Controlling Parallel Robots during Singular Assembly Mode Changing

Sébastien Briot[1], Nicolas Bouton[2] and Pascal Bigras[3]

[1]*IRCCyN, UMR CNRS 6597 (Nantes, France), Sebastien.Briot@irccyn.ec-nantes.fr*
[2]*Sigma, Institut Pascal, UMR CNRS 6602 (Clermont-Ferrand, France), nicolas.bouton@sigma-clermont.fr*
[3]*CoRo Lab., ETS (Montréal, Canada), Pascal.Bigras@etsmtl.ca*

ABSTRACT — *In order to increase the reachable workspace of parallel robots, a promising solution consists of the definition of optimal trajectories that ensure the non-degeneracy of the dynamic model in the Type 2 (or parallel) singularity and allow to cross it. However, this assumes that the control law can perfectly track the desired trajectory, which is impossible due to modeling errors.*

This paper improves the results obtained previously by two of the authors of the present work that defined a robust multi-model approach allowing parallel robots to cross Type 2 singularities. The main idea was to shift near singularities the general dynamic model to a simplified one that can never degenerate. However, because in Type 2 singularity, the robot is locally overconstrained, we will show that the simplified dynamic model admits infinity of exact solutions. We propose to use the solution leading to a null overconstraint in order to avoid undesirable effects near singularities, such as buckling or unstability.

The mentioned multi-model control approach is modified accordingly and it is then applied on a prototype of five-bar mechanism to show the effects of correctly and not-correctly managed overconstraint.

1 Introduction

Probably, the most important drawback of parallel robots is the presence of Type 2 (or parallel) singularities [1] in their workspace which divide it into different aspects (each aspect corresponding to one or more assembly modes [2]) and near which their performance is drastically reduced. Many strategies for enlarging the size of the operational workspace of parallel robots have been defined (e.g. optimal design [3], redundancy [4, 5], working mode changing [6]). However, all these methods have some limitations: they require the architectural modifications or they cannot necessary lead to the access of the total workspace.

In order to increase the size of the reachable workspace, methods for working mode or assembly mode changing have been more recently proposed [6, 7, 8, 9]. One of them consists in crossing the Type 2 singularities [1] by defining optimal trajectories respecting a criterion based on the analysis of the degeneracy conditions of the dynamic model [8, 9]. This last approach is based on the following considerations.

As shown in [2, 8, 9], the inverse dynamic model, which allows the computation of the motor input efforts, depends on the Jacobian matrix. In Type 2 singularity, the determinant of the inverse Jacobian matrix tends to zero [1] (and thus, the Jacobian matrix cannot be computed). As a result, along a trajectory passing through a Type 2 singularity, the desired motor input efforts may tend to infinity near the singularity, leading to motion infeasibility. However, the authors of [8] were the first to provide a physical criterion for changing assembly modes by passing through the Type 2 singularities, which, mathematically speaking, can be summarized as follows: in order to avoid the dynamic model degeneracy, the trajectory must be planned so that the efforts applied by the legs, the external system and the internal dynamics to the platform are reciprocal to the null space of the matrix inverse Jacobian matrix.

Thanks to the proposed approach, it was possible to plan optimal *desired* trajectories for crossing the singularities. However, due to modeling errors, a robot will never be able to *perfectly* follow the optimal trajectory whatever

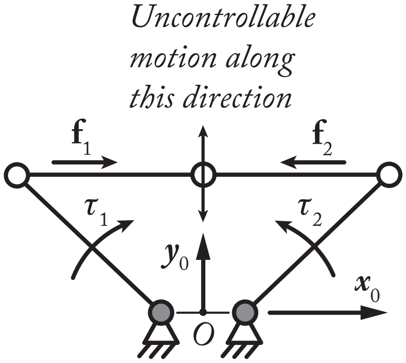


Fig. 1: A five-bar mechanism in a Type 2 singularity: in absence of any other effects, the torque τ_i of the motor i leads to the application of a force $\mathbf{f}_i$ on the end-effector. In singularity, $\mathbf{f}_1$ is collinear to $\mathbf{f}_2$. As a result, a motion is gained in the direction reciprocal to $\mathbf{f}_i$ (i.e. the end-effector is underconstrained along the direction of this uncontrollable motion) and, to produce a given force $\mathbf{f}$ along the axis $\boldsymbol{x}_0$, there is an infinity of solutions for the forces $\mathbf{f}_i$, i.e. the mechanism is overconstrained along $\boldsymbol{x}_0$.

the controller efficiency, thus potentially leading to failure of the Type 2 singularity crossing approach. In order to solve this issue, an advanced control law has been defined in [10] to finally ensure the change of assembly mode. This controller is based on the definition of a multi-model computed torque control law (e.g. see [11]) including two types of dynamic models:

1. a general dynamic model which is used for ensuring the robot trajectory tracking far from singularity,
2. a second simplified dynamic model which can never degenerate (due to the explicit deletion of the terms that may degenerate in singularity), and which is only used when the robot is close from a Type 2 singularity. This second dynamic model remain a good representation of the robot dynamics based on the fact that deleted terms stay close to zero near the singularity by planning an optimal trajectory.

The switch between the two models in the controller was made near singularity by using a sigmoid function [10].

All these results, applied either on planar [9] or spatial robots [8], showed that it was possible to successfully cross the Type 2 singularities and thus to enlarge the robot operational workspace.

Despite these rather encouraging results, some questions are still (partially or in totality) unanswered. One of them concerns the management of the local overconstraints in singularity. Indeed, in Type 2 singularities, the robot gains instantaneously an uncontrollable motion. This is due to the fact that the system of actuation wrenches on the end-effector is degenerated. As a result, the end-effector is underconstrained along the direction of the uncontrollable motion, but overconstrained along the other directions (Fig. 1). This overconstraint along the controllable directions of the motion may lead to some undesirable effects on the robot, such as buckling or instability of the end-effector [12].

To avoid the problems due to the local overconstraint, we propose in this work to modify the multi-model computed torque control law [10] so that, near singularity, the part of the control input signal leading to mechanism overconstraint is canceled.

The present paper is decomposed as follows. The next section makes necessary recalls on the parallel robot dynamic model degeneracy conditions and gives all the possible solution of the inverse dynamic model in Type 2 singularities. Then, Section 3 presents a modification of the control law [10] in order to cancel the overconstraint when crossing the singularity. In Section 4, we provide experimental results on a five-bar mechanism (DexTAR robot from Mecademic) in order to show the relevance and the robustness of the proposed advanced control law and the effect of an overconstraint when crossing the singularity. Finally, in Section 5, conclusions are drawn.

2 Recalls and solutions of the dynamic model in Type 2 singularities

2.1 Recalls on the dynamic model degeneracy conditions in Type 2 singularities

As proven in [9], the inverse dynamic model of a general parallel robots takes the following form if there is no (serial) leg singularities (out of the scope of the present analysis):

$$\boldsymbol{\tau} = \mathbf{w}_b - \mathbf{B}^T \boldsymbol{\lambda} \tag{1}$$

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

where

- $\boldsymbol{\tau}$ are the input torques / forces,
- $\boldsymbol{\lambda}$ is a vector of Lagrange multipliers,
- matrices $\mathbf{A}$ and $\mathbf{B}$ are defined though the first-order input-output kinematic constraint relation given by

  $$\mathbf{A}\dot{\mathbf{x}} + \mathbf{B}\dot{\mathbf{q}} = \mathbf{0} \tag{3}$$

  in which

  - $\mathbf{A} = \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]$, $\mathbf{B} = \left[\frac{\partial \mathbf{h}}{\partial \mathbf{q}}\right]$ where $\mathbf{h}$ is the vector of the loop-closure equations [2]; matrices $\mathbf{A}$ and $\mathbf{B}$ are square,
  - $\mathbf{q}$ is the vector of the active joint coordinates ($\dot{\mathbf{q}}$ being its derivative with respect to time),
  - $\mathbf{x}$ are the platform coordinates ($\dot{\mathbf{x}}$ being the derivative of the components of $\mathbf{x}$ with respect to time),
- $\mathbf{w}_b$ and $\mathbf{w}_p$ can be defined such that:

  $$\mathbf{w}_b = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} \tag{4}$$

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

  in which L is the Lagrangian of the system which depends explicitly on $\mathbf{q}$, $\mathbf{x}$, $\dot{\mathbf{q}}$ and $\dot{\mathbf{x}}$ (only).

It is known from [1] that Type 2 singularities appear when the matrix $\mathbf{A}$ defined in (3) is rank-deficient. Outside from Type 2 singularities, this matrix is invertible and the inverse dynamic model is given by

$$\boldsymbol{\tau} = \mathbf{w}_b + \mathbf{J}_{inv}^{-T} \mathbf{w}_p \tag{6}$$

where $\mathbf{J}_{inv} = -\mathbf{B}^{-1}\mathbf{A}$ is the inverse Jacobian matrix. Expression (6) can be obtained by finding the value of $\boldsymbol{\lambda} = \mathbf{A}^{-T}\mathbf{w}_p$ thanks to (2) and then substituting it in (1).

In Type 2 singularities, the matrix $\mathbf{A}$, and thus the inverse Jacobian matrix $\mathbf{J}_{inv}$, are rank deficient. As a result, the model (6) is no more valid. In order to avoid the dynamic model degeneracy, the following facts must be considered. First, as the matrix $\mathbf{A}$ is rank deficient[1], it exists a non-zero vector $\mathbf{t}_s$ belonging to the null-space of $\mathbf{A}$ such that:

$$\mathbf{A}\mathbf{t}_s = \mathbf{0} \Leftrightarrow \mathbf{t}_s^T \mathbf{A}^T = \mathbf{0} \tag{7}$$

$\mathbf{t}_s$ represents the direction of the uncontrollable motion inside the singularity [2]. Thus, multiplying the left-hand side of (2) by $\mathbf{t}_s^T$, we must obtain

$$\mathbf{t}_s^T \mathbf{A}^T \boldsymbol{\lambda} = 0 \tag{8}$$

[1] We consider here a loss of rank equal to 1, but a loss of rank of higher order could be considered.

As a result, in order to have consistency in the dynamic equations, by also multiplying the right-hand side of (2) by $\mathbf{t}_s^T$, the robot motion must ensure that, in the singularity, we have

$$\mathbf{t}_s^T \mathbf{w}_p = 0 \tag{9}$$

This criterion was already provided in [8, 9] and, physically speaking, it can be summarized as follows: in order to avoid the dynamic model degeneracy, the trajectory must be planned so that the efforts applied by the legs, the external system and the internal dynamics to the platform (represented by the vector $\mathbf{w}_p$) are reciprocal to the null space of the inverse Jacobian matrix (represented by the vector $\mathbf{t}_s$).

By planning a trajectory so that (9) is respected, the robot can cross a Type 2 singularity, as shown in several examples validated through experimentations [8, 9].

2.2 Solutions of the inverse dynamic model in Type 2 singularities

A question arises: even if the criterion (9) is respected, what could be the solution(s) to the inverse dynamic model given in (1) and (2)?

Mathematically speaking, if (9) is respected, it exists (at least) an exact and finite solution $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$ to the equation (2) (see Appendix A). On the contrary, if the criterion (9) is not respected, there is no solution to this equation. All the possible solutions $\boldsymbol{\lambda}^*$ are given by:

$$\boldsymbol{\lambda}^* = \mathbf{A}^{T+}\mathbf{w}_p + \left(\mathbf{I}_n - \mathbf{A}^{T+}\mathbf{A}^T\right)\boldsymbol{\xi} \tag{10}$$

where $\mathbf{A}^{T+}$ is the Moore-Penrose pseudo-inverse of $\mathbf{A}^T$, $\mathbf{I}_n$ is the $(n \times n)$ identity matrix and $\boldsymbol{\xi}$ is a vector of dimension n taking any values. This results in an infinity of possible values for $\boldsymbol{\lambda}^*$, and thus from (1), for the input efforts $\boldsymbol{\tau} = \mathbf{w}_b - \mathbf{B}^T\boldsymbol{\lambda}^*$. Physically speaking, this means that, in Type 2 singularity, the system is overconstrained. From this last expression of $\boldsymbol{\tau}$, it is quite difficult to estimate the value of the overconstraint. This is why it is preferable to rewrite the dynamic model equations (6) as follows:

$$\mathbf{J}_{inv}^T \boldsymbol{\tau} = \mathbf{J}_{inv}^T \mathbf{w}_b + \mathbf{w}_p \tag{11}$$

Expression (11) can be obtained by finding the value of $\boldsymbol{\lambda} = -\mathbf{B}^{-T}(\boldsymbol{\tau} - \mathbf{w}_b)$ thanks to (1) and then substituting it in (2). Matrix $\mathbf{B}$ is of full rank in Type 2 singularities, while matrix $\mathbf{J}_{inv}^T$ is not. If the criterion (9) is respected in Type 2 singularities, then all possible exact solutions $\boldsymbol{\tau}^*$ for the value of the motor input efforts are given by:

$$\boldsymbol{\tau}^* = \mathbf{J}_{inv}^{T+}\left(\mathbf{J}_{inv}^T \mathbf{w}_b + \mathbf{w}_p\right) + \left(\mathbf{I}_n - \mathbf{J}_{inv}^{T+}\mathbf{J}_{inv}^T\right)\boldsymbol{\eta} \tag{12}$$

where $\mathbf{J}_{inv}^{T+}$ is the Moore-Penrose pseudo-inverse of $\mathbf{J}_{inv}^T$ and $\boldsymbol{\eta}$ is a vector of dimension n taking any values. $\boldsymbol{\eta}$ is called the overconstraint [13] and $\boldsymbol{\tau}^*$ is of minimal norm when $\boldsymbol{\eta} = \mathbf{0}$, i.e. when

$$\boldsymbol{\tau}^* = \mathbf{G}\mathbf{w}_b + \mathbf{J}_{inv}^{T+}\mathbf{w}_p \tag{13}$$

with $\mathbf{G} = \mathbf{J}_{inv}^{T+}\mathbf{J}_{inv}^T$. This solution is appealing because a too high overconstraint along the controllable directions of the motion may lead to some undesirable effects on the robot, such as buckling or instability of the end-effector [12]. Therefore this solution should be considered when developing a controller for crossing Type 2 singularities. This is what we did in the next Section.

3 Design of a controller for crossing Type 2 singularities

3.1 Recalls on the previous works

As previously mentioned, since the robot will never be able to perfectly follow the optimal trajectory that should be planned in order to respect the criterion (9), an advanced control law was defined in [14] to finally ensure the change of assembly mode. This controller was based on the definition of a multi-model computed torque control law (e.g. see [11]) including two types of dynamic models:

1. a general dynamic model (6) which is used for ensuring the robot trajectory tracking far from singularity,
2. a second simplified dynamic model equal to $\boldsymbol{\tau}' = \mathbf{w}_b$ which can never degenerate (due to the explicit deletion of the terms $\mathbf{J}_{inv}^{-T}\mathbf{w}_p$ from (6)), and which is only used when the robot is close from a Type 2 singularity.

This second dynamic model remain a good representation of the robot dynamics based on the fact that the desired trajectory was planned so that the term $\mathbf{w}_p$ and its m derivatives with respect to time are zero in singularity, thus resulting in the fact that:

- $\mathbf{w}_p = \mathbf{0}$ respects the criterion (9),
- $\mathbf{w}_p$ and its m derivatives with respect to time are zero allows $\mathbf{w}_p$ to be zero not only in the singularity but also around it.

3.1.1 Computed torque control law outside of Type 2 singularities

Starting from the definitions (4) and (5) and the fact that the Lagragian depends explicitly on $\mathbf{q}$, $\mathbf{x}$, $\dot{\mathbf{q}}$ and $\dot{\mathbf{x}}$ only, we know from [15] that the terms $\mathbf{w}_b$ and $\mathbf{w}_p$ can be written under the following form:

$$\mathbf{w}_b = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} = \mathbf{M}_q(\mathbf{q},\mathbf{x})\ddot{\mathbf{q}} + \mathbf{c}_q(\dot{\mathbf{q}},\mathbf{q},\dot{\mathbf{x}},\mathbf{x}) \tag{14}$$

$$\mathbf{w}_p = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{x}}}\right) - \frac{\partial L}{\partial \mathbf{x}} = \mathbf{M}_x(\mathbf{q},\mathbf{x})\ddot{\mathbf{x}} + \mathbf{c}_x(\dot{\mathbf{q}},\mathbf{q},\dot{\mathbf{x}},\mathbf{x}) \tag{15}$$

It should be mentioned that the expressions of matrices $\mathbf{M}_q$ and $\mathbf{M}_x$ and vectors $\mathbf{c}_q$ and $\mathbf{c}_x$ do not degenerate in Type 2 singularities even if the criterion (9) is not respected [15].

By using the second-order kinematic constraint relations, obtained by differentiating (3) with respect to time and given by

$$\mathbf{A}\ddot{\mathbf{x}} + \mathbf{B}\ddot{\mathbf{q}} = \mathbf{b} \Rightarrow \ddot{\mathbf{x}} = -\mathbf{A}^{-1}\mathbf{B}\ddot{\mathbf{q}} + \mathbf{b}' = \mathbf{J}_{inv}^{-1}\ddot{\mathbf{q}} + \mathbf{b}' \tag{16}$$

where $\mathbf{b} = -\dot{\mathbf{A}}\dot{\mathbf{x}} - \dot{\mathbf{B}}\dot{\mathbf{q}}$ and $\mathbf{b}' = \mathbf{A}^{-1}\mathbf{b}$, and then introducing them into (6), we get a general dynamic model of the form:

$$\boldsymbol{\tau} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{c} \tag{17}$$

where

- $\mathbf{M} = \mathbf{M}_q + \mathbf{J}_{inv}^{-T}\mathbf{M}_x\mathbf{J}_{inv}^{-1}$ is the robot matrix of inertia
- $\mathbf{c} = \mathbf{c}_q + \mathbf{J}_{inv}^{-T}\mathbf{c}_x + \mathbf{J}_{inv}^{-T}\mathbf{M}_x\mathbf{b}'$ is the vector of Coriolis, centrifugal effects, gravity and friction efforts.

Even though this model is not linear regarding the position and the velocities of the mechanism, it is linear regarding its acceleration. Therefore, by replacing the angular acceleration $\ddot{\mathbf{q}}$ in (17) by an adapted control signal $\mathbf{u}$, the dynamics of the system is linear with respect to the control variable:

$$\boldsymbol{\tau} = \mathbf{M}\mathbf{u} + \mathbf{c} \tag{18}$$

A double integrator between the control signal and the joint variables appears and thus, only a PD control law is used to impose the control signal (it can be a PID if static friction is high) [11]:

$$\mathbf{u} = \ddot{\mathbf{q}}_d + \mathbf{K}_v(\dot{\mathbf{q}} - \dot{\mathbf{q}}_d) + \mathbf{K}_p(\mathbf{q} - \mathbf{q}_d) \Rightarrow \ddot{\mathbf{e}} + \mathbf{K}_v\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} = \mathbf{0} \tag{19}$$

where

- $\mathbf{q}_d$, $\dot{\mathbf{q}}_d$ and $\ddot{\mathbf{q}}_d$ are the desired joint position, velocity and acceleration, respectively,

- $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are the current and measured joint position, velocity and acceleration, respectively,
- $\mathbf{e} = \mathbf{q} - \mathbf{q}_d$ is the current tracking error in the joint space,
- $\mathbf{K}_v$ and $\mathbf{K}_p$ are two constant gain matrices.

This control signal is a classic second-order control law. The control input converts a complicated nonlinear controller design problem into a linear system consisting of n decoupled subsystems. Of course, this controller is based on the dynamic model of the mechanism. If this model is not accurate, the tracking error can therefore be important, but the control signal still guarantees that this tracking error respects Eq. (19) and therefore tends to zero with the desired second order dynamics. Consequently, the computed torque controller (CTC) computes the input efforts that ensure the second order dynamics on the tracking error:

$$\boldsymbol{\tau} = \mathbf{M}(\ddot{\mathbf{q}}_d + \mathbf{K}_v(\dot{\mathbf{q}} - \dot{\mathbf{q}}_d) + \mathbf{K}_p(\mathbf{q} - \mathbf{q}_d)) + \mathbf{c} \tag{20}$$

3.1.2 Computed torque control law around Type 2 singularities

As previously mentioned, thanks to an optimal trajectory planning around singularity, we impose the fact that $\mathbf{w}_p = \mathbf{0}$ (and more specifically, that $\mathbf{M}_x\ddot{\mathbf{x}} = \mathbf{0}$ and $\mathbf{c}_x = \mathbf{0}$). As a result, the dynamic model is simplified into the new model $\boldsymbol{\tau}' = \mathbf{w}_b$ which can not degenerate in Type 2 singularities even if the criterion (9) is not respected [15] and which takes the form, from (14),

$$\boldsymbol{\tau}' = \mathbf{w}_b = \mathbf{M}_q\ddot{\mathbf{q}} + \mathbf{c}_q \tag{21}$$

Therefore, by replacing the angular acceleration $\ddot{\mathbf{q}}$ in (21) by an adapted control signal $\mathbf{u}$, the dynamics of the system is linear with respect to the control variable:

$$\boldsymbol{\tau}' = \mathbf{M}_q\mathbf{u} + \mathbf{c}_q \tag{22}$$

Then, as previously, imposing a PD control law (19) to the control signal $\mathbf{u}$ leads to

$$\boldsymbol{\tau}' = \mathbf{M}_q(\ddot{\mathbf{q}}_d + \mathbf{K}_v(\dot{\mathbf{q}} - \dot{\mathbf{q}}_d) + \mathbf{K}_p(\mathbf{q} - \mathbf{q}_d)) + \mathbf{c}_q \tag{23}$$

3.1.3 Multi-model control law

Finally, the multi-model control law defined in Fig. 2 is used. In this scheme, we switch from the general model $\boldsymbol{\tau} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{c}$ (used far from singularities) to the simplified model $\boldsymbol{\tau}' = \mathbf{M}_q\ddot{\mathbf{q}} + \mathbf{c}_q$ (used in the neighborhood of singularities).

The switch between the two models in the controller is made near singularity by using a sigmoid function σ which depends on a criterion δ characterizing the proximity of singularity, as shown in [9, 14]. In what follows, it is decided that, if ε is a threshold (positive) for the metrics δ characterizing the proximity to singularity, and that $|\delta| < \varepsilon$ characterizes the presence of singularity. The control input sent to the robot is finally:

$$\boldsymbol{\tau}^{\mathrm{f}} = (1 - \sigma)\boldsymbol{\tau} + \sigma\boldsymbol{\tau}' \tag{24}$$

where σ is a piecewise continuous function defined such that:

- $\sigma = 0$ “far” from singularity ($|\delta| \geq \alpha\varepsilon$, where α is a positive number chosen by the user)
- $\sigma = 1$ “near” singularity ($|\delta| < \varepsilon$)
- $\sigma \in [0\,1]$ in between ($\varepsilon \leq |\delta| < \alpha\varepsilon$), where σ is a smooth monotonic increasing function.

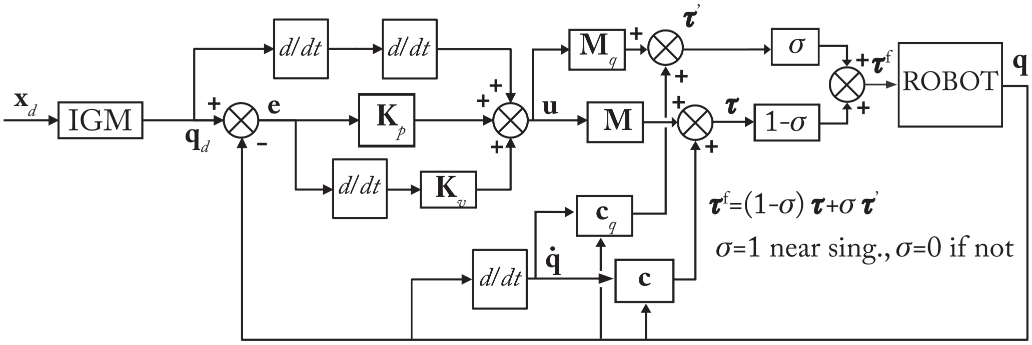


Fig. 2: The multi-model control scheme able to allow the singularity crossing. Far from singularity, $\sigma = 0$, while if the robot is close to a singularity, $\sigma = 1$. The change of the values in σ is not discontinuous and there is a transition phase during which $0 < \sigma < 1$.

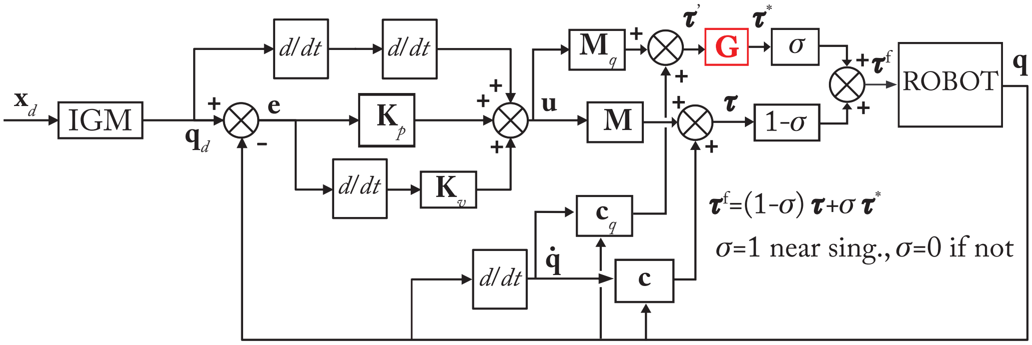


Fig. 3: The modified multi-model control scheme able to suppress the overconstraint during the singularity crossing.

3.2 Modification of the controller for canceling the overconstraint

The solution of the dynamic model that was chosen in the previous work was $\boldsymbol{\tau}' = \mathbf{w}_b$ (for $\mathbf{w}_p = \mathbf{0}$) which is not necessarily the solution leading to zero overconstraint in the system. From (13), this solution should be

$$\boldsymbol{\tau}^* = \mathbf{G}\mathbf{w}_b = \mathbf{G}(\mathbf{M}_q\ddot{\mathbf{q}} + \mathbf{c}_q) = \mathbf{G}\boldsymbol{\tau}' \tag{25}$$

$\mathbf{w}_p$ being considered null. In order to impose this solution in the controller, we multiply the signal τ' by the matrix $\mathbf{G}$ (Fig. 3), so that the control signal when we are close from singularity becomes:

$$\boldsymbol{\tau}^* = \mathbf{G}(\mathbf{M}_q\mathbf{u} + \mathbf{c}_q) = \mathbf{G}(\mathbf{M}_q(\ddot{\mathbf{q}}_d + \mathbf{K}_v(\dot{\mathbf{q}} - \dot{\mathbf{q}}_d) + \mathbf{K}_p(\mathbf{q} - \mathbf{q}_d)) + \mathbf{c}_q) \tag{26}$$

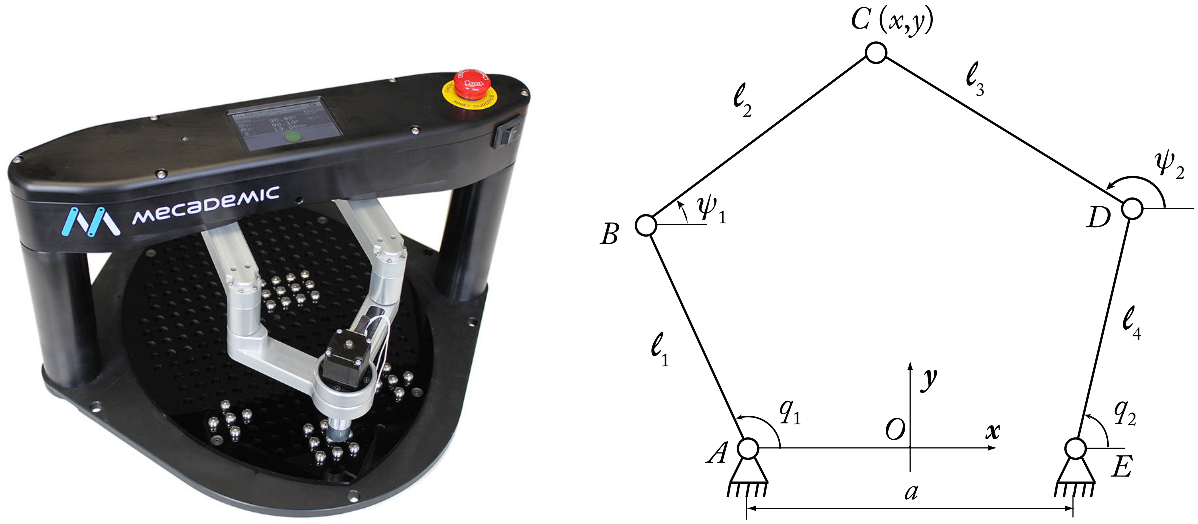
The control input sent to the robot is finally:

$$\boldsymbol{\tau}^{\mathrm{f}} = (1 - \sigma)\boldsymbol{\tau} + \sigma\boldsymbol{\tau}^* \tag{27}$$

where the function σ was defined in the previous section.

3.3 Note on the computation of the term G near singularities

We recall from (13) that $\mathbf{G} = \mathbf{J}_{inv}^{T+}\mathbf{J}_{inv}^{T}$. $\mathbf{J}_{inv}^{T+}$ is the Moore-Penrose pseudo-inverse of $\mathbf{J}_{inv}^{T}$ and it can be computed through SVD by the suppression or modification of the smallest singular value of $\mathbf{J}_{inv}^{T}$.



(a) Prototype of DexTAR

(b) Kinematic architecture

Fig. 4: The five-bar mechanism under study.

In order to avoid computational issues due to imperfect trajectory tracking, it is preferable to ensure the cancellation of the smallest singular value of $\mathbf{J}_{inv}^T$ when $\sigma = 1$, while it is not necessary to suppress it when $\sigma = 0$. During the transition phase, in order to avoid discontinuities, it is better to use a transition function so that, if s is the modified value of the smallest singular value of $\mathbf{J}_{inv}^T$, and s_0 is the value when $|\delta| = \alpha\varepsilon$, we have

- $s = s_0$ "far" from singularity ($|\delta| \geq \alpha\varepsilon$, where α is a positive number chosen by the user)
- $s = 0$ "near" singularity ($|\delta| < \varepsilon$)
- $s \in [0\ s_0]$ in between ($\varepsilon \leq |\delta| < \alpha\varepsilon$), where s is a smooth monotonic increasing function.

4 Case study

In this section, we provide experimental results on a five-bar mechanism (DexTAR robot from Mecademic) in order to show the relevance and the robustness of the proposed advanced control law and the effect of an overconstraint when crossing the singularity. First, we presents the benchmark, and then show the experimental results.

4.1 Benchmark

The DexTAR robot (Fig. 4(a)) is a five-bar mechanism (Fig. 4(b)), i.e. a planar parallel robot composed of two actuators located at the revolute joints positioned at points A and E and 3 passive revolute joints at points B, C and D. The prototype of the five-bar was designed such that $\ell_i = 90$ mm ($i = 1,\ldots,4$) and $a = 118$ mm.

To control the robot, the motors amplifiers receive a command proportional to the torques to deliver by the actuator. To send this control command, a dSPACE 1104 control board was used. The controller sampling period is 1 ms. The multi-model computed torque controller presented in the present paper was implemented on the robot. To design the robot controller, the software Matlab/Simulink combined with the software dSPACE ControlDesk has been used.

For implementing the multi-model computed torque controller, the dynamic model of the robot was necessary. It was identified by using the procedure shown in [16] and it takes the following form:

$$\boldsymbol{\tau} = \begin{bmatrix} zz_{1R} & 0 \\ 0 & zz_{2R} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} fs_1 \mathrm{sign}(\dot{q}_1) \\ fs_2 \mathrm{sign}(\dot{q}_2) \end{bmatrix} + m_R \mathbf{J}_{inv}^{-T} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} \tag{28}$$

where

- $zz_{1R} = 0.0137$ kgm$^2 \pm 0.51$ % and $zz_{2R} = 0.0145$ kgm$^2 \pm 0.54$ % are grouped moments of inertia around $\boldsymbol{z}$,
- $fs_1 = 0.30$ Nm ± 1.43 % and $fs_2 = 0.37$ Nm ± 1.34 % are Coulomb friction terms in the actuated joints,
- $m_R = 0.51$ kg ± 1.26 % is a grouped mass on the end-effector,
- $\mathbf{J}_{inv}$ is the inverse Jacobian matrix,
- $\text{sign}(u)$ is the sign function ($\text{sign}(u) = +1$ if $u > 0$, $\text{sign}(u) = -1$ if $u < 0$, and $\text{sign}(u) = 0$ if $u = 0$),
- x and y are the Cartesian coordinates of the end-effector along the axes $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively,
- q_1 is the joint angle of motor located at point A while q_2 is the joint angle of motor located at point E.

The terms zz_{iR}, fs_i and m_R are the essential parameters [17], i.e. the parameters that are well-identified because they represent the large majority of the robot dynamics effects. As a result, the model (28) is able to predict the input torques of the DexTAR with an error lower than 10 %.

It should be mentioned that the computation of the platform accelerations and robot Jacobian matrix are detailed in [9].

From (17), the model (28) can be written under the following matrix form:

$$\boldsymbol{\tau} = \mathbf{M} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \mathbf{c} \tag{29}$$

where

- $\mathbf{M} = \mathbf{M}_q + \mathbf{J}_{inv}^{-T} \mathbf{M}_x \mathbf{J}_{inv}^{-1}$, with $\mathbf{M}_q = \begin{bmatrix} zz_{1R} & 0 \\ 0 & zz_{2R} \end{bmatrix}$ and $\mathbf{M}_x = m_R \mathbf{I}_2$
- $\mathbf{c} = \mathbf{c}_q + \mathbf{J}_{inv}^{-T} \mathbf{c}_x + \mathbf{J}_{inv}^{-T} \mathbf{M}_x \mathbf{b}'$, where $\mathbf{c}_q = \begin{bmatrix} fs_1 \text{sign}(\dot{q}_1) \\ fs_2 \text{sign}(\dot{q}_2) \end{bmatrix}$, $\mathbf{c}_x = \mathbf{0}$, and $\mathbf{b}'$ is a bi-dimensional vector that can be obtained from the second-order kinematics (16).

By identification with the simplified dynamic model given in (22), the simplified dynamics of the five-bar that will be used in the controller near singularities is given by:

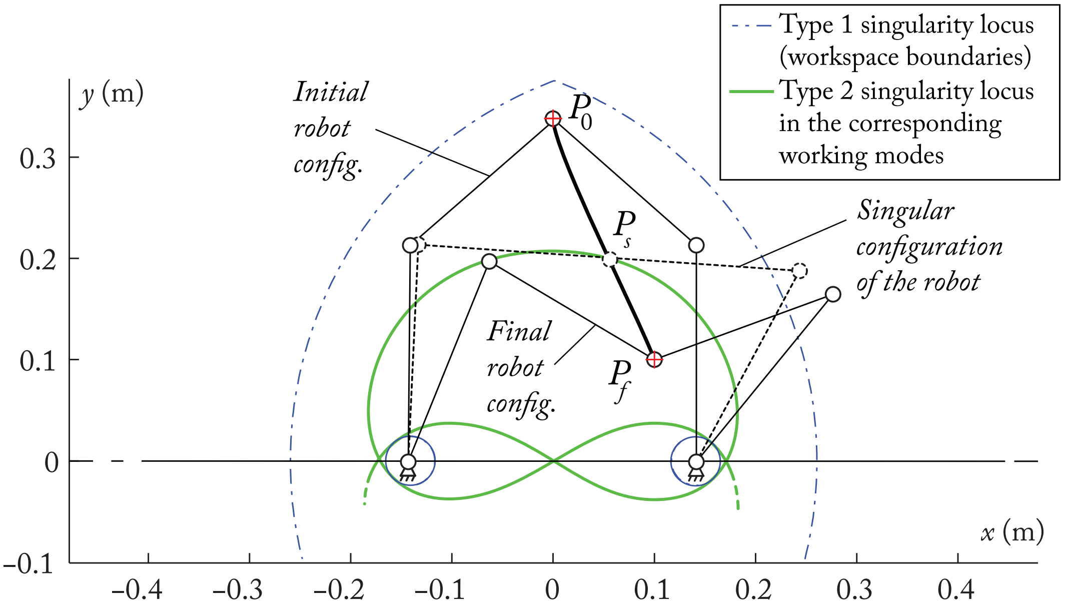
$$\boldsymbol{\tau}^* = \mathbf{G}(\mathbf{M}_q \ddot{\mathbf{q}} + \mathbf{c}_q) = \mathbf{G} \left(\begin{bmatrix} zz_{1R} & 0 \\ 0 & zz_{2R} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} fs_1 \text{sign}(\dot{q}_1) \\ fs_2 \text{sign}(\dot{q}_2) \end{bmatrix} \right) \tag{30}$$

Let us recall that $\mathbf{G} = \mathbf{J}_{inv}^{T+} \mathbf{J}_{inv}^{T}$, in which $\mathbf{J}_{inv}^{T}$ is rank deficient near singularity (see the section 3.3 for the computation of $\mathbf{J}_{inv}^{T+}$).

4.2 Experimental results

Based on the criterion $\mathbf{w}_p = \mathbf{0} \Rightarrow \ddot{x} = \ddot{y} = 0$ (which respects the criterion (9) and allows to use the control approach developed in Section 3) and the cancellation of its first and second derivatives with respect to time, it was possible to generate desired trajectory which avoids the degeneracy of the dynamic model in and around Type 2 singularities.

We planned a trajectory through singularity (Fig. 5) where the initial point is $P_0 = [0 0.15\ m]^T$, the final point is $P_f = [0 0.02\ m]^T$ and the point at which we decide to cross the singularity is $P_s = [0 0.084\ m]^T$. Once point P_f is attained, the robot do the reverse path. The duration for the full trajectory is of 0.8 sec while the robot must cross the singularity at $t = 0.3$ sec and $t = 0.5$ sec.



Type 1 singularity locus (workspace boundaries)
Type 2 singularity locus in the corresponding working modes
y (m)
x (m)
Initial robot config.
Singular configuration of the robot
Final robot config.
P_0
P_s
P_f
0.3
0.2
0.1
0
−0.1
−0.4
−0.3
−0.2
−0.1
0
0.1
0.2
0.3
0.4

Fig. 5: Starting point P_0 and ending point P_f of the Type 2 singularity crossing trajectory

Results along this trajectory, when the controller shown in Fig. 3 is implemented, are presented in Fig. 6. They showed that the robots was able to cross the singularity without any problem.[2]

Then, we decided to analyze the effect of a too high unmanaged overconstraint when crossing a singularity. For that, instead of using the solution (13) of the dynamic model inside the singularity, we chose a more general solution (12) where the overconstraint term $\boldsymbol{\eta}$ is different from zero and is equal $\boldsymbol{\eta} = [-2\ 2]^T$. Results along the same trajectory are presented in Fig. 7. Around $t = 0.3$ sec, the input efforts are saturated (saturation fixed at ±8 Nm) and the robot is no more able to follow the trajectory. This results in growing tracking errors (up to 15 deg) as the robots was indeed blocked into the Type 2 singularity.

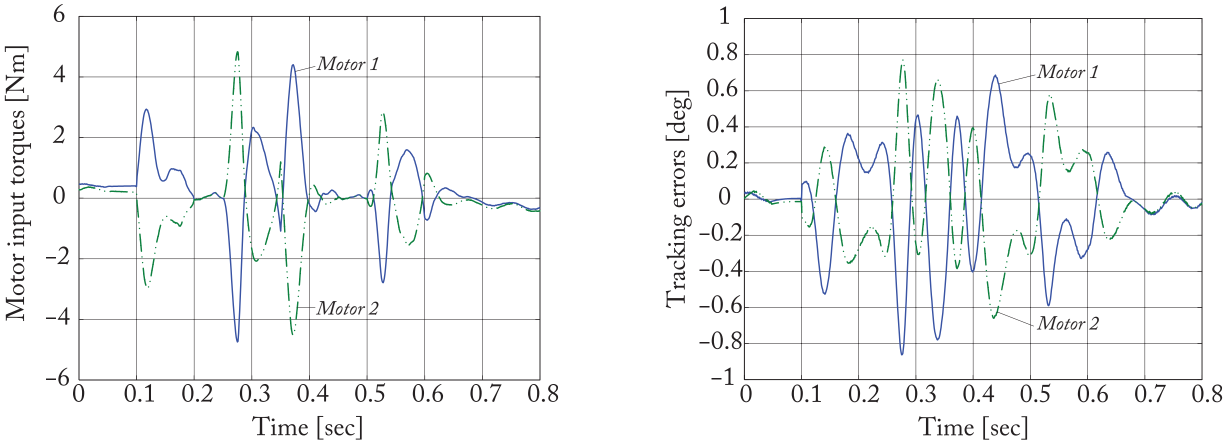
5 Conclusion

Increasing the size of the operational workspace of parallel robots is a keypoint in order to favor their industrial development. A promising solution consists in crossing Type 2 singularities in order to access other workspace aspects. This can be done by the definition of optimal trajectories that ensure the non-degeneracy of the dynamic model in the Type 2 (or parallel) singularity. However, this assumes that the control law can perfectly track the desired trajectory, which is impossible due to modeling errors.

In their previous works, two of the authors of the present paper defined a robust multi-model approach allowing parallel robots to cross Type 2 singularities. The main idea was to shift near singularities the general dynamic model to a simplified dynamic model that can never degenerate. However, because in Type 2 singularity, the robot is locally overconstrained, we showed in the present paper that the simplified dynamic model admits infinity of exact solutions, among which it is necessary to select one. We proposed to use the solution leading to a null overconstraint in order to avoid undesirable effects near singularities, such as buckling or unstability.

Accordingly, the mentioned multi-model control approach was modified and it was then applied on a prototype of five-bar mechanism. We made two types of experiments: (i) tracking a trajectory by using the new controller where the overconstraint is cancelled, and (ii) tracking a trajectory when the overconstraint is too high. In the first case, the robot was able to cross the singularity without any problem while in the second case, the robot stay blocked in it, which showed the efficiency of the proposed approach.

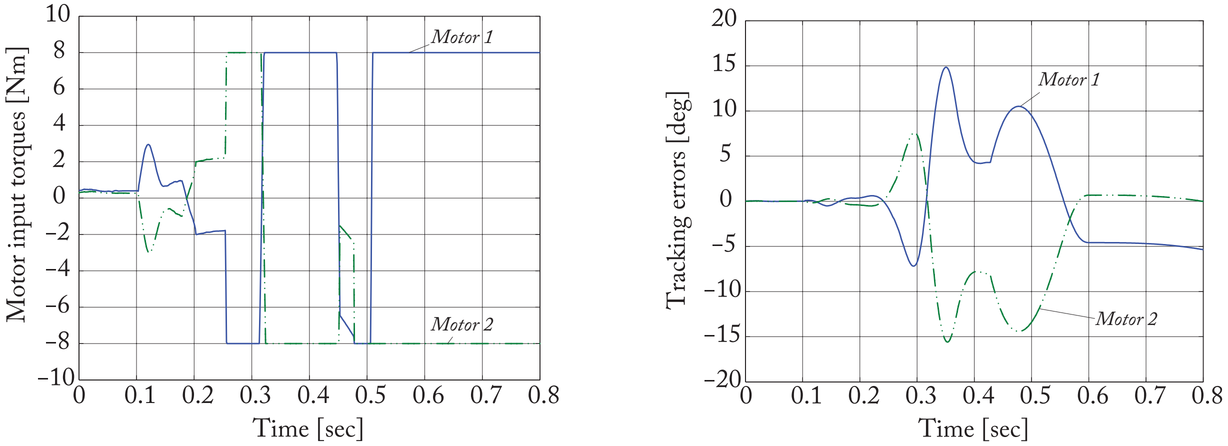
[2] The video of the DexTAR moving along this trajectory can be downloaded in: `https://youtu.be/UR9b5ouFy0o`.



(a) Input torques (joint space)

(b) Tracking errors (joint space)

Fig. 6: Results in terms of input torques and tracking errors when using the controller of Fig. 3.



(a) Input torques (joint space)

(b) Tracking errors (joint space)

Fig. 7: Results in terms of input torques and tracking errors when the overconstraint is too high.

Acknowledgements

This work was supported by the French ANR project ARROW (ANR-2011BS3-006-01).

A Appendix

Let us consider that the robot is in Type 2 singularity. As a result, the matrix $\mathbf{A}$ of (3) is rank deficient and we can define non-zero *unit* vector $\mathbf{t}_s$ belonging to the null-space of $\mathbf{A}$ [3].

The dynamics equation (2), which is recalled here for reasons of clarity,

$$\mathbf{A}^T \boldsymbol{\lambda} = \mathbf{w}_p \tag{31}$$

are expressed in the Cartesian frame (because the term $\mathbf{w}_p$ is the derivative of the Lagrangian with respect to $\mathbf{x}$ and $\dot{\mathbf{x}}$) but they can also be projected in another basis $\mathscr{B}$.

[3] We consider once again a loss of rank equal to 1, but a loss of rank of higher order could be considered.

So, let us project this equation in another basis $\mathscr{B}$ which contains as a principal vector the unit vector $\mathbf{t}_s$. As $\mathbf{A}$ is of dimension $(n \times n)$, it is possible to define $n-1$ vectors $\mathbf{t}_s^{\perp j}$ such that

- $\|\mathbf{t}_s^{\perp j}\| = 1$
- $\mathbf{t}_s^T \mathbf{t}_s^{\perp j} = 0$
- $(\mathbf{t}_s^{\perp k})^T \mathbf{t}_s^{\perp j} = 0$ for any $k \neq j$ $(j,k = 1,...,n-1)$.

Let us group all vector $\mathbf{t}_s^{\perp j}$ in a matrix $\mathbf{T}_s^{\perp}$ such that:

$$\mathbf{T}_s^{\perp} = \begin{bmatrix} \mathbf{t}_s^{\perp 1} & \dots & \mathbf{t}_s^{\perp n-1} \end{bmatrix} \tag{32}$$

Passing from the Cartesian frame to the basis $\mathscr{B}$ can be done by using the transformation $\mathbf{R}_{\mathscr{B}}$ defined such that

$$\mathbf{R}_{\mathscr{B}} = \begin{bmatrix} \mathbf{T}_s^{\perp} & \mathbf{t}_s \end{bmatrix} \tag{33}$$

$\mathbf{R}_{\mathscr{B}}$ is an orthogonal matrix: as a result $\mathbf{R}_{\mathscr{B}}^{-1} = \mathbf{R}_{\mathscr{B}}^T$ and $\det(\mathbf{R}_{\mathscr{B}}) = \pm 1$.

Thus, if we project the equation (31) in $\mathscr{B}$, we have

$$\mathbf{R}_{\mathscr{B}}^T \mathbf{A}^T \boldsymbol{\lambda} = \mathbf{R}_{\mathscr{B}}^T \mathbf{w}_p \Rightarrow \begin{bmatrix} (\mathbf{T}_s^{\perp})^T \mathbf{A}^T \\ \mathbf{t}_s^T \mathbf{A}^T \end{bmatrix} \boldsymbol{\lambda} = \begin{bmatrix} (\mathbf{T}_s^{\perp})^T \mathbf{w}_p \\ \mathbf{t}_s^T \mathbf{w}_p \end{bmatrix} \tag{34}$$

If the criterion (9) is respected, and as $\mathbf{t}_s$ is in the null-space of $\mathbf{A}$, the equation (34) can be rewritten as

$$\begin{bmatrix} (\mathbf{T}_s^{\perp})^T \mathbf{A}^T \\ \mathbf{0}_{1\times n} \end{bmatrix} \boldsymbol{\lambda} = \begin{bmatrix} (\mathbf{T}_s^{\perp})^T \mathbf{w}_p \\ 0 \end{bmatrix} \Leftrightarrow (\mathbf{T}_s^{\perp})^T \mathbf{A}^T \boldsymbol{\lambda} = (\mathbf{T}_s^{\perp})^T \mathbf{w}_p \tag{35}$$

$\boldsymbol{\lambda}$ being a vector with n independent variables, and (35) containing $n-1$ independent equations, there is an infinity of exact solutions to the equation (35), and as a result, to the equation (31).

If the criterion (9) is not respected, i.e. $\mathbf{t}_s^T \mathbf{w}_p \neq 0$, the equation (34) can be rewritten as

$$\begin{bmatrix} (\mathbf{T}_s^{\perp})^T \mathbf{A}^T \\ \mathbf{0}_{1\times n} \end{bmatrix} \boldsymbol{\lambda} = \begin{bmatrix} (\mathbf{T}_s^{\perp})^T \mathbf{w}_p \\ \mathbf{t}_s^T \mathbf{w}_p (\neq 0) \end{bmatrix} \tag{36}$$

Here, the dynamic equations are no more consistent, and there is no exact solutions to the equation (36). However, we can find an approximate solution $\boldsymbol{\lambda}^a$ which minimizes the norm of the error $\mathbf{e} = \mathbf{A}^T \boldsymbol{\lambda} - \mathbf{w}_p$.

References

[1] C. Gosselin and J. Angeles, "Singularity analysis of closed-loop kinematic chains," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 281–290, 1990.

[2] J. Merlet, *Parallel Robots*. Springer, 2nd ed., 2006.

[3] X. Kong and C. Gosselin, "A class of 3-dof translational parallel manipulators with linear input-output equations," in *Proceedings of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, (Québec City, QC, Canada), pp. 3–4, October 2002.

[4] A. Muller, "Internal preload control of redundantly actuated parallel manipulators - its application to backlash avoiding control," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 668–677, 2005.

[5] J. Kotlarski, D. Trung, B. Heimann, and T. Ortmaier, "Optimization strategies for additional actuators of kinematically redundant parallel kinematic machines," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2010)*, pp. 656–661, May 2010.

[6] F. Bourbonais, P. Bigras, and I. Bonev, "Minimum-time trajectory planning and control of a reconfigurable pick-and-place parallel robot," *IEEE/ASME Transactions on Mechatronics*, 2014.

[7] M. Zein, P. Wenger, and D. Chablat, "Non-singular assembly-mode changing motions for 3-RPR parallel manipulators," *Mechanism and Machine Theory*, vol. 43, no. 4, pp. 480–490, 2008.

[8] S. Briot and V. Arakelian, "Optimal force generation of parallel manipulators for passing through the singular positions," *International Journal of Robotics Research*, vol. 27, no. 8, pp. 967–983, 2008.

[9] S. Briot, G. Pagis, N. Bouton, and P. Martinet, "Degeneracy conditions of the dynamic model of parallel robots," *Multibody System Dynamics*, 2016.

[10] G. Pagis, N. Bouton, S. Briot, and P. Martinet, "Enlarging parallel robot workspace through Type-2 singularity crossing," *ontrol Engineering Practice*, vol. 39, pp. 1–11, 2015.

[11] W. Khalil and E. Dombre, *Modeling, Identification and Control of Robots.* Hermes Penton London, 2002.

[12] A. Pashkevich, A. Klimchik, and D. Chablat, "Enhanced stiffness modeling of manipulators with passive joints," *Mechanism and Machine Theory*, vol. 46, no. 5, pp. 662–679, 2011.

[13] A. Muller, "Internal prestress control of redundantly actuated parallel manipulators its application to backlash avoiding control," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 668–677, 2005.

[14] G. Pagis, N. Bouton, S. Briot, and P. Martinet, "Design of a controller for enlarging parallel robots workspace through Type 2 singularity crossing," in *Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA 2014)*, (Hong Kong, China), may 2014.

[15] S. Briot and W. Khalil, *Dynamics of Parallel Robots: From Rigid Bodies to Flexible Elements*. Springer, 2015.

[16] S. Briot and M. Gautier, "Global identification of joint drive gains and dynamic parameters of parallel robots," *Multibody System Dynamics*, 2014. in press.

[17] W. Khalil, M. Gautier, and P. Lemoine, "Identification of the payload inertial parameters of industrial manipulators," in *Proceedings IEEE ICRA*, (Roma, Italy), pp. 4943–4948, April 2007.