

# Motion Discontinuity-Robust Controller for Steerable Mobile Robots

Mohamed Sorour, Andrea Cherubini, Philippe Fraisse, Robin Passama

## ► To cite this version:

Mohamed Sorour, Andrea Cherubini, Philippe Fraisse, Robin Passama. Motion Discontinuity-Robust Controller for Steerable Mobile Robots. IEEE Robotics and Automation Letters, IEEE 2017, 2 (2), pp.452-459. <10.1109/LRA.2016.2638466>. <hal-01489043>

**HAL Id: hal-01489043**

**<https://hal.archives-ouvertes.fr/hal-01489043>**

Submitted on 16 Mar 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Motion Discontinuity-Robust Controller for Steerable Mobile Robots

Mohamed Sorour<sup>1,2</sup>, Andrea Cherubini<sup>1</sup>, Philippe Fraisse<sup>1</sup>, and Robin Passama<sup>1</sup>

**Abstract**—Steerable wheeled mobile robots (SWMR) are able to perform arbitrary 3D planar trajectories, only after initializing the steer joint vector to the proper values. These robots employ fully steerable conventional wheels. Hence, they have higher load carrying capacity than their holonomic counterparts, and as such are preferable for industrial applications. Industrial setups nowadays are being prepared for the emerging field of human-robot collaboration/cooperation. Such field is highly dynamic, due to fast moving human workers, sharing the operation space. This imposes the need for human safe trajectory generators, that can lead to frequent halts in motion, re-planning, and to sudden, discontinuous changes in the position of the robot's instantaneous center of rotation (ICR). Indeed, this requires steer joint reconfiguration to the newly computed trajectory. This issue is almost ignored in the literature, and motivates this work. The authors propose a new ICR-based kinematic controller, that is capable of handling discontinuity in commanded velocity, while respecting the maximum joint performance limit. This is done by formulating a quadratic optimization problem with linear constraints in the 2D ICR space. The controller is also robust against representation and kinematic singularities. It has been tested successfully on the Neobotix-MPO700 industrial mobile robot.

**Index Terms**—Steerable Wheeled Mobile Robots, Pseudo-omni Mobile Robots, Nonholonomic Omnidirectional Wheeled Mobile Robots, Kinematic Control.

## I. INTRODUCTION

Steerable wheeled mobile robots gain mobility by employing fully steerable wheels, having two active joints, one for steering, and another for driving. Despite having only one degree of mobility (DOM) (defined here as the instantaneously accessible degrees of freedom DOF), corresponding to the rotation about the instantaneous center of rotation (ICR), such robots can perform complex 3D planar trajectories. This can be done only after orienting the wheels to the proper angles, depending on the trajectory to be performed. Fully omnidirectional robots, on the other hand, employing either castor, Swedish or omni wheels, have 3DOM and, as such, can instantaneously track any desired trajectory. However, this type of wheels has limited load carrying capacity, and may cause vibration during robot motion. For these reasons, SWMR are the preferred choice for industrial setups.

Due to their wide use in the industry, many research efforts have been made to enhance their performance, against kinematic (ICR at the steering joint axis) and representation (from the mathematical model) singularities. The latter have been solved, in the case of SWMR with three or more wheels, by using the 3D Cartesian velocity in deriving the kinematic model [1]. The former was addressed as a constraint to robot motion in [2], or by using repulsive potential fields in [3], [4], [5]. However, reduction in robot maneuverability, by

employing these methods, is not acceptable, and attempts were made to deal with kinematic singularity by handling the maximum joint limits [6] and by changing coordinate system [7]. Steering mechanical limits were also studied in [5] and [8]. Simulations in [7] do not indicate the behavior of the steer joints in terms of respecting the velocity limits at kinematic singularity, while in [6], it is shown that the steer rate will saturate and keep operating at the maximum limit.

On the other hand, to the best of the authors knowledge, no thorough investigation has been conducted on the issue of reorienting the wheels, once discontinuity in the robot trajectory occurs. Usually, steer reconfiguration is performed in a *manual* fashion, depending on the test trajectory. This is found in the literature under various designations, e.g. "initial phase" in [1], and "open-loop starting procedure" in [2]. Although an ICR-based controller is the most suited to handle such cases, the work in [6] and [9] is limited by the assumption of continuous and differentiable desired signal, whereas in [5] the singularities imposed by the ICR-based model are handled by reducing the robot maneuverability.

Such situation - discontinuity in robot motion - is more likely to happen nowadays, in the emerging field of human-robot collaboration. Mobile robots working in the vicinity of fast moving human workers, will usually encounter discontinuity in the online computed trajectory. In case of static obstacles, the online planner can be less restricted in the form of the output trajectory, so that smooth behavior can always be expected. Instead, here, sudden appearance of an active worker can result in prohibiting motion, and re-routing to follow other trajectories. In such cases, the state of the art will fail to provide proper solutions.

In this work, we propose a kinematic control framework that is:

- 1) robust against trajectory discontinuity,
- 2) capable of handling kinematic singularity,
- 3) compliant with the maximum steer joint limits in terms of velocity and acceleration (or jerk, seamlessly).

The framework consists of two decoupled kinematic controllers: a Cartesian-velocity based controller, and an ICR-based one. The former is used to control the drive rate "wheel speed", employing a Cartesian space kinematic model. The latter controls the steering rate, while respecting the maximum steer joint limits, by using optimization to locate the "next sample time" ICR coordinates. The developed 3D Cartesian space kinematic model is free from representational singularity, while the kinematic singularities are being handled in the 2D ICR space controller. The benefit of using separate kinematic controllers for the drive and steering rates, is does not require mapping the 2D ICR-coordinate space to the 3D Cartesian space, hence avoids associated singularities and inconveniences. Thanks to the formulated optimization problem, discontinuity in robot velocity trajectory is handled, while respecting the steer joint limits.

<sup>1</sup>Interactive Digital Human group IDH, Laboratory for Computer Science, Micro-electronics and Robotics LIRMM, University of Montpellier CNRS, 860 rue Saint Priest, 34090 Montpellier, France. [firstname.lastname@lirimm.fr](mailto:firstname.lastname@lirimm.fr)

<sup>2</sup>PSA Peugeot-Citroen, Velizy Villacoublay, France.

In the following, section II presents the Cartesian space kinematic model. The velocity discontinuity robust controller is detailed in section III. Experiments are depicted in section IV. Conclusions are finally given in section V.

## II. KINEMATIC MODEL

The kinematic model presented in this section (detailed in previous work by the authors [10]) is inspired by the pioneer work of Muir [11], Campion [12], Betourne [13] and Low *et al.* [14]. The schematic of a SWMR is shown in Fig. 1 for a 4 wheeled robot. However, the model is generic for SWMR with number of wheels  $N \geq 3$ .

