Motion discontinuity-robust controller for steerable wheeled mobile robots

Mohamed Sorour

Collège Doctoral du Languedoc-Roussillon

THÈSE

Pour obtenir le grade de

Docteur

Délivré par l'**Université de Montpellier**

Préparée au sein de l'école doctorale **I2S**
Et de l'unité de recherche **LIRMM UMR 5506**

Spécialité: **Systèmes Automatiques et Microélectroniques**

Présentée par **Mohamed SOROUR**

Motion discontinuity-robust controller for steerable wheeled mobile robots

Soutenue le 06/11/2017 devant le jury composé de

M. Reza Ghabcheloo	Associate Prof.	AUT, TUT	Rapporteur
M. Paolo Robuffo Giordano	DR, HDR	CNRS	Rapporteur
Mme. Nahid Armand	DR, PHD	PSA	Examinateur
M. André Crosnier	PU, HDR	UM, LIRMM	Président
M. Andrea Cherubini	MCF, HDR	UM, LIRMM	Co-encadrant
M. Philippe Fraisse	PU, HDR	UM, LIRMM	Directeur de Thèse
M. Lionel Lapierre	MCF, HDR	UM, LIRMM	Invité

I2S: École Doctorale Information Structures Systèmes

To my family and every teacher, and professor I have met and learnt from.

Abstract

Steerable wheeled mobile robots (SWMR) gain mobility by employing fully steerable conventional 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 $2D$ planar trajectories. They are cheaper and have higher load carrying capacity than non-conventional wheels (e.g., Sweedish or Omni-directional), and as such preferred for industrial applications. However, this type of mobile robot structure presents challenging *basic* control issues of steering coordination to avoid actuator fighting, avoiding kinematic (ICR at the steering joint axis) and representation (from the mathematical model) singularities. In addition to solving the *basic* control problems, this thesis also focuses attention and presents solutions to *application level* problems. Specifically we deal with two problems: the first is the necessity to "discontinuously" reconfigure the steer joints, once discontinuity in the robot trajectory occurs. 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. The second appears in applications requiring that some heading angle is to be maintained, some object or feature stays in the field of view (e.g., for vision-based tasks), or the translation verse changes. Then, the ICR point is required to move long distances from one extreme of the workspace to the other, usually passing by the robot geometric center, where the feasible robot velocity is limited. In these application scenarios, the state-of-art ICR based controllers will lead to unsatisfactory behavior/results. In this thesis, given a SWMR with maximum steering velocity and acceleration bounds, an algorithmic controller is designed to allow following - as faithfully as possible - a desired non-smooth (discontinuous), $3D$ Cartesian velocity trajectory assuming the no-slip, no-skid kinematic constraints are well respected, while being robust against kinematic and representation singularities. Our findings has been validated experimentally on an industrial mobile base.

Résumé

Les robots mobiles orientables à roues gagnent en mobilité en utilisant des roues conventionnelles entièrement orientables, ayant deux articulations actives, une pour la direction et une autre pour la conduite. En dépit d'un seul degré de mobilité (DOM) (défini ici comme les degrés de liberté instantanément accessibles DOF), correspondant à la rotation autour du centre de rotation instantané (ICR), de tels robots peuvent réaliser des trajectoires complexes $2D$ planaires. Ils sont moins chers et ont une capacité de charge plus élevée que les roues non conventionnelles (par exemple Sweedish ou Omni-directional), et sont donc préférés pour des applications industrielles. Cependant, ce type de structure de robot mobile présente des problèmes de contrôle de la coordination de la direction pour éviter la lutte contre les actionneurs, en évitant les singularités cinématiques (ICR à l'axe de direction) et de représentation (à partir du modèle mathématique). En plus de résoudre les problèmes de contrôle *basic*, cette thèse concentre également l'attention et présente des solutions aux problèmes de *niveau d'application*. En particulier, nous traitons de deux problèmes: le premier est la nécessité de reconfigurer "de façon discontinue" les joints de direction, une fois que la trajectoire du robot est discontinue. Une telle situation - la discontinuité dans le mouvement du robot - est plus susceptible de se produire de nos jours, dans le domaine émergent de la collaboration homme-robot. Les robots mobiles travaillant à proximité de travailleurs humains en déplacement rapide rencontreront généralement une discontinuité dans la trajectoire calculée en ligne. La seconde apparaît dans les applications nécessitant qu'un certain angle de cap soit maintenu, un objet ou une caractéristique reste dans le champ de vision (par exemple, pour des tâches basées sur la vision), ou le verset de traduction change. Ensuite, le point ICR est nécessaire pour déplacer de longues distances d'un extrême de l'espace de travail à l'autre, en passant généralement par le centre géométrique du robot, où la vitesse du robot est limitée. Dans ces scénarios d'application, les contrôleurs à base d'ICR à la pointe de la technologie conduiront à des comportements / résultats insatisfaisants. Dans cette thèse, nous résolvons les problèmes de niveau d'application mentionnés ci-dessus; à savoir la discontinuité dans les commandes de vitesse du robot, et une planification meilleure / efficace pour le contrôle du mouvement du point ICR tout en respectant les limites maximales de performance des articulations directrices et en évitant les singularités cinématiques et représentatives. Nos résultats ont été validés expérimentalement sur une base mobile industrielle.

Acknowledgements

I want to thank all the nice colleagues and friends that I have been in contact with during these three years of research. I would like to thank my main supervisor, Phlippe Fraisse for his continuous encouragement and fruitful discussions. I also want to thank my direct supervisor, Andrea Cherubini for the numerous, long and exhausting discussions and for being patient despite me sometimes being obstinate. I greatly thank my friends; Gamal Elghazaly, Moussab Bennehar, Mohamed Boushaki, Mohamed Haijoubi, Osama Mazhar, Firas Kaddachi, Benjamin Navarro for the releiving discussions and coffee breaks. I would like to thank the reviewers; Reza Ghabcheloo, and Paolo Robuffo Giordano for their nice and detailed review of my thesis, despite exhausting. I finally wish to thank my family, to whom I dedicate this work.

Contents

1 Introduction **1**

1.1 Motivation: kitting in the automotive industry 1

1.2 Wheeled Mobile Robots 2

1.3 Fully Omni-directional Robots 3

1.4 Steerable Mobile Robots 4

1.5 Thesis Outline 6

2 State of the Art **8**

2.1 SWMR Platforms 8

2.2 Gaps in the Literature 12

2.3 Thesis Contributions 14

2.3.1 Singularity-free $3D$ Cartesian Space Model 14

2.3.2 Discontinuity Robust ICR Based Controller 14

2.3.3 Complementary ICR Route 15

3 Steerable Wheeled Robot Kinematic Modeling **17**

3.1 Introduction 17

3.2 Kinematic Modeling 18

3.2.1 Cartesian Space Model Formulation 18

3.2.2 Odometry Model 21

3.2.3 $2D$ ICR Space to $3D$ Cartesian Space Mapping 21

3.3 Kinematic Singularity Treatment 22

3.4 Benchmark Test Trajectory 24

3.4.1 Parabolic ICR position profile 24

3.4.2 ICR point at kinematic singularity 27

3.4.3 Pure linear motion 27

3.4.4 Straight line motion of the ICR between steering axes 27

3.4.5 Zero Velocity 28

3.5 Experiments and Results . . . 28
3.6 Drawbacks . . . 31
3.7 Conclusion . . . 33
3.8 Link with Next Chapter . . . 33

4 Motion Discontinuity Robust ICR Controller 34
4.1 Introduction . . . 34
4.1.1 Motivation . . . 35
4.1.2 Proposed control framework . . . 36
4.2 ICR Velocity Controller . . . 36
4.3 Handling Kinematic Singularity . . . 39
4.4 Optimal next sample time ICR Point . . . 40
4.5 Fixing Numeric Issues . . . 43
4.6 The Driving Rate . . . 43
4.7 Experiments . . . 44
4.7.1 Benchmark trajectory . . . 44
4.7.1.1 Parabolic ICR position profile . . . 44
4.7.1.2 ICR point at kinematic singularity . . . 45
4.7.1.3 Pure linear motion . . . 45
4.7.1.4 Straight line motion of the ICR between steering axes 45
4.7.1.5 Zero Velocity . . . 46
4.7.2 Results and Discussion . . . 46
4.8 Conclusion . . . 48
4.9 Link with Next Chapter . . . 48

5 Complementary Route Based ICR Controller 50
5.1 Introduction . . . 50
5.2 Complementary ICR Route . . . 53
5.2.1 Complementary Versus Direct Routes . . . 54
5.2.2 One Sample Period ICR Border Switching . . . 57
5.2.3 Route Planning . . . 58
5.2.4 Automatic Joint Space Border Switching . . . 60
5.3 Direct OR Complementary . . . 62
5.4 Experiments . . . 65
5.4.1 Command Fulfillment Index . . . 66
5.4.2 Discontinuous Velocity Commands Simulation . . . 66
5.4.3 Practical Application Scenario and Results . . . 72

5.5 Conclusion 78

6 Conclusion and Future Perspectives **79**

6.1 Summary of the Thesis 79

6.2 Conclusions and Future Work 81

A Singularity Treatment and Acceleration Limits **83**

A.1 Case I: $\delta_{2c} \leq 0$ 83

A.2 Case II: $\delta_{2c} > 0$ 84

Bibliography **85**

List of Symbols and Acronyms

$\boldsymbol{\omega}$	3×1 Angular velocity vector
$\boldsymbol{\xi}$	Mobile robot pose vector
$\boldsymbol{A}^{+}$	Moore-Penorse pseudoinverse of matrix $\boldsymbol{A}$
$\boldsymbol{A}^{T}$	Transpose of matrix $\boldsymbol{A}$
$\boldsymbol{I}_n$	The n-dimensional identity matrix
$\boldsymbol{v}$	3×1 Linear velocity vector
$\mathcal{F}_i$	Cartesian coordinate system of frame i
$\mathcal{N}(\boldsymbol{A})$	Null space of matrix $\boldsymbol{A}$
$\mid * \mid$	Modulus
$\mid\mid * \mid\mid$	Euclidean norm
$SO(3)$	Special orthogonal group
${}^{j}\boldsymbol{p}_i$	Position vector of the origin of frame i with respect to the origin of frame j
${}^{j}\boldsymbol{R}_i$	Rotation matrix of frame i with respect to frame j
DOF	Degree Of Freedom
DOM	Degree Of Mobility
FAKM	Forward Actuation Kinematic Model
FOMR	Fully Omni-directional Mobile Robot

IAKM	Inverse Actuation Kinematic Model
ICR	Instantaneous Center of Rotation
SWMR	Steerable Wheeled Mobile Robot

List of Figures

1.1 Moving cart holding the automobile dashboard and the kit (colored white in the left picture), photo courtesy of PSA. 2
1.2 Neobotix-MPO700 steerable mobile robot used in this thesis. 5
1.3 EZ10 bi-steerable vehicle used in [NTLM16]. 5

2.1 The Care-O-bot-3 service robot in [RCF+09] employs a pseudo omni-directional mobile structure. 9
2.2 The iMoro steerable wheeled mobile manipulator [OGM13] (a), and (b) the ExoMars rover 6 wheeled SWMR of the European Space Agency [SPS12]. 9
2.3 Azimut SWMR development. Azimut-1 of [LNL+06a] (left), Azimut-2 of [LNL+06b] (middle) and Azimut-3 of [CLM10] (right). 10
2.4 Rollin' Justin variable footprint SWMR. The dual arm mobile manipulation system [BOW+07, FBG+09] (left), legged SWMR extended [GFASH09] (middle) and folded (right) 11
2.5 The experimental 2 wheeled SWMR used in [GH14] (a), and (b) another 2 wheeled steerable mobile manipulator used in [SMG15]. . . . 12
2.6 HERMIES-III robot, 2 wheeled SWMR used in [RU93] (a), and (b) six wheeled field SWMR of [MF00]. 13
2.7 Field Steerable mobile vehicle used in [CLTB09]. 13

3.1 Schematic model of a four wheeled SWMR. 19
3.2 Effect of different values of δ_2 on the steering velocity while passing kinematic singularity. The lower the value, the more the required acceleration. 23
3.3 Schematic model of the 5 benchmark test trajectories. 25
3.4 Linear with fifth order blends ICR point motion trajectory in first experiment (upper), its first derivative being a constant with fourth order blends (middle), and the second derivative (lower). 26

3.5 Photograph of the Neobotix MPO700 industrial SWMR. 29
3.6 Output commands of the kinematic model employing the proposed singularity treatment, and excited with the benchmark test trajectory. 30
3.7 Measured response of the experiments employing the embedded controller, while executing the benchmark test trajectory. 31
3.8 Measured response of the experiments employing the kinematic model plus singularity treatment, while executing the benchmark test trajectory. 32

4.1 Proposed motion discontinuity-robust control framework. 37
4.2 Relationship 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 $\boldsymbol{ICR}_{curr}$. 40
4.3 ICR point approaching the singularity zone (in red) of the 2^{nd} steer joint. ICR motion direction is indicated by the top arrow. 41
4.4 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. 42
4.5 Results of the proposed controller along the "discontinuous" benchmark trajectory. 47
4.6 Joint space response results of the proposed controller, applied to the MPO700 industrial mobile robot. 48

5.1 An example of heading control application, where changing the error sign will result in vast ICR point motion requirement. 51
5.2 An example of heading control application, where changing the error sign will result in vast ICR point motion requirement. 52
5.3 Steerable mobile robot $2D$ ICR space is bounded in this work by 4 border lines corresponding to $x = \pm R_\infty$ and $y = \pm R_\infty$ defined in the robot frame $\boldsymbol{\mathcal{F}}_b$ with origin at the robot center. An ICR point moving from the current to the desired value can follow a direct route (in red), or a complementary one (in green). Here the parameter $R_\infty = 10m$. . 53
5.4 Illustrative schematic showing the complementary versus the direct ICR routes. 54
5.5 The locally optimal complementary ICR route (in green) among other possible routes (magenta and blue) satisfying one constraint $y = R_\infty$. 55

5.6 The global optimum complementary route (magenta) among the four local optima, each satisfying one straight line constraint. 56
5.7 Evaluating the minimum value of $R_{\infty(e)}$ at the worst case scenario (biggest change in steer joint $\delta\beta$) to be able to perform border switching in one sample period. The mobile base schematic is magnified w.r.t. the ICR border lines for illustrative purposes. 58
5.8 Implementation of the complementary route; (a) step1: moving to the optimum border point, (b) step2: extending the border lines, (c) border switching in one sample period, and (d) retracting border line, then completion as direct route. 59
5.9 Motion-discontinuity robust controller architecture with the complementary controller addition shown in colored blocks. 61
5.10 Variation of the maximum ICR velocity with robot's ICR position, featuring the Neobotix-MPO700 industrial mobile robot approximately to scale, with $R_\infty = 10m$. 64
5.11 Intersection of the direct ICR route with the elliptic footprint of the mobile robot should be in favor of using the complementary route. . . 65
5.12 Desired (discontinuous) ICR points used in simulation. The relative dimensions of the mobile base schematic as well as the points are drawn at approximately 1 : 1 scale for clarification. The initial steer joint configuration corresponds to the equivalent ICR point of the first velocity command vector. 68
5.13 Simulation results using the complementary route controller with (a), (b) the $x - axis$, and $y - axis$ components respectively of the desired and reference (output of the controller) ICR points (in *meters*), (c) the command fulfillment index (*dimensionless*). In all plots, the abscissa represents the simulation time instant (in *seconds*). 69
5.14 Simulation results using the complementary route controller with (a), (b) the steering velocity (in rad/s) and acceleration (in rad/s^2) respectively, and (c) the time instants at which the complementary route algorithm is active. In all plots, the abscissa represents the simulation time instant (in *seconds*). 70

5.15 Simulation results using direct route (conventional) controller with (a), (b) the $x-axis$, and $y-axis$ components respectively of the desired and reference (output of the controller) ICR points (in *meters*), (c) the command fulfillment index (*dimensionless*). In all plots, the abscissa represents the simulation time instant (in *seconds*). 71

5.16 Simulation results using direct route (conventional) controller with (a), (b) the steering velocity (in rad/s) and acceleration (in rad/s^2) respectively. In all plots, the abscissa represents the simulation time instant (in *seconds*). 72

5.17 V-rep scene of the practical scenario, featuring the Neobotix-MPO700 mobile robot navigating to a target while performing static obstacle avoidance. The simulated visual cone applies a constraint on the heading of the robot so that the target is always seen by the robot, as is common in vision-based navigation. Here we apply such constraint to show the utility of the proposed controller. $\boldsymbol{\mathcal{F}}_t = (o_t, \boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{z}_t)$ denotes the target object frame. 73

5.18 Simulation results of the practical application using the complementary route controller with (a), (b), the desired and reference (output of the controller) $3D$ robot velocity respectively (linear velocity in m/s and angular velocity in rad/s), (c), (d), the $x-axis$, and $y-axis$ components respectively of the desired and reference ICR points (in *meters*). 74

5.19 Simulation results of the practical application using the complementary route controller with (a) the command fulfillment index (*dimensionless*) and the evolution of its average value $\overline{CFI}$, (b) the corresponding steering velocity (in rad/s). 75

5.20 Simulation results of the practical application using the direct route (conventional) controller with (a), (b), the desired and reference (output of the controller) $3D$ robot velocity respectively (linear velocity in m/s and angular velocity in rad/s), (c), (d), the $x-axis$, and $y-axis$ components respectively of the desired and reference ICR points (in *meters*). 76

5.21 Simulation results of the practical application using the direct route (conventional) controller with (a) the command fulfillment index (*dimensionless*) and the evolution of its average value $\overline{CFI}$, (b) the corresponding steering velocity (in rad/s). 77

5.22 Snap shots of the real experiment employing the complementary route ICR controller. 78

List of Tables

3.1 Robot and model parameters used in the singularity treatment experiments. 29

4.1 Robot and Controller parameters used in the singularity-robust control framework experiments. 44

5.1 Robot and Controller parameters used in the complementary route experiments. 67

Chapter 1

Introduction

Contents

1.1	**Motivation: kitting in the automotive industry**	**1**
1.2	**Wheeled Mobile Robots**	**2**
1.3	**Fully Omni-directional Robots**	**3**
1.4	**Steerable Mobile Robots**	**4**
1.5	**Thesis Outline**	**6**

1.1 Motivation: kitting in the automotive industry

In the automotive industry, final assembly operations are highly dominated by human operators with automation level lower than 30% [HTKR+15]. However some of these assembly tasks are of structured nature, consider for example the post-kitting tasks. The term kitting refers to the process of collecting the parts to be assembled in a bigger component, and inserting them into pockets of a mobile platform, named the kit. The kit then proceeds to the assembly line, where all the carried parts are assembled in the bigger automobile component. Examples include the door, or to one bigger entity on board of the kit itself like the dashboard, shown in Fig. 1.1. In the latter application, the kit containing the parts to be assembled, as well as the dashboard itself are in a state of continuous motion between several posts along the assembly line. Such motion can be discontinuous, due to human-worker safety protocols implemented on the kit-porting mobile platform.

The general objective of this thesis, is to robotize more assembly tasks and to increase the automation level. In this particular application, several robotic technologies are required, e.g., to identify the object to be assembled from the kit using vision, for vision-aided grasping, then dual arm manipulation to fit the part inside the

bigger entity (here the dashboard). All these steps are to be done while the kit/cart is in motion. Such objective requires the utilization of robust and highly maneuverable mobile robots in addition to developing new controllers taking into account the frequent human-worker presence in the robot workspace. This presence can cause highly discontinuous velocity commands to be sent to the mobile robot, and this is the target and main contribution of the thesis in hand. In what follows, we briefly introduce available mobile robot structures and come to a conclusion of the one most suitable to perform the desired task.

1.2 Wheeled Mobile Robots

Mobile robots are autonomous vehicles that are capable of navigating in target environments, being indoor, outdoor with smooth or rough terrain, and programmed to perform a variety of tasks depending on the application. They are used in rehabilitation/health care [LL00, YBB+09, SL09, LMD+17], surveillance [US16, RO18], home-/personal service [GHS04, BO08, PTC+15], factory automation [HNB+11, HBMK09], agriculture [BJ04, MF00] to name only a few applications. Wheeled mobile robots employ wheels as means of navigation, a distinct classification from treaded and legged robots (tanks, humanoids, hexapods, ... etc.). These in turn are mainly classified as conventional or non-conventional wheeled robots according to the wheel type, with the former being the famous standard flat disk wheel/tire, and the latter being including other wheel nature, e.g., omni-directional, Swedish, ball wheels to name only a few.

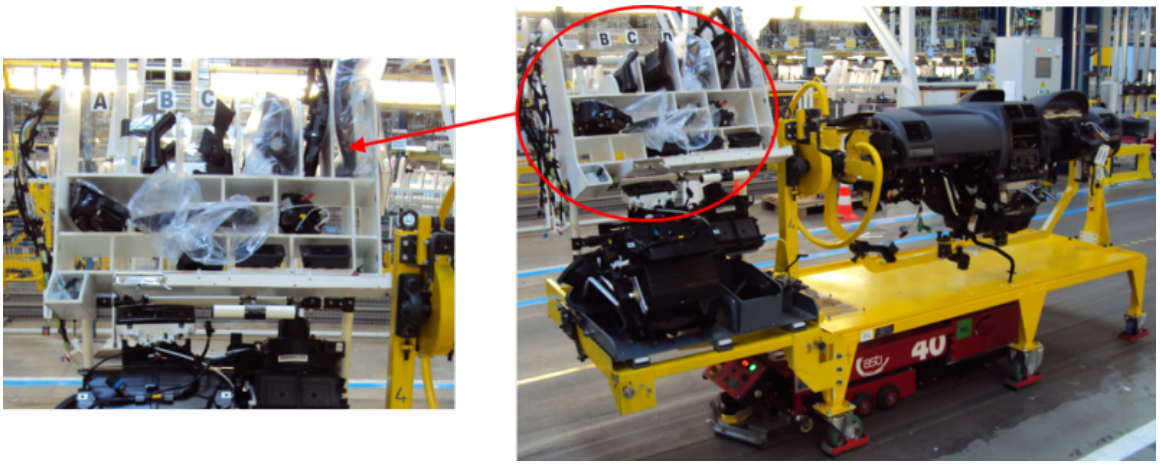


Figure 1.1: Moving cart holding the automobile dashboard and the kit (colored white in the left picture), photo courtesy of PSA.

Conventional wheeled mobile robot structure is classified based on the degrees of mobility and steerability into 5 types, according to the pioneering work of [CBN96]. Although this classification is not comprehensive (as it doesn't handle the case steerable wheels with offset), it covers most of the commonly used conventional wheeled mobile robots available nowadays. It is worth pointing out that the classification is not flawless. Differential drive system is classified as being of type $(2, 0)$ with 2 being the degree of mobility (the immediately accessible degrees of freedom) and 0 as the degree of steerability (the number of independently steerable wheels), the tricycle structure is of type $(1, 1)$, and the steerable mobile robots (synchronous drive) which is the focus of this thesis is designated as type $(1, 2)$. For these three examples, the classification does not indicate which is better in terms of mobility/maneuverability, ability to perform complex trajectories, as the synchronous drive is the most perform-able although its DOM is lower than that of the differential drive. Consequently, among the conventional wheeled mobile robots, the SWMR structure is the most convenient for the application in hand.

1.3 Fully Omni-directional Robots

Fully omni-directionality means that the mobile robot has 3DOM, that is, it can instantaneously move in any of the planar 3DOF. There are numerous ways of obtaining full omnidirectionality, using:

- non conventional wheels,
- differential drive system with rotating offset turrets mounted on it [CY17, ESWBB07],
- fully actuated castor wheels [Wad06, HK00].

In [ESWBB07], the authors introduced the idea of using a differential drive system fixed to an offset turret, to obtain an omni-directional system. Comparison was held with other omni-directional robot structures, employing powered castor wheels, and omni-wheels. In [CY17], the geometric parameters of such robot configuration were optimized based on the "mobility ellipsoid" introduce in the same work in analogy to the "manipulability ellipsoid" of arm manipulation [Yos85]. A comparison between different omni-directional wheel structures can be found in [FRC96]. The second and third methods (based on differential drive or Castor wheels) rely on conventional wheels in operation, which in turn can avoid operational problems of non-conventional

wheels (detailed in the sequel). However, there exist no commercially available FOMR employing these mobile robot structures, making them practically impossible to use in industrial setups in the near future.

Fully omni-directional mobile robots with non conventional wheels use:

- orthogonal wheels (also known as omni-directional wheels) as in [PK94]; these consist of several "perpendicular" small rollers mounted on the circumference of a conventional wheel, enabling side motion of the wheel and hence being holonomic,
- ball wheel mechanisms as in [WA95, CCT+13, TNFM12, MC02],
- Mecanum wheels, which are by far the most popular among non-conventional wheel types.

The Mecanum wheel (named after the Mecanum company) or Swedish wheel was invented by the Swedish engineer Bengt Ilon in 1973 [Ilo75]. Its design [Gfr08] varies from the omni-wheel by orienting the rollers at an angle other than 90^o (usually 45^o), providing smoother performance than omni-directional ones, and was first used in constructing the mobile robot "Uranus" [MN87a]. However, Swedish wheels require special attention in their design, to avoid vibration due to discontinuous wheel contact with the ground [BS03, DBB+02]. Other modifications are proposed to allow access of Mecanum wheels in rough terrains [RSK10]. Disadvantages of the Mecanum wheels include:

- poor odometry estimation without exteroceptive sensors [HKL10],
- vibration due to spaces between rollers,
- high cost,
- lower load carrying capacity than conventional wheels.

1.4 Steerable Mobile Robots

Also known in the literature by the names: synchronous drive, pseudo omni-directional, non-holonomic omni-directional, quasi omni-directional robots, steerable wheeled mobile robots consist of two or more independently and fully steerable conventional wheels. A fully steerable wheel contains two actuated degrees of freedom,

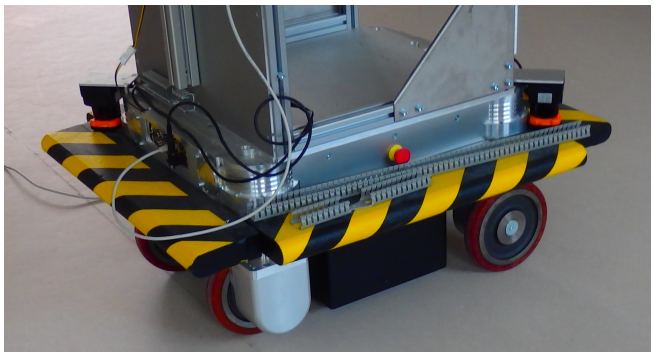


Figure 1.2: Neobotix-MPO700 steerable mobile robot used in this thesis.

one for steering, and the other for driving (wheel motion). In practice, SWMR exist in a configuration of 3 or more fully steerable wheels for robot mechanical stability.

Research topics in SWMR include but are not limited to:

- Kinematic modeling and control [SN11]
- Estimation of the ICR point [CLM10, LQXX09] and behavior in case of compliant actuators [SCSB06]
- Design and kinematic analysis [MLA+03, MLA+05, LNL+06a, FCF+10, BJ04, MF00].
- Extending the conventional car driving steering wheel (as interface for human driver) from the car like vehicles to the pseudo omni-directional ones [LQXX09].

Conventional wheeled mobile robots are classified according to their degrees of mobility and steerability defