]^T$ that represent respectively the vector of active joints variables and active joints velocities,
- $\mathbf{x} = [x, y, z, \phi, \psi, \theta]^T$ and $\mathbf{v} = [\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\psi}, \dot{\theta}]^T$ that are the end-effector pose parameters and their derivatives with respect to time, respectively. x, y and z represent the position of the platform controlled point and ϕ, ψ and θ the orientation parameters of the platform about three axes $\mathbf{a}_\phi, \mathbf{a}_\psi$ and $\mathbf{a}_\theta$.

[1] Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN), UMR CNRS 6597, 44321 Nantes, France {Sebastien.Briot,Philippe.Martinet}@irccyn.ec-nantes.fr
[2] Institut Pascal, IFMA, MMS department, UMR CNRS 6602, 63000 Clermont-Ferrand, France {Georges.Pagis, Nicolas.Bouton}@ifma.fr
[3] LUNAM, Ecole Centrale de Nantes, 44321 Nantes, France

Using the Lagrangian formalism, the mechanism's dynamic model can be written as:

$$\tau = \mathbf{w}_b + \mathbf{B}^T \lambda, \tag{1}$$

$$\mathbf{w}_p = \mathbf{A}^T \lambda \tag{2}$$

where

- τ is the vector of the input efforts,
- λ is the vector of the Lagrange multipliers,
- $\mathbf{A}$ and $\mathbf{B}$ are two matrices deduced from the mechanism loop-closure equations, such that $\mathbf{A}\mathbf{v} = \mathbf{B}\dot{\mathbf{q}}$ [4],
- $\mathbf{w}_b$ and $\mathbf{w}_p$ are related to the Lagrangian L of the system by:

$$\mathbf{w}_b = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} \quad , \quad \mathbf{w}_p = \frac{d}{dt}\left(\frac{\partial L}{\partial \mathbf{v}}\right) - \frac{\partial L}{\partial \mathbf{x}} \tag{3}$$

In this expression, $\mathbf{w}_p$ is the wrench applied to the platform by the legs and external forces [12].

Then, assuming that the matrix $\mathbf{A}$ can be inverted, by substituting (2) into (1) the general dynamic model of parallel manipulators is obtained [14]:

$$\tau = \mathbf{w}_b + \mathbf{J}^T \, {}^0\mathbf{w}_p, \tag{4}$$

where

- ${}^0\mathbf{w}_p$ is the expression of the wrench $\mathbf{w}_p$ in the base frame, i.e. ${}^0\mathbf{w}_p = \mathbf{D}\mathbf{w}_p$ with $\mathbf{D}$ the matrix relating the platform twist $\mathbf{t}$ (expressed in the base frame) to the vector $\mathbf{v}$ by $\mathbf{t} = \mathbf{D}\mathbf{v}$ [4],
- $\mathbf{J} = {}^0\mathbf{A}^{-1}\mathbf{B}$ is the Jacobian matrix between the platform twist $\mathbf{t}$ and $\dot{\mathbf{q}}$, with ${}^0\mathbf{A}$ is the expression of the matrix $\mathbf{A}$ in the base frame, i.e. ${}^0\mathbf{A} = \mathbf{A}\mathbf{D}^{-1}$.

B. Type 2 singularity crossing

Based on the analysis of the kinematic model, the authors of [3] proposed a classification of the singularities in three different types:

Type 1 singularities or serial singularities occur when the mechanism is in a position such as the kinematic matrix $\mathbf{B}$ becomes rank deficient. In such configurations, the mechanism loses the ability to move in one given direction.

Type 2 singularities or parallel singularities occur when the kinematic matrix ${}^0\mathbf{A}$ becomes rank deficient. In Type 2 singularities, one (or more) robot degree of freedom becomes uncontrollable. Such singularities divide the workspace in different aspects, resulting in a reduction of the manipulator's workspace. Moreover, in the presence of such singularities, the robot may also not be able to resist to an external wrench applied on the platform and the reactions in joints grow to infinity.

Type 3 singularities are configurations where both Type 1 and Type 2 singular configurations appear at the same time. They are discarded in the following of the paper as they appear if both Types 1 and 2 singularity exist.

Finally parallel mechanism with less than 6 *dof* can have another type of singularity such as the constraint singularities [17], [2].

If a parallel mechanism is in a singular position of Type 2, the matrix $\mathbf{A}^T$ cannot be inverted in Equation (2). The dynamic model degenerates and therefore cannot be solved. However, as explained in the introduction, it has been proven in [12] that a mechanism can cross a Type 2 singularity without torque discontinuity. Indeed, on a Type 2 singularity, the columns of ${}^0\mathbf{A}$ are linearly dependent, i.e. there exist a vector $\mathbf{t}_s$ such that:

$$ {}^0\mathbf{A}\mathbf{t}_s = 0 \Leftrightarrow \mathbf{t_s}^T \, {}^0\mathbf{A}^T = 0 \tag{5}$$

The vector $\mathbf{t}_s$ represents the twist of the uncontrollable motion of the platform in the singularity locus [4]. Thus, multiplying (2) by $\mathbf{t}_s^T$ leads to:

$$\mathbf{t}_s^T \, {}^0\mathbf{A}^T \lambda = 0 \tag{6}$$

In that case, the following condition must also be satisfied:

$$\mathbf{t}_s^T \, {}^0\mathbf{w}_p = 0 \tag{7}$$

which is the condition for the non-degeneracy of the dynamic model [12].

As a result, if the desired manipulator motion doesn't guarantee the achievement of a wrench $\mathbf{w}_p$ that respects the condition (7), the dynamic model is degenerated and the manipulator desired input efforts must grow to infinity to produce the desired platform motion. Physically, this condition means that the parallel manipulator can cross the Type 2 singularity if and only if the wrench ${}^0\mathbf{w}_p$ exerted by the legs and external efforts on the platform is reciprocal to the twist $\mathbf{t_s}$ of the uncontrollable motion in the Type 2 singularity.

C. Generation of a robust trajectory for crossing Type 2 singularity

In order to cross a singularity without torque discontinuity, the mechanism has to follow a trajectory that respects the criterion (7) on the singularity locus. Theoretically, the dynamic model degenerates only on the singularity locus, however *numerically* the matrix $\mathbf{A}$ is singular on a space around the singularity locus. Therefore the criterion (7) has to be respected in this space around the singularity locus in order to prevent the dynamic model from degenerating.

The trajectory generation is achieved using polynomials, which degree can vary. Indeed, in order to guarantee that the criterion (7) is respected around the singularity locus, it is proposed *in this work* to vanish the criterion (7) and n of its derivatives:

$$\mathbf{t}_s^T \frac{d^i \mathbf{w}_p}{dt^i} = 0 \quad i = 1, ..., n, \tag{8}$$

To the best of our knowledge, this is the first time that such criteria are proposed. Generating a trajectory based on these criteria (8) allows to increase the robustness to model uncertainties and control error around the singularity. Experimental results shows that nullifying the 2 first derivatives of the criterion might be enough.

Next section will present the control law used to allow the singularity crossing.

III. Control law dedicated to Type 2 singularity crossing

A. Computed Torque Control

The Computed Torque Control (*CTC*) [16] is a control law that computes the input torques that the actuators must apply on the mechanism in order to follow a given trajectory. It is based on the dynamic model presented in Part II. As for any type of control law in robotics, the aim of the controller *CTC* is to minimize the error in either joint or task space. Since at proximity of a Type 2 singularity the kinematic matrix **A** is singular, it is not possible to compute the Cartesian velocities from the joint velocities using the *DKM* (Direct kinematic Model). Sensors being usually measuring joint space values, only the joint space control law should be used to cross Type 2 singularities,.

The control error **e** to minimize can be expressed as:

$$\mathbf{e} = \mathbf{q}_d - \mathbf{q} \ \Rightarrow \ \dot{\mathbf{e}} = \dot{\mathbf{q}}_d - \dot{\mathbf{q}} \ \Rightarrow \ \ddot{\mathbf{e}} = \ddot{\mathbf{q}}_d - \ddot{\mathbf{q}} \qquad (9)$$

where

- $\mathbf{q}_d$ (respectively $\dot{\mathbf{q}}_d$ and $\ddot{\mathbf{q}}_d$) is the desired joint position (respectively velocity and acceleration),
- $\mathbf{q}$ (respectively $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$) is the current measured joint position (respectively velocity and acceleration).

In order to force the error to tend to zero, a second order differential equation on **e** is imposed by:

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = 0 \qquad (10)$$

By substituting Eq. (9) into Eq. (10) one can obtain:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} \qquad (11)$$

Let us rewrite the dynamic model of the mechanism as [16], [18]:

$$\tau = \mathbf{w}_b + \mathbf{J}^T \mathbf{w}_p = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q},\dot{\mathbf{q}}) \qquad (12)$$

Consequently, *CTC* computes the input torques by substituting Eq. (11) into the dynamic model presented in Eq. (12):

$$\tau = \mathbf{M}\left(\ddot{\mathbf{q}}_d + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e}\right) + \mathbf{H}(\mathbf{q},\dot{\mathbf{q}}) \qquad (13)$$

It should be noted that the vector of positions in the task space **x** is necessary to compute matrices **M** and **H**. However, most mechanisms have sensors measuring the vector of positions in the joint space **q**. Therefore the *DKM* is needed. Unfortunately, when planning a Type 2 singularity crossing trajectory, the mechanism changes its assembly mode, and so the solution of the *DKM* has to change. To do so, the controller need the information that the mechanism changed its assembly mode which cannot be deduced from the joint positions. Experimentally, the most reliable solution is to choose the *DKM* solution based on the desired trajectory.

It should be mentioned that, when using the proposed *CTC* for crossing Type 2 singularities, the trajectory planned for respecting the crossing criteria (cf. Part II) will be different from the real one due to errors in the dynamic model. As

a result, speaking numerically, the **crossing criteria (7) will never be respected** and the computed torque control could send infinite torques to the robot that will prevent the singularity crossing due to the inversion of the matrix **A**.

Next section presents a multi-model control law allowing the controller to avoid this issue.

B. Multi-model control law

To avoid these numeric issues, the proposed solution is to plan a trajectory respecting around the singularity locus the criterion:

$$\mathbf{w}_p = 0 \qquad (14)$$

This new criterion still guarantees that the dynamic criterion $\mathbf{t_s}^T \mathbf{w}_p = 0$ is respected. Moreover, the first n derivatives of this new criterion are also nullified. The direction $\mathbf{t_s}$ being time independent, this guarantees that the first n derivatives of the dynamic criterion are also null as presented in Section II-C. Of course during the real robot displacement, numerically $\mathbf{w}_p$ will not be null, but such a new criterion allows the implementation of a multi-model control law. The multi-model *CTC* law presented in this paper consists in using two models (Fig. (1)) :

- *Model 1* – The complete dynamic model as long as **A** is invertible:

$$\tau = \mathbf{w}_b + \mathbf{J}^T \mathbf{w}_p = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q},\dot{\mathbf{q}}) \qquad (15)$$

- *Model 2* – A reduced dynamic model that cannot degenerate when the mechanism is close to a singular position:

$$\tau = \mathbf{w}_b = \mathbf{M}'\ddot{\mathbf{q}} + \mathbf{H}'(\mathbf{q},\dot{\mathbf{q}}) \qquad (16)$$
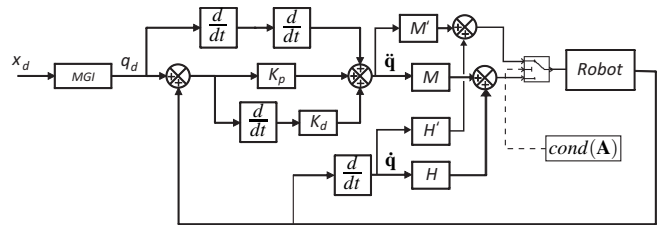

Fig. 1.   Multi-model computed torque control law

The second dynamic model is used to compute input torques only when the trajectory has been planned in order to have $\mathbf{w}_p = 0$. Considering that the control law is correctly adjusted, the effective trajectory is close enough to the desired one and therefore the hypothesis $\mathbf{w}_p = 0$ is acceptable.

Once the mechanism is far enough from the singularity locus, i.e. the matrix **A** is numerically invertible, the control switches back to the complete dynamic model and the mechanism can finish its trajectory.

Moreover, in order to cross the singularity locus, the controller need a metric that defines the moment when the *Model 2* has to be used. The discussion about the best indicator of singularity proximity is a well known problematic [4], [19], [20]. Here, as the controller may become unstable due to numerical problems linked to the inversion of the matrix **A**,

it has been decided to use the numerical condition number of that matrix to define the space where the dynamic model might degenerates and therefore where the control law has to switch between the two models (as represented by $cond(\mathbf{A})$ on Fig. (1)).

Next sections present experimental results of Type 2 singularity crossing using this multi-model control law on a planar Five-bar mechanism.

IV. CASE STUDY

A. Presentation of the Five-bar mechanism

A Five-bar mechanism is a planar parallel mechanism composed of two actuators located at the revolute joints positioned in points A and E and 3 passive revolute joints in points B, C and D (Fig. 2). The mechanism used in this work has been designed so that it can reach all the workspace positions without collision between the proximal and the distal legs.
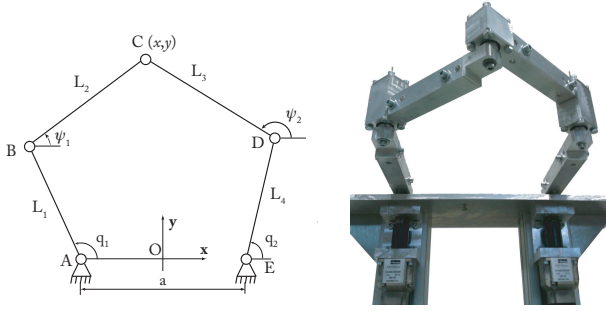

Fig. 2. Five-bar mechanism designed and manufactured at IFMA

The mechanism and its parameters are presented in Fig. 2. The links dimensions have been calibrated using a Laser Tracker (Table I).

TABLE I

FIVE-BAR MECHANISM'S GEOMETRIC PARAMETERS

Parameter	a	L_1	L_2	L_3	L_4
Value (m)	0.2822	0.2130	0.1888	0.1878	0.2130
Precision (m)	1.10^{-5}	1.10^{-5}	1.10^{-5}	1.10^{-5}	1.10^{-5}

B. Gain tunning

The proportional and derivative gains have been tuned based on the mechanism's natural frequency [14]. This frequency was retrieved using a ring-out procedure. The mechanism was excited using an impedance hammer, and its response was recorded using 5 accelerometers. The first natural frequency of the Five-bar mechanism in its isotropic configuration (when links BC and CD are perpendicular) is at 34.2 Hz.

For a given control bandwidth fixed by a frequency ω both gains are adjusted as :

$$K_p = \omega^2, \quad K_d = 2\xi\omega \quad (17)$$

where ξ is a a damping coefficient usually fixed to 1 to have a critically damped system. To guarantee that the gains do not bring the system in the neighbourhood of the instability domain, the chosen frequency must be smaller than the natural resonance frequency [14]. Therefore a frequency of $\omega = \omega_r/3$ has been chosen, resulting in gains values :

$$K_p = 1150, \quad K_d = 70 \quad (18)$$

C. Dynamic model

A full dynamic model of the robot has been computed using the methodology presented in [22] and its identification was made using a weight least square method based on the use of exciting trajectories followed with a classic geometrical control law [21]. The identification results in the following model that fully describes the robot dynamics of the studied mechanism:

$$\tau = m_3 \mathbf{J}^T \begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} + \begin{pmatrix} zz_1 \ \ddot{q}_1 \\ zz_2 \ \ddot{q}_2 \end{pmatrix} + \\ \begin{pmatrix} f_{v1}\dot{q}_1 \\ f_{v2}\dot{q}_2 \end{pmatrix} + \begin{pmatrix} f_{s1}sign(\dot{q}_1) \\ f_{s2}sign(\dot{q}_2) \end{pmatrix} \quad (19)$$

where :

- m_3 is a mass equivalent located on the end effector; $m_3 = 0.40 \pm 0.02$kg
- zz_1 and zz_2 are rotational equivalent inertial terms respectively on the first and second actuator; $zz_1 = 1.83 \cdot 10^{-2} \pm 6.97 \cdot 10^{-4}$kg.m^2; $zz_2 = 1.96 \cdot 10^{-2} \pm 6.60 \cdot 10^{-4}$kg.m^2;
- f_{s1} is a Coulomb friction term on the first actuator (respectively f_{s2} on the second actuator); $f_{s1} = 2.94 \pm 0.10$N.m; $f_{s2} = 2.95 \pm 0.09$N.m;
- f_{v1} is a viscous friction term on the first actuator (respectively f_{v2} on the second actuator); $f_{v1} = 6.76 \pm 0.018$N.m.s $f_{v2} = 6.75 \pm 0.17$N.m.s.

This identified dynamic model is related to the Eq. (2) by:

$$\mathbf{w}_p = m_3 \begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix},$$

$$\mathbf{w}_b = \begin{pmatrix} zz_1 \ \ddot{q}_1 \\ zz_2 \ \ddot{q}_2 \end{pmatrix} + \begin{pmatrix} f_{v1}\dot{q}_1 \\ f_{v2}\dot{q}_2 \end{pmatrix} + \begin{pmatrix} f_{s1}sign(\dot{q}_1) \\ f_{s2}sign(\dot{q}_2) \end{pmatrix} \quad (20)$$

It should be noted that the friction terms in both passive joints are insignificant and therefore the identification routine returned null values.

D. Control law implementation

The Five-bar mechanism is controlled by an industrial control architecture developed by ADEPT with an open architecture. This control architecture allows the user to control the mechanism either in position, speed or torques, using a C/C++ software developed by ADEPT France: CIDE. This software was designed mostly for position control, therefore every securities preventing mostly physical damage had to be developed for the computed torque control law.

The dynamic model identified contain both accelerations in the joint space and in the task space. Therefore, in order

to express the dynamic model as in Eq. (12), the task space acceleration has to be expressed as a function of the joint space acceleration. This can be done by differentiating the kinematic model:

$$\mathbf{v} = \mathbf{J}\dot{\mathbf{q}} \Rightarrow \dot{\mathbf{v}} = \dot{\mathbf{J}}\dot{\mathbf{q}} + \mathbf{J}\ddot{\mathbf{q}} \qquad (21)$$

By substituting (21) into (19) one can obtain the dynamic model used for the computed torque control law as presented in Equation (12):

$$\tau = \mathbf{M}\begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{pmatrix} + \mathbf{H}, where \qquad (22)$$

$$\mathbf{M} = m_3 \mathbf{J}^T \mathbf{J} + \begin{pmatrix} zz_1 & 0 \\ 0 & zz_4 \end{pmatrix} \qquad (23)$$

$$\mathbf{H} = m_3 \mathbf{J}^T \dot{\mathbf{J}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix} + \begin{pmatrix} f_{v1}\dot{q}_1 + f_{s1}sign(\dot{q}_1) \\ f_{v2}\dot{q}_2 + f_{s2}sign(\dot{q}_2) \end{pmatrix} \qquad (24)$$

Finally, in order to compute on-line the dynamic model of the robot and the *CTC* control law, the actuator's positions and speeds have been filtered by using the oversampling method at 1Khz (the control law turning at 250 Hz).

V. EXPERIMENTAL RESULTS

A. Generation of a crossing trajectory

The crossing trajectories are generated using two polynomials P_x and P_y such as :

$$\begin{aligned} x &= P_x(x_f - x_0) + x_0, \\ y &= P_y(y_f - y_0) + y_0 \end{aligned} \qquad (25)$$

where

- $P_x(t_0) = P_y(t_0) = 0$,
- $P_x(t_f) = P_y(t_f) = 1$,

Each of them are 8th order polynomial, corresponding to 8 conditions on each axes: two conditions for the initial position and speed, two for the final position and speed, one for the singular position and three to guarantee that the singularity crossing criteria (14) is respected around the singularity locus [12].

Figure 3 represents a crossing trajectory in the task space as well as the evolution of the task space coordinates along this trajectory and the evolution of the dynamic criteria (14) for:

$$\begin{aligned} x_0 &= 0.1, & y_0 &= 0.34, \\ x_s &= 0.05475, & y_s &= 0.2, \\ x_f &= 0, & y_f &= 0.1, \end{aligned} \qquad (26)$$

B. Type 2 singularity crossing results and process repeatability

This section presents the results of Type 2 singularity crossing for different trajectories computed according to the method presented in Section II-C and in the previous paragraph. Figure 4 represents the input torques computed by the computed torque control law along different crossing trajectories from one assembly mode to another, the desired
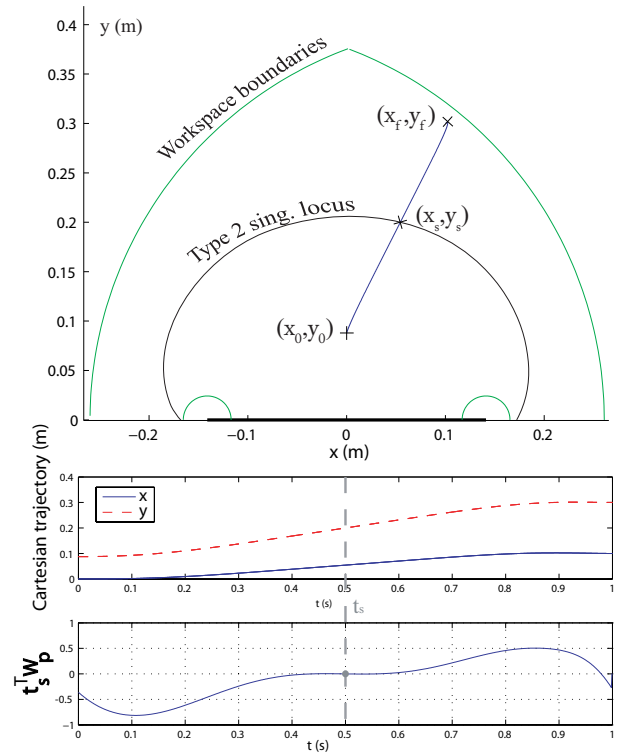

Fig. 3. Example of 8th order polynomial trajectory crossing the Type 2 singularity locus

trajectory and the control error. For each trajectory, the mechanism crosses the singularity without torque discontinuity, with a conditioning number limit set to 30 for switching between the two models.

The multi-model control law engenders an increase of the error around the singularity locus. Therefore, when the control law switches back to the complete dynamic model, the input torques can significantly increase in order to nullify this error. This can be seen on the first trajectory at 0.7s.

All three trajectories represented in Figure 4 were planned to cross the singularity at 0.5s and end at 1s. For each trajectory, the first figure represents the desired trajectory in the task space and the Type 2 singularity of the mechanism.

For testing the robustness of the proposed controller, each trajectory is run five times. Moreover, the starting point and the ending point were chosen randomly and neither those points nor the crossing direction have any effect on the singularity crossing. During all our experiments using this controller, the robot has always successfully crossed the singularity loci without any difficulty. Thus, our controller is totally robust to the desired trajectory.

VI. CONCLUSION

One of the most important drawbacks of parallel robots is the small size of their workspace, which is moreover reduced by the presence of singularities. In order to increase the workspace size, several solutions have been proposed. One promising approach is the definition of optimal trajectories able to cross the Type 2 singularities. However,
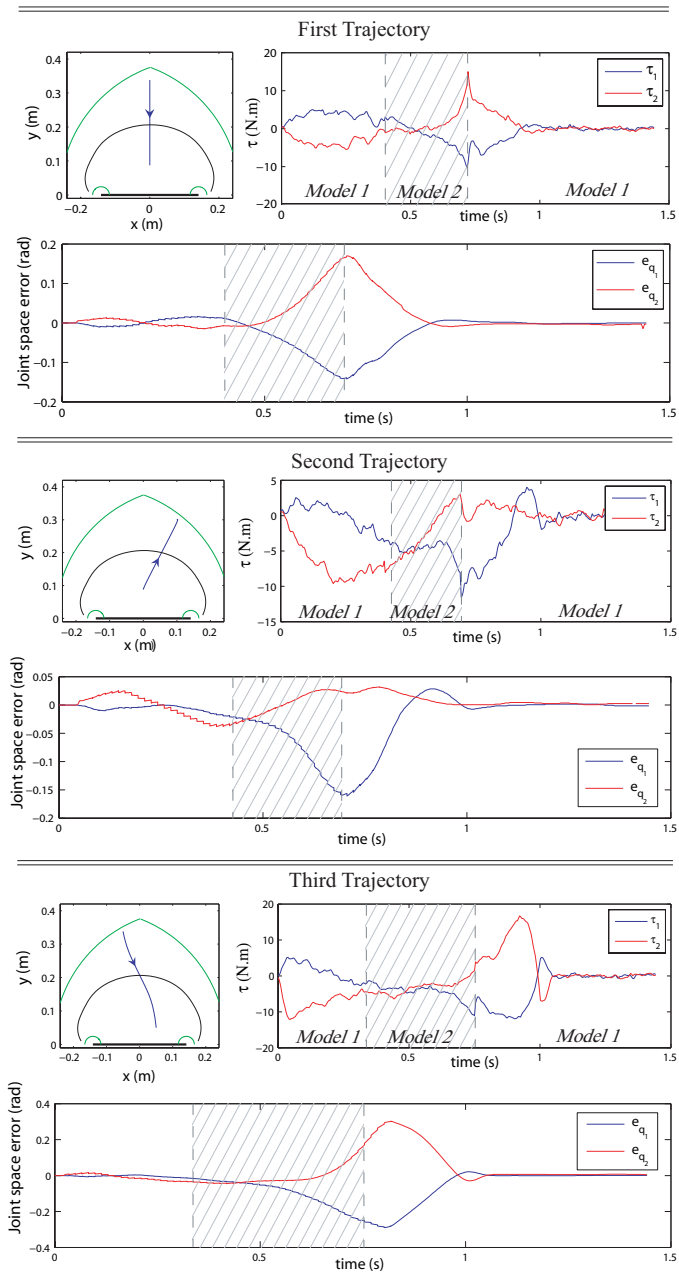
Fig. 4. Different crossing trajectories and corresponding input torque applied

singularities from the full robot dynamic model to another simplified one that can never degenerate. This controller was coupled with an optimal trajectory planning methodology that makes the singularity crossing more robust to modelling errors.

Such a controller has been validated experimentally on a prototype of a Five-bar planar parallel mechanism. The results have shown the robustness and the relevancy of the controller dedicated to parallel robot Type 2 singularity crossing.

REFERENCES

[1] Arakelian, V., Briot, S., & Glazunov, V., 2008. Increase of singularity-free zones in the workspace of parallel manipulators using mechanisms of variable structure. Mechanism and Machine Theory.

[2] Conconi, M., & Carricato, M., 2009. A new assessment of singularities of parallel kinematic chains IEEE Transactions on Robotics, 25(4), pp. 757-770.

[3] Gosselin, C., & Angeles, J. (1990). Singularity analysis of closed-loop kinematic chains. Robotics and Automation, IEEE Transactions on, 6(3), 281-290.

[4] Merlet, J. P. (1993). Closed-form resolution of the direct kinematics of parallel manipulators using extra sensors data. and Automation, 1993. Proceedings., 1993 IEEE, 54(2), 200-204.

[5] Briot, S., Pashkevich, A., & Chablat, D., 2010. Optimal technology-oriented design of parallel robots for high-speed machining applications. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA 2010).

[6] Gogu, G., 2004. Structural synthesis of fully-isotropic translational parallel robots via theory of linear transformations. European Journal of Mechanics. A/Solids.

[7] Alexandre Figielski, Ilian A. Bonev, Pascal Bigras, Towards Development of a 2-DOF Planar Parallel Robot with Optimal Workspace Use, 2007 IEEE International Conference on Systems, Man, and Cybernetics, Montreal, QC, Canada, 7-10 October 2007.

[8] R. Kurtz and V. Hayward, Multiple-goal kinematic optimization of a parallel spherical mechanism with actuator redundancy, IEEE Transactions on Robotics and Automation, vol. 8, no. 5, pp. 644-651, 1992.

[9] M. A. Nahon and J. Angeles, Force optimization in redundantly actuated closed kinematic chains, in Proceedings of the International Conference on Robotics and Automation, 1989.

[10] N Rakotomanga, D Chablat, S Caro, Kinetostatic performance of a planar parallel mechanism with variable actuation, Advances in Robot Kinematics, 2008.

[11] Zein, M., Wenger, P., & Chablat, D., 2008. Non singular assembly-mode changing motions for 3-RPR parallel manipulators. Mechanism and Machine Theory, 43(4), pp. 480-490.

[12] Briot, S., Arakelian, V., & Chablat, D. (2008). Optimal Force Generation in Parallel Manipulators for Passing through the Singular Positions. The International Journal of Robotics Research, 27(8), 967-983.

[13] Ider, S. K., 2005. Inverse dynamics of parallel manipulators in the presence of drive singularities. Mechanism and Machine Theory, 40, pp. 33-44.

[14] Khalil, W., & Dombre, E. (2004). Modeling, identification & control of robots.

[15] Craig, J. J., & Hall, P. P. (n.d.). Introduction to Robotics.

[16] Spong, M., & Vidyasagar, M. (2008). Robot dynamics and control.

[17] Zlatanov, D., & Bonev, I. (2002). Constraint singularities of parallel mechanisms. Robotics and Automation.

[18] Ghorbel, F., Chetelat, O., & Longchamp, R. (1994). A reduced model for constrained rigid bodies with application to parallel robots. IFAC symposium on robot Control SYROCO'94, (3), 57-62.

this solution assumes that the controller is not subject to modelling errors and that the robot is able to perfectly track the desired trajectory. Nevertheless, due to modelling errors, this assumption cannot be verified. As a consequence, if the optimal trajectory is not perfectly tracked, the dynamic model can degenerate near the Type 2 singularities, this can cause the computation of infinite torques and the robot might stay blocked into the singularity.

In order to avoid this problem, this paper has proposed a controller dedicated to the Type 2 singularity crossing. This controller was based on a multi-model approach that allowed the parallel robots to cross the Type 2 singularities without any torque discontinuity. The main idea was to shift near

[19] Glazunov, V., Arakelian, V., Briot, S., & Rashoyan, G. V. (2011). Speed and Force Criteria for the Proximity to Singularities of Parallel Structure Manipulators. Machinery Mechanics.

[20] Voglewede, P. A., & Ebert-Uphoff, I. (2004). Measuring closeness to singularities for parallel manipulators. IEEE Int. Conf. on Robotics and Automation, 4539-4544.

[21] Gautier, M. (1997). Dynamic identification of robots with power model. Robotics and Automation 1997, 3, 1922-1927.

[22] Briot, S., & Gautier, M. (2012). Global Identification of Drive Gains and Dynamic Parameters of Parallel Robots. Proceeding of the 19th CISM-Iftomm Symposium, 93-100.