### A. Cartesian Space 3D Model Formulation

Let  $\mathcal{F}_I = (o_I, \mathbf{x}_I, \mathbf{y}_I, \mathbf{z}_I)$  be the inertial frame,  $\mathcal{F}_b = (o_b, \mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b)$  the mobile base frame, with origin  $o_b$  located at the base geometric center,  $\mathcal{F}_{hi} = (o_{hi}, \mathbf{x}_{hi}, \mathbf{y}_{hi}, \mathbf{z}_{hi})$  the  $i^{th}$  hip frame ( $i = 1, \dots, N$ ), attached to the fixed part of the steering joint, and related to the base frame by a fixed transformation matrix, and  $\mathcal{F}_{si} = (o_{si}, \mathbf{x}_{si}, \mathbf{y}_{si}, \mathbf{z}_{si})$  the steering frame, attached to the movable part. The hip and steering frames share the same origin, with relative orientation  $\beta_i$  (the steering angle). Frame  $\mathcal{F}_{wi} = (o_{wi}, \mathbf{x}_{wi}, \mathbf{y}_{wi}, \mathbf{z}_{wi})$  is attached to (but not rotating with) the  $i^{th}$  wheel, assigned such that  $\mathbf{x}_{wi}$  points along the heading of the wheel, which rotates about  $\mathbf{y}_{wi}$  by the driving angle  $\phi_i$ . All the frames have the  $z$  axis pointing upwards. Let the mobile base pose w.r.t. the inertial frame define the 3D task space coordinates  $\xi = [x \ y \ \theta]^T$ . A left superscript is added to indicate the frame in which the pose is expressed, for instance  $^I\xi$  and  $^b\xi$  denote the robot pose, expressed in the inertial and base frames respectively. In the sequel, the left superscript is omitted in case of vectors expressed in the base frame to lighten the notation, unless otherwise specified.

Considering the  $i^{th}$  wheel velocity  $\mathbf{v}_{ci} = [v_{ti} \ v_{ni} \ 0]^T$  at the ground contact point  $o_{ci}$  (expressed in the wheel frame  $\mathcal{F}_{wi}$ ), with  $v_{ti}$  and  $v_{ni}$  respectively the  $i^{th}$  tangential and normal velocities, it can be shown that [10]:

$$v_{ti} = \mathbf{f}(\beta_i)^b \dot{\xi} + d\dot{\beta}_i - r_w \dot{\phi}_i, \quad (1)$$

$$v_{ni} = \mathbf{g}(\beta_i)^b \dot{\xi}, \quad (2)$$

$$\mathbf{f}(\beta_i) = [c(\beta_i) \ s(\beta_i) \ d - h_{yi}c(\beta_i) + h_{xi}s(\beta_i)],$$

$$\mathbf{g}(\beta_i) = [-s(\beta_i) \ c(\beta_i) \ h_{xi}c(\beta_i) + h_{yi}s(\beta_i)],$$

where  $h_{xi} = \pm b$  and  $h_{yi} = \pm a$  denote the position of the  $i^{th}$  hip frame origin  $o_{hi}$  in the base frame,  $c(*)$  and  $s(*)$  short hand  $\cos(*)$  and  $\sin(*)$  respectively. The parameters  $d$  and  $r_w$  are the steerable wheel offset and radius respectively. Setting  $v_{ti} = 0$  and  $v_{ni} = 0$ , (1) and (2) represent the *rolling with no slipping* and the *no lateral skidding* kinematic constraints, respectively. The *no skidding* constraint imposes restrictions on the robot motion (the wheel cannot move sideways), and forces the existence of a unique ICR point, around which the base frame and all wheels must rotate. From such constraint, we construct the kinematic constraint matrix  $\mathbf{G}(\beta)$  as:

$$\mathbf{G}(\beta)_{N \times 3} = [\mathbf{g}(\beta_1)^T \ \dots \ \mathbf{g}(\beta_N)^T]^T. \quad (3)$$

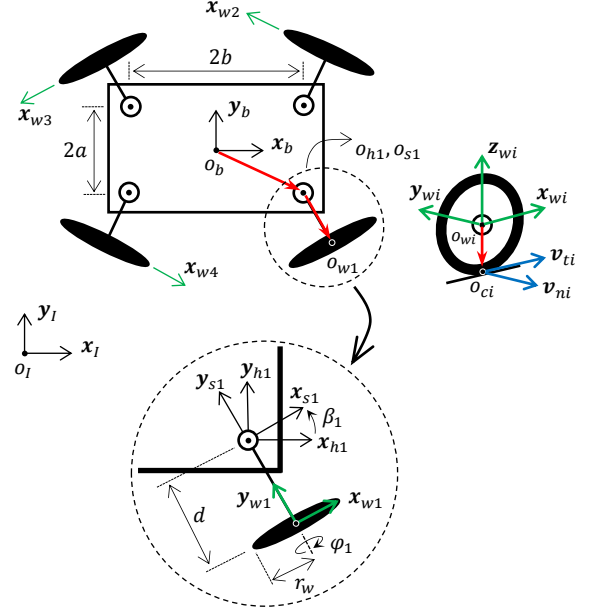


Fig. 1: Schematic model of a four wheeled SWMR.

Equation (2) is differentiated w.r.t time, rearranged to evaluate the required steer joint rate when  $v_{ni} = 0$  as:

$$\dot{\beta}_i = \frac{-\mathbf{g}(\beta_i)^b \ddot{\xi}}{\frac{d\mathbf{g}(\beta_i)^b}{d\beta_i} \dot{\xi}}. \quad (4)$$

Then, the wheel drive rate is obtained from (1), when  $v_{ti} = 0$ :

$$\dot{\phi}_i = \frac{1}{r_w} \mathbf{f}(\beta_i)^b \dot{\xi} + \frac{d}{r_w} \dot{\beta}_i. \quad (5)$$

Equations (4) and (5) represent the SWMR inverse actuation kinematic model (IAKM).

### B. Odometry Model

In order to compute the task space velocity response from the joint space velocity measurements, we need the **odometry model, or the forward actuation kinematic model (FAKM)**:

$$\dot{\xi} = \mathbf{F}_{(d)}^+(\hat{\beta})(r_w \hat{\phi} - d\hat{\beta}), \quad (6)$$

$$\mathbf{F}(\hat{\beta})_{N \times 3} = [\mathbf{f}(\hat{\beta}_1)^T \ \dots \ \mathbf{f}(\hat{\beta}_N)^T]^T,$$

where the hat symbol  $\hat{\cdot}$  indicates a measurement value and  $\mathbf{F}_{(d)}^+(\cdot)$  denotes the damped pseudo-inverse of  $\mathbf{F}(\cdot)$ , evaluated using [15] and [16]:

$$\mathbf{F}_{(d)}^+(\cdot) = (\mathbf{F}^T(\cdot)\mathbf{F}(\cdot) + \delta^2 \mathbf{I}_{3 \times 3})^{-1} \mathbf{F}^T(\cdot),$$

with  $\delta \in \mathbb{R}$  the damping factor. We use the damped pseudo-inverse to overcome algorithmic singularities, occurring when the mobile base moves with null angular velocity (all steer angles identical), as in such case,  $\mathbf{F}(\beta)$  loses rank.

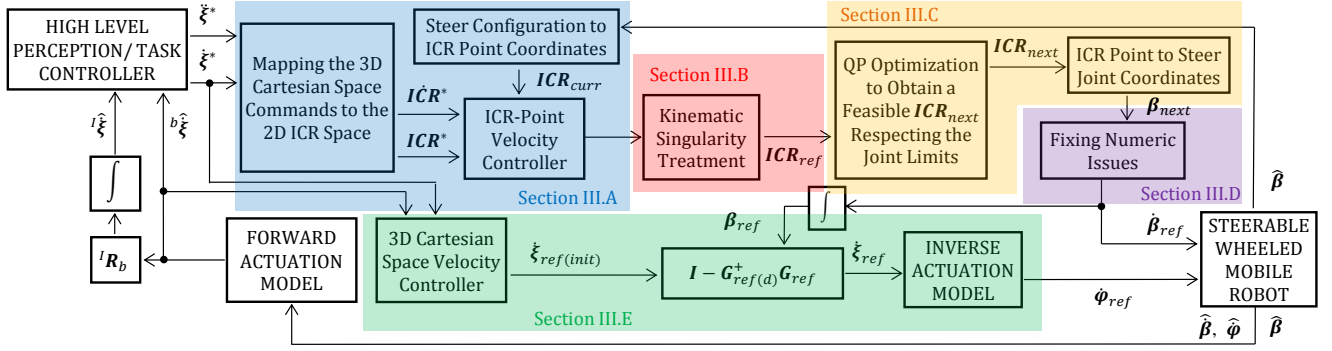


Fig. 2: Proposed control framework.

### III. DISCONTINUITY ROBUST ICR-POINT CONTROLLER

In this section, we present our control framework for injecting the steer joint initialization in the trajectory performed by SWMR, and for handling run-time trajectory discontinuities, which “depending on the application” can be often necessary, especially in human robot interaction tasks.

The whole framework is depicted in Fig. 2. The desired 3D Cartesian space robot motion ( $\dot{\xi}^*$ , and  $\ddot{\xi}^*$ ) is generated by a **high level perception controller** (out of the scope of this work). This is then mapped to the 2D ICR space, and the output desired ICR motion ( $ICR^*$ , and  $\dot{ICR}^*$ ) is fed to the ICR velocity controller, along with the current ICR coordinates  $ICR_{curr}$ . The output reference signal  $ICR_{ref}$  is then used by an optimization algorithm to decide the “next sample time” ICR coordinates  $ICR_{next}$  that will minimize the quadratic cost error:  $\|ICR_{ref} - ICR_{next}\|_2^2$  while respecting the joint performance limits formulated as linear constraints. We use the  $ICR_{ref}$  rather than  $ICR^*$  in the cost function in order to obtain smooth behavior since it is error dependant. The corresponding steer joint reference signal  $\beta_{ref}$  is then evaluated (while fixing the numeric issues involved) and differentiated, to obtain the  $\dot{\beta}_{ref}$  that is sent to the robot low level controller. At the same time, a decoupled robot twist controller is implemented, the initial output of which  $\dot{\xi}_{ref}(init)$  is projected onto the null space of the “next sample time” robot configuration (represented by the kinematic constraint matrix  $G(\beta_{ref})$ ), to obtain the feasible control signal  $\dot{\xi}_{ref}$ . The reference wheel rate  $\dot{\phi}_{ref}$  is then obtained using the IAKM in (5). The colored blocks in Fig. (2) are detailed in each of the following subsections.

#### A. ICR Velocity Controller

In Fig. 2, the desired base frame motion is the output of the high level perception block. As shown in Fig. 3, the corresponding desired ICR coordinates, expressed in base frame,  $ICR^* = [X^* \ Y^*]^T$ , are usually evaluated using:

$$X^* = -\dot{y}^*/\dot{\theta}^*, \quad Y^* = \dot{x}^*/\dot{\theta}^*.$$

However, this formula is singular in pure translation motion. Alternatively, here we propose to use:

$$\begin{aligned} X^* &= R_\infty \tanh\left(\left(\frac{-\dot{y}^*}{\dot{\theta}^* + \text{sign}(\dot{\theta}^*)\delta_1}\right)/R_\infty\right), \\ Y^* &= R_\infty \tanh\left(\left(\frac{\dot{x}^*}{\dot{\theta}^* + \text{sign}(\dot{\theta}^*)\delta_1}\right)/R_\infty\right), \end{aligned} \quad (7)$$

$$\text{sign}(\dot{\theta}^*) = \begin{cases} +1, & \forall \dot{\theta}^* \geq 0 \\ -1, & \forall \dot{\theta}^* < 0 \end{cases},$$

with  $R_\infty$  a large positive scalar representing the radius of curvature when the ICR point is very far.  $R_\infty$  should be chosen carefully, to avoid numerical instability. On the other hand,  $\delta_1$  is an infinitesimally small positive scalar value. Equation (7) is both singularity free, and provides bounded  $ICR^*$  components thanks to the  $\tanh()$  function. Similarly, the desired ICR velocity,  $\dot{ICR}^*$  can be computed using:

$$\begin{aligned} \dot{X}^* &= \dot{V}_{max} \tanh\left(\left(\frac{-\ddot{y}^*\dot{\theta}^* + \dot{y}^*\ddot{\theta}^*}{(\dot{\theta}^*)^2 + \delta_1}\right)/\dot{V}_{max}\right), \\ \dot{Y}^* &= \dot{V}_{max} \tanh\left(\left(\frac{\ddot{x}^*\dot{\theta}^* - \dot{x}^*\ddot{\theta}^*}{(\dot{\theta}^*)^2 + \delta_1}\right)/\dot{V}_{max}\right), \end{aligned} \quad (8)$$

where  $\dot{V}_{max}$  limits  $\dot{ICR}^*$ .

On the other hand, the current ICR coordinates  $ICR_{curr} = [X_{curr} \ Y_{curr}]^T$  are computed using the current “measured” steer joint configuration,  $\hat{\beta}$ . We assume that this configuration corresponds to a unique valid ICR point, i.e., that the low level robot controller properly tracks the reference steer rate  $\dot{\beta}_{ref}$ , and as such the measured steer joint values correspond to a “unique”  $ICR_{curr}$ . We use the pair of steer joints with the biggest angular difference as depicted in Fig. 3 for numeric robustness, whose indexes  $i, j \in \{1, \dots, N\}$  are given by:

$$[i, j] = \begin{cases} \max_{i,j} \left( |\hat{\beta}_i - \hat{\beta}_j| \right), & \forall |\hat{\beta}_i - \hat{\beta}_j| \leq \pi/2 \\ \max_{i,j} \left( |\hat{\beta}_i - \hat{\beta}_j| - \pi \right), & \forall |\hat{\beta}_i - \hat{\beta}_j| > \pi/2 \end{cases},$$

then,

$$\begin{aligned} Y_{curr} &= \begin{cases} R_\infty c(\hat{\beta}_i), & \forall \hat{\beta}_i < \beta_{th} \\ \frac{h_{xi} - h_{xj} + h_{yi}t(\hat{\beta}_i) - h_{yj}t(\hat{\beta}_j)}{t(\hat{\beta}_i) - t(\hat{\beta}_j)}, & \text{otherwise} \end{cases}, \\ X_{curr} &= \begin{cases} R_\infty s(\hat{\beta}_i), & \forall \hat{\beta}_i < \beta_{th} \\ h_{xi} - (Y_{curr} - h_{yi})t(\hat{\beta}_i), & \text{otherwise} \end{cases}, \end{aligned} \quad (9)$$

where  $\beta_{th}$  is a small threshold value, and  $t(*)$  short-hands the  $\tan(*)$  function. The result of (9) is saturated to  $\pm R_\infty$  to obtain  $X_{curr}$  and  $Y_{curr}$ . Finally, the *ICR-Point Velocity*

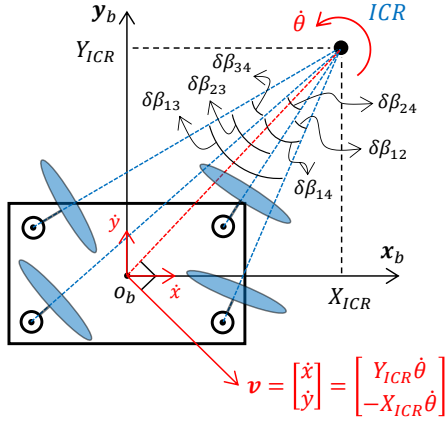


Fig. 3: Relation between the 3D Cartesian velocity space and the 2D ICR coordinate space. The steer joint pair with the biggest angular difference  $\delta\beta_{ij} = |\beta_i - \beta_j|$  is used to compute  $ICR_{curr}$ .

Controller is formulated as:

$$\begin{aligned} \dot{ICR}_{ref} &= \dot{ICR}^* + \lambda ICR_{err}, \\ ICR_{err} &= ICR^* - ICR_{curr}, \end{aligned} \quad (10)$$

where  $\lambda$  is a positive scalar proportional gain, and  $ICR_{err}$  is the error between the desired and current ICR coordinates.

### B. Kinematic Singularity Treatment

Kinematic singularity occurs whenever the ICR point approaches any of the steering axes (where  $Y_{ref} \approx h_{yi}$ , and  $X_{ref} \approx h_{xi}$ ), since evaluating the singular steer angle using:

$$\beta_i = \arctan 2(Y_{ref} - h_{yi}, X_{ref} - h_{xi}) - \pi/2,$$

will result in undefined value. In such case, the steering rate grows unbounded. This is handled in the literature either by constraining the robot velocity space [2], [3], [4], [5], so that the ICR never passes by any steering axis, or by saturating the steer rate at singularity [6]. In previous work, we developed a method to dampen such effect: the steer rate slows down, and is zeroed, as the ICR approaches and then reaches the steer axis [10]. However, it cannot be applied here, as we do steer joint control in the 2D ICR space.

Instead, here, we construct a singularity zone (circle with fixed radius  $R_{zone}$ ) around each steering axis (the 2<sup>nd</sup> steer joint zone in Fig. 4). If the ICR enters the circle,  $ICR_{ref}$  is set on the opposite side, along the straight line normal to the singular wheel. Thanks to this approach, the motion of the singular joint is minimal. The  $ICR_{ref}$  is modified as:

$$\begin{aligned} X_{ref} &= X_{curr} + 2R_{zone} \cos(\beta_s + \pi/2), \\ Y_{ref} &= Y_{curr} + 2R_{zone} \sin(\beta_s + \pi/2), \end{aligned} \quad (11)$$

with  $R_{zone}$ , the zone radius, and  $\beta_s$ , the singular steer angle.

### C. Optimization to obtain a feasible ICR

The reference control signal  $ICR_{ref}$  obtained from (10) is error driven. Whenever trajectory discontinuity occurs, or when sending the initial motion commands where the error is maximum, excessive joint velocity/acceleration is required. This corresponds to an ICR reference point that is far from the current one. In this section, a quadratic

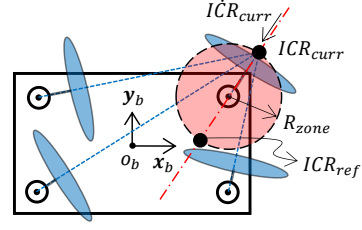


Fig. 4: ICR point approaching the singularity zone (in red) of the 2<sup>nd</sup> steer joint. ICR motion direction is indicated by the top arrow.

programming optimization is formulated to determine the best feasible "next sample time" ICR coordinates  $ICR_{next}$  among a feasible set. Such set is constructed by formulating the maximum and minimum "next sample time" steer angles as linear constraints depending on the current steer joint state and on its maximum performance (here, velocity and acceleration) limits. Then,  $ICR_{next}$  will replace  $ICR_{ref}$  in computing the steer joint commands.

Fig. 5 shows the feasible change, in one sample time period  $t_s$ , in each of the steer joints (in positive and negative directions) based on: 1. the  $ICR_{curr}$ , and 2. the maximum steer joint velocity and acceleration. Indeed, the feasible set of  $ICR_{next}$  is within the intersection of the extreme feasible changes of all steer joints. The feasible region for  $ICR_{next}$  shown in Fig. 5.a corresponds to a stationary  $ICR_{curr}$ , which can move arbitrarily in any direction within the indicated feasible set. Instead, Fig. 5.b depicts the case of moving  $ICR_{curr}$  (i.e.,  $\dot{ICR}_{curr} \neq 0$ ). In such case, the maximum steering deceleration constraints (shown in Fig. 5.b as  $\beta_{i(t+t_s)min+}$  and  $\beta_{i(t+t_s)min-}$  for minimum change in steer angle in positive and negative directions, respectively) will apply, to further restrict the feasible set, dividing it into four regions based on the current steer rate direction. Figure 5.c shows a particular case of Fig. 5.b where  $\dot{\beta}_2 > 0$  and  $\dot{\beta}_3 > 0$ . Also, the case of discontinuous change in the desired motion trajectory is depicted, where a new reference ICR point  $ICR_{ref}$  appeared while the  $ICR_{curr}$  is following an old one  $ICR_{ref(old)}$ . Thanks to this approach, discontinuity in velocity command can be handled seamlessly with the same formulation, and no steer joint reconfiguration is needed.

All cases can be addressed by formulating a quadratic optimization problem, subject to linear constraints, to minimize the error between the ICR reference signal  $ICR_{ref}$  and the best feasible ICR at the next sample time  $ICR_{next} = [X_{next} \ Y_{next}]^T$ , being also the decision variable:

$$\begin{aligned} &\text{minimize}_{ICR_{next}} \quad \|ICR_{ref} - ICR_{next}\|_2^2 \\ &\text{subject to} \quad (-1)^{q_i(min)} A_{i(min)} (ICR_{next} - h_i) \geq 0, \\ &\quad \quad \quad (-1)^{q_i(max)+1} A_{i(max)} (ICR_{next} - h_i) \leq 0, \end{aligned}$$

where  $\|*\|_2^2$  is the squared Euclidean norm,  $A_{i(max/min)} = [\cot(\beta_{i(max/min)}) \ 1]$ ,  $h_i = [h_{xi} \ h_{yi}]^T$ , and  $q_{i(max/min)}$  is a parameter indicating the  $\beta_{i(max/min)}$  quadrature:

$$q_{i(max/min)} = \begin{cases} 0, & \text{if } \beta_{i(max/min)} \in 1^{st} \vee 4^{th} \text{ quadrant,} \\ 1, & \text{if } \beta_{i(max/min)} \in 2^{nd} \vee 3^{rd} \text{ quadrant.} \end{cases}$$



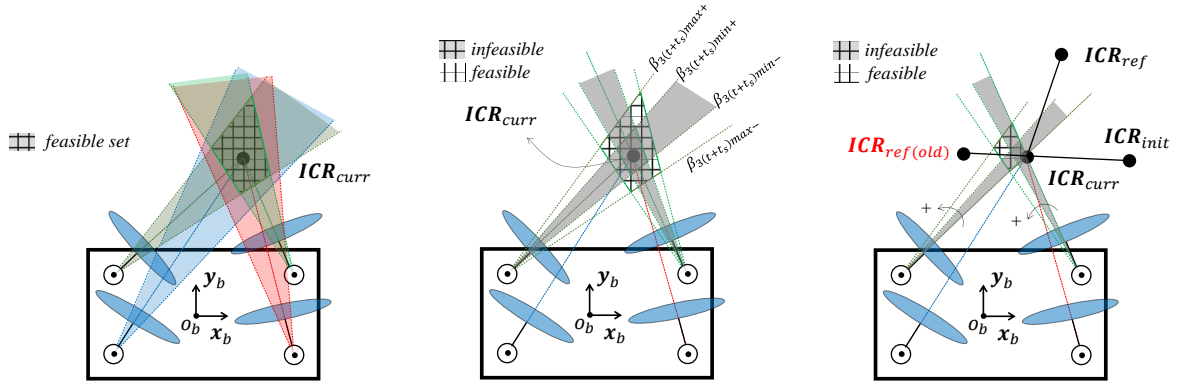


Fig. 5: Feasible set of solutions for the ICR coordinates, respecting the maximum steer joint performance with: (a) stationary ICR, (b) moving ICR (general case), and (c) moving ICR (particular case) with  $\dot{\beta}_2 > 0$  and  $\dot{\beta}_3 > 0$ ,  $t_s$  is the sample time.

The  $ICR_{ref}$  is obtained from (10) by numeric integration, or from (11) in case of kinematic singularity. Using  $ICR_{ref}$  instead of  $ICR^*$  provides smooth damped behavior, since its computation depends on  $ICR_{err}$  (refer again to (10)). The constraints are the straight lines of slopes  $\beta_{i(max)}$  and  $\beta_{i(min)}$ :

$$\begin{aligned} \dot{\beta}_{i(max)(ref)} &= \begin{cases} \dot{\beta}_i + \ddot{\beta}_{max} * t_s, & \dot{\beta}_{i(max)(ref)} < \dot{\beta}_{max} \\ \dot{\beta}_{max}, & \text{otherwise} \end{cases}, \\ \dot{\beta}_{i(min)(ref)} &= \begin{cases} \dot{\beta}_i - \ddot{\beta}_{max} * t_s, & \dot{\beta}_{i(min)(ref)} > -\dot{\beta}_{max} \\ -\dot{\beta}_{max}, & \text{otherwise} \end{cases}, \end{aligned}$$

$$\beta_{i(max/min)} = \beta_i + \dot{\beta}_{(max/min)(ref)} * t_s. \quad (12)$$

Finally, using the optimal  $ICR_{next}$ , the next best steer angles are computed:

$$\beta_{i(next)} = \arctan 2(Y_{next} - h_{yi}, X_{next} - h_{xi}) - \frac{\pi}{2}. \quad (13)$$

#### D. Fixing Numeric Issues

Two numeric issues can be observed, when employing the previously described controller. The first originates from the  $\arctan 2()$  function in (13) whenever  $\beta_{i(next)}$  moves between  $4^{th}$  and  $1^{st}$ , or between  $2^{nd}$  and  $3^{rd}$  quadrants. This results in  $2\pi$  jumps in the computed angle. The second occurs when the ICR passes by kinematic singularity: moving on a straight line passing by a steer joint axis, and then moving from one side of that axis to the other, will require a  $\pi$  jump in  $\beta_{i(next)}$ . This problem has been avoided in our previous work [10], by employing the steer joint rate. However, a different method is necessary here, since we compute the joint angle rather than its rate in (13). To handle these two issues, we use  $\Delta\beta_i = \beta_{i(next)} - \beta_i$  to detect and fix the jumps:

$$\dot{\beta}_{i(ref)} = \begin{cases} (\Delta\beta_i + l\pi)/t_s, & \Delta\beta_i < -l\pi + 2\dot{\beta}_{max}t_s \\ (\Delta\beta_i - l\pi)/t_s, & \Delta\beta_i > l\pi - 2\dot{\beta}_{max}t_s \\ \Delta\beta_i/t_s, & \text{otherwise,} \end{cases} \quad (14)$$

where  $l \in \{1, 2\}$  depending on the case being handled, and  $\dot{\beta}_{i(ref)}$  is the joint-space steering reference/command signal.

#### E. The Driving Rate

Computing the wheel drive rate  $\dot{\phi}_{i(ref)}$  from (5), requires the reference velocity vector  $\dot{\xi}_{ref}$  equivalent to  $ICR_{next}$ . However, this is not feasible, since one information is missing when mapping from the  $2D$  space of  $ICR_{next}$  to the  $3D$  space of  $\xi_{ref}$ . To avoid such inconvenience, we propose a separate robot *Space Velocity Controller*, to compute the driving rate. First, a simple proportional controller is used to set the initial value of the reference signal  $\dot{\xi}_{ref(init)}$ :

$$\dot{\xi}_{ref(init)} = K_p(\xi^* - \dot{\xi}), \quad (15)$$

with  $K_p$  a positive scalar gain, and  $\xi^*$  the desired robot velocity output by the task controller. The corresponding  $\dot{\xi}_{ref(init)}$  (obtained by numeric integration of (15)) may be incompatible with the "next sample time" robot configuration obtained by substituting  $\beta_{ref}$  (obtained by integrating (14)) in the kinematic constraint matrix (3), where  $G_{ref} = G(\beta_{ref})$ . Hence,  $\dot{\xi}_{ref}$  is obtained by projecting  $\dot{\xi}_{ref(init)}$  onto the null-space of  $G_{ref}$ :

$$\dot{\xi}_{ref} = (I - G_{ref(d)}^+ G_{ref}) \dot{\xi}_{ref(init)}. \quad (16)$$

Substituting by  $\dot{\xi}_{ref}$  in (5), we finally obtain  $\dot{\phi}_{i(ref)}$ .

#### IV. EXPERIMENTS

In this section, the benchmark test trajectory used to evaluate the proposed discontinuity robust controller is detailed and validated experimentally on the industrial mobile robot Neobotix MPO700 (shown in Fig. 6). All parameters used in the experiments are provided in Table I. A video of the experiment can be found on the IDH YouTube channel <sup>1</sup>

##### A. Benchmark trajectory

The benchmark trajectory proposed in [10], depicted in Fig. 7, will be employed to test the robustness of the proposed controller against both representation and kinematic singularities. It consists of 5 distinct velocity trajectories, with connections not tailored. Hence, the performance against commanded velocity discontinuity can also be verified. In the following, we detail each of the 5 velocity trajectories (the desired acceleration is the derivative of the provided velocity).

<sup>1</sup><https://youtu.be/P9aIIwz6qLE>



Fig. 6: Photograph of the Neobotix MPO700 industrial SWMR.

1) *Parabolic ICR position profile*: This test excites steer joint motion in the vicinity of kinematic singularity, to check if it will respect the joint limits. A parabolic ICR-point motion with vertex at one of the steering axes is used, and implemented smoothly using a constant velocity profile:

$$\begin{cases} \dot{x}^* = 0.5 * (2\dot{y} + h_{x2})^2 + h_{y2} \\ \dot{y}^* = T_{c4b}(\dot{y}_i = 0, \dot{y}_f = -h_{x2}, t_i = 2, t_f = 6, t) \\ \dot{\theta}^* = 0.5. \end{cases} \quad (17)$$

Here,  $T_{c4b}$  is a 4<sup>th</sup> order trajectory, taking initial velocity  $\dot{y}_i$ , final velocity  $\dot{y}_f$ , initial time  $t_i$ , final time  $t_f$ , and current time  $t$  as arguments:

$$T_{c4b} = \begin{cases} a_0 + a_1\delta t_1 + a_2\delta t_1^2 + a_3\delta t_1^3 + a_4\delta t_1^4, & t_i \leq t < t_1 \\ a_5, & t_1 \leq t < t_2 \\ a_0 + a_1\delta t_3 + a_2\delta t_3^2 + a_3\delta t_3^3 + a_4\delta t_3^4, & t_2 \leq t < t_f. \end{cases}$$

Coefficients  $a_0 \dots a_5$ , are computed by setting the initial and final acceleration/jerk (*boundary conditions*) to zero,  $\delta t_1 = t - t_i$ ,  $\delta t_2 = t - t_1$ , and  $\delta t_3 = t - t_2$ , while  $t_1 = t_i + 0.1 * (t_f - t_i)$ , and  $t_2 = t_f - 0.1 * (t_f - t_i)$ . Before employing the velocity profile in (17), the following velocity command is applied, to guarantee that the parabolic ICR profile starts at the correct initial condition:

$$\dot{\xi}^* = 0.5 * [h_{x2}^2 + h_{y2} \quad 0 \quad 1]^T, \quad 0 \leq t < 2.$$

2) *ICR point at kinematic singularity*: This test reveals the behavior of the steer joint exactly at kinematic singularity. Will it respect the maximum steer joint performance limits? Will it keep operating at the maximum limits or move at "low" velocity? The latter is preferable, in terms of energy consumption and hardware safety. We used:

$$\dot{\xi}^* = 0.5 * [h_{y1} \quad -h_{x1} \quad 1]^T, \quad 6 \leq t < 8.$$

TABLE I: Robot and Controller parameters used in the experiments.

$a$	0.19m	$b$	0.24m	$d$	0.045m
$r_w$	0.09m	$\delta$	0.001	$R_\infty$	10m
$\delta_1$	$1^{-9}$	$\dot{V}_{max}$	10m/s	$\beta_{th}$	0.005rad.
$\lambda$	3.7	$R_{zone}$	0.015m	$\beta_{max}$	2rad./s
$\beta_{max}$	25rad./s <sup>2</sup>	$t_s$	25ms	$K_p$	2

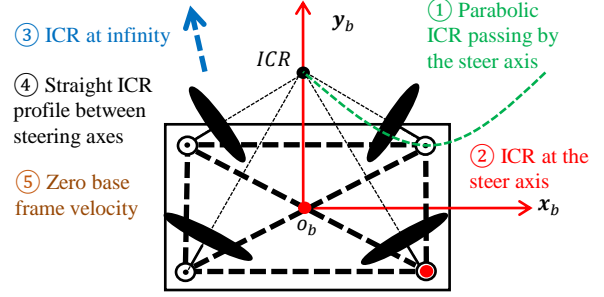


Fig. 7: Schematic model of the 5 benchmark test trajectories.

3) *Pure linear motion*: Tests the robustness against representational singularity in ICR-based models and controllers, that usually employ  $\theta$  in the denominator, by sending "zero angular velocity" and arbitrary linear velocity  $v_{ar}$ :

$$\dot{\xi}^* = [v_{ar1} \quad v_{ar2} \quad 0]^T, \quad 8 \leq t < 10.$$

4) *Straight line motion of the ICR between steering axes*: Consists in moving the ICR along a straight line connecting any two steering axes (*i.e.*, six straight lines for the 4 SWMR). This tests the performance of Cartesian space kinematic models, developed using only two steerable wheels, since an ICR motion on the line connecting them will result in undefined motion for the other steering joints. This test is realized by the following command sequence:

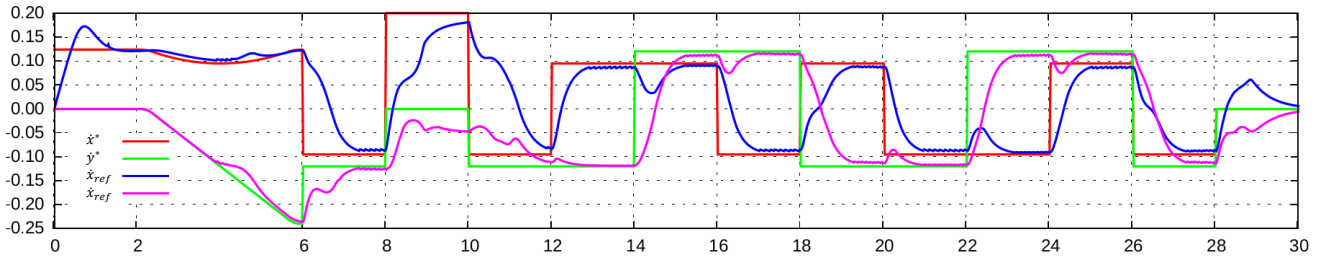
$$\begin{aligned} \dot{\xi}^* &= 0.5 * [h_{y1} \quad -h_{x1} \quad 1]^T, & 10 \leq t < 12, \\ \dot{\xi}^* &= 0.5 * [h_{y2} \quad -h_{x2} \quad 1]^T, & 12 \leq t < 14, \\ \dot{\xi}^* &= 0.5 * [h_{y3} \quad -h_{x3} \quad 1]^T, & 14 \leq t < 16, \\ \dot{\xi}^* &= 0.5 * [h_{y4} \quad -h_{x4} \quad 1]^T, & 16 \leq t < 18, \\ \dot{\xi}^* &= 0.5 * [h_{y2} \quad -h_{x2} \quad 1]^T, & 18 \leq t < 20, \\ \dot{\xi}^* &= 0.5 * [h_{y1} \quad -h_{x1} \quad 1]^T, & 20 \leq t < 22, \\ \dot{\xi}^* &= 0.5 * [h_{y4} \quad -h_{x4} \quad 1]^T, & 22 \leq t < 24, \\ \dot{\xi}^* &= 0.5 * [h_{y3} \quad -h_{x3} \quad 1]^T, & 24 \leq t < 26, \\ \dot{\xi}^* &= 0.5 * [h_{y1} \quad -h_{x1} \quad 1]^T, & 26 \leq t < 28. \end{aligned}$$

5) *Zero Velocity*: Tests the robustness against representation singularity in Cartesian velocity based models employing  $\dot{x}$ ,  $\dot{y}$ , and  $\dot{\theta}$  in the denominator of (4). We use:

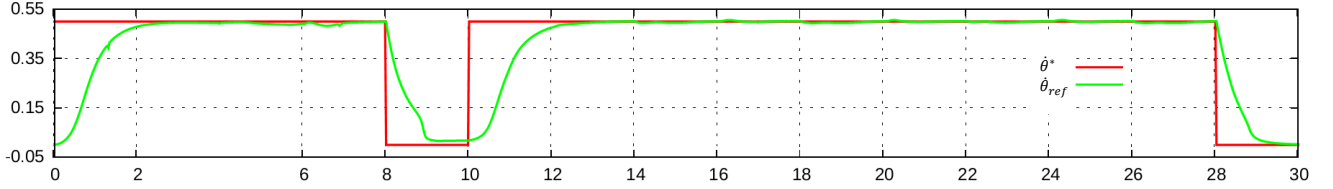
$$\dot{\xi}^* = [0 \quad 0 \quad 0]^T, \quad 28 \leq t < 30.$$

## B. Results and Discussion

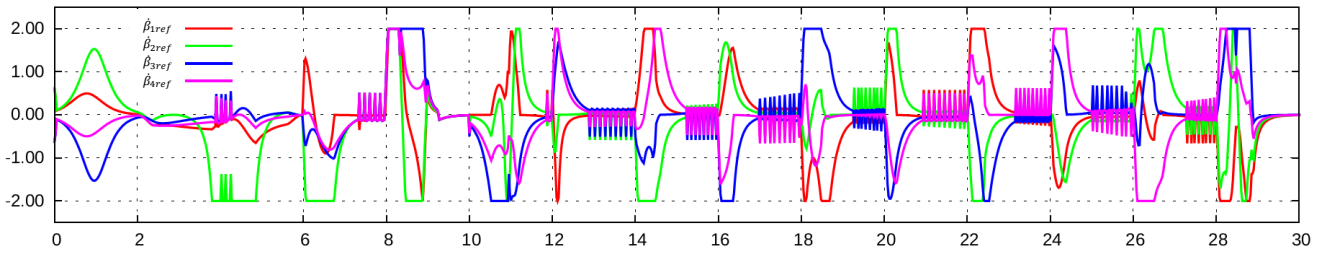
Applying the benchmark test trajectory shown in Fig. 8a and Fig. 8b to the proposed controller, we obtain the steer joint velocity and acceleration commands in Fig. 8c and Fig. 8d, respectively. As shown, these commands respect the maximum joint performance limits. The corresponding Cartesian space velocity is also shown in Fig. 8a and Fig. 8b, where a deviation from the trajectory is autonomously performed to accommodate for the "planned discontinuity". When implemented on the Neobotix-MPO700, the steer joint velocity response is obtained as in Fig. 9a, showing some overshoot due to imperfect embedded controller. However,



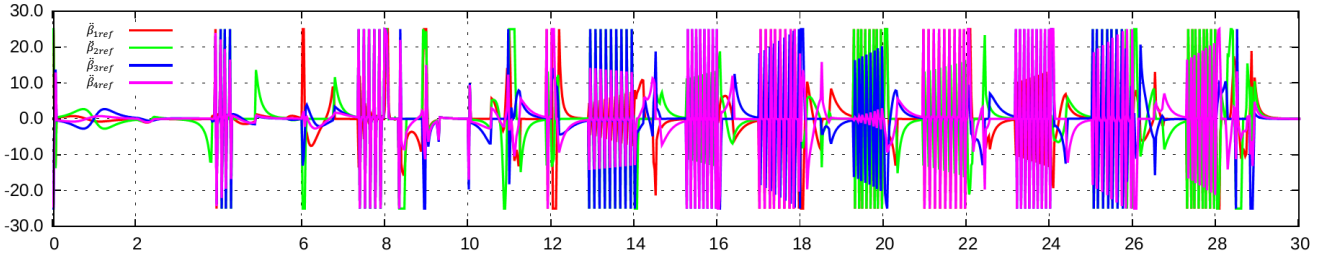
(a) Linear velocity: desired and output by the proposed controller.



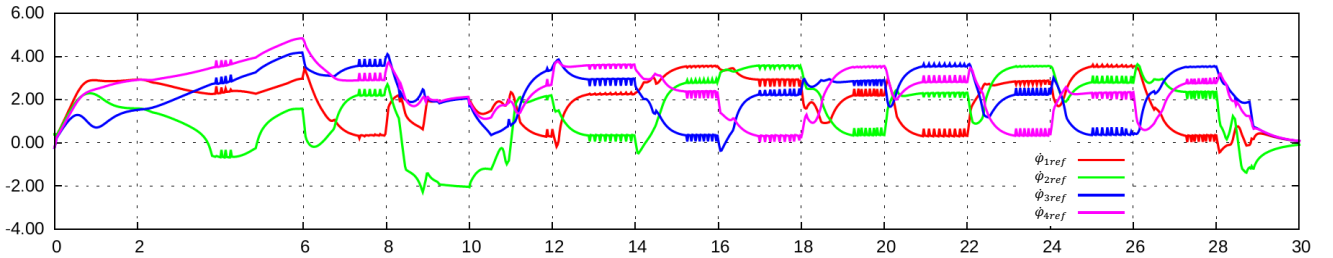
(b) Angular velocity: desired and output by the proposed controller.



(c) Output steering velocity command of the proposed controller.



(d) Output steering acceleration command of the proposed controller.



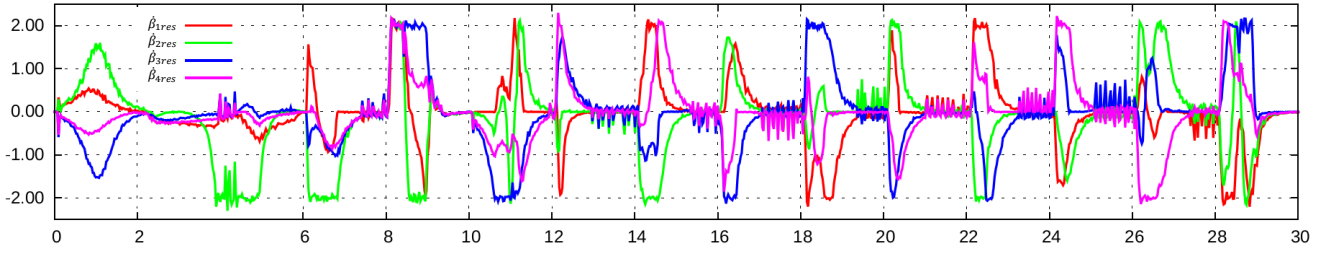
(e) Output drive speed command of the proposed controller.

Fig. 8: Results of the proposed controller along the “discontinuous” benchmark trajectory.

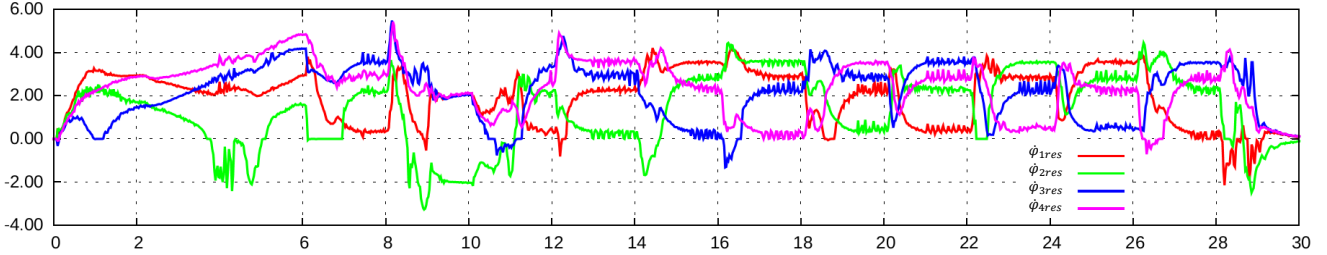
a better tuning of the embedded controller parameters, to improve the response, is still an open technical issue. Similarly, the command and response signals of the wheel drive joints are shown in Fig. 8e and Fig. 9b, respectively. In Fig. 8c, we can see that during the parabolic ICR motion, while passing the singular configuration, the second steer joint velocity grows, while respecting the maximum limit. In the second test where the robot must rotate about the

first steer joint axis (kinematic singularity), we can observe that: 1. The singular joint does not rotate at the maximum limit, which is favorable as noted earlier, 2. Low amplitude vibration exists in the steering joints other than the singular one (see also test 4). This is due to the singularity treatment algorithm, which keeps pushing the  $ICR_{ref}$  from one side of the singular zone circle to the other. This vibration can be further lowered in amplitude, by decreasing the controller





(a) Output steering velocity response of the proposed controller.



(b) Output drive speed response of the proposed controller.

Fig. 9: Joint space response results of the proposed controller, applied to the MPO700 industrial mobile robot.

sample time  $t_s$ , beyond the response time of the mechanical system (the control loop can run at 2 milliseconds sample time on a CORE i7 processor). However this was not achievable without rewriting the codes of the embedded low level controller, a task that we plan among future work. The frequency of such vibration can also be attenuated by taking into account the maximum steering jerk in the QP formulation. It is worth noting that this kinematic controller assumes a sufficiently good low level joint space dynamic controller implemented on the robot in hand. Otherwise, low acceleration values are assumed with maximum performance parameters like  $\dot{V}_{max}$ ,  $\dot{\beta}_{max}$ , and  $\dot{\phi}_{max}$  computed at the worst case loading scenario based on the application in hand.

## V. CONCLUSIONS

A motion-discontinuity robust controller has been developed and successfully tested on an industrial mobile robot. A discontinuous benchmark test trajectory that excites representation and kinematic singularities has been performed by the proposed controller with success. Maximum steer joint performance limits are taken into account and shown to be well respected throughout the experiments. In future work, the drive joint maximum performance limits will be added to the proposed framework.

## ACKNOWLEDGMENTS

This work is supported by the French region Occitanie (project CoBot@LR) and by the PSA Robotics Department.

## REFERENCES

- [1] P. R. Giordano, M. Fuchs, A. Albu-Schaffer, and G. Hirzinger, "On the kinematic modeling and control of a mobile platform equipped with steering wheels and movable legs," in *IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 4080–4087.
- [2] B. Thuilot, B. d'AAndrea Novel, and A. Micaelli, "Modeling and feedback control of mobile robots equipped with several steering wheels," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 3, pp. 375–390, 1996.
- [3] A. Dietrich, T. Wimbeck, A. Albu-Schffer, and G. Hirzinger, "Singularity avoidance for nonholonomic, omnidirectional wheeled mobile platforms with variable footprint," in *IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 6136–6142.
- [4] C. P. Connette, C. Parltitz, M. Hagele, and A. Verl, "Singularity avoidance for over-actuated, pseudo-omnidirectional, wheeled mobile robots," in *IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 4124–4130.
- [5] U. Schwesinger, C. Pradalier, and R. Siegwart, "A novel approach for steering wheel synchronization with velocity/acceleration limits and mechanical constraints," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 5360–5366.
- [6] R. Oftadeh, R. Ghabcheloo, and J. Mattila, "A novel time optimal path following controller with bounded velocities for mobile robots with independently steerable wheels," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013, pp. 4845–4851.
- [7] C. Connette, M. Hgele, and A. Verl, "Singularity-free state-space representation for non-holonomic, omnidirectional undercarriages by means of coordinate switching," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 4959–4965.
- [8] S. Chamberland, E. Beaudry, L. Clavien, F. Kabanza, F. Michaud, and M. Lauriay, "Motion planning for an omnidirectional robot with steering constraints," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 4305–4310.
- [9] R. Oftadeh, R. Ghabcheloo, and J. Mattila, "Time optimal path following with bounded velocities and accelerations for mobile robots with independently steerable wheels," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014, pp. 2925–2931.
- [10] M. Sorour, A. Cherubini, R. Passama, and P. Fraise, "Kinematic modeling and singularity treatment of steerable wheeled mobile robots with joint acceleration limits," in *IEEE Int. Conf. on Robotics and Automation*, 2016, pp. 2110–2115.
- [11] P. F. Muir and C. P. Neuman, "Kinematic modeling of wheeled mobile robots," *Journal of Robotic Systems*, vol. 4, no. 2, pp. 281–340, 1987.
- [12] G. Campion, G. Bastin, and B. D'Andrea-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.
- [13] A. Betourne and G. Campion, "Kinematic modelling of a class of omnidirectional mobile robots," in *IEEE Int. Conf. on Robotics and Automation*, 1996, pp. 3631–3636.
- [14] K. Low and Y. Leow, "Kinematic modeling, mobility analysis and design of wheeled mobile robots," *Advanced Robotics*, vol. 19, no. 1, pp. 73–99, 2005.
- [15] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *ASME Journal of Dyn. Sys., Measur., and Control*, vol. 108, no. 3, pp. 163–171, 1986.
- [16] C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Trans. on Sys., Man, and Cybernetics*, vol. 16, no. 1, pp. 93–101, 1986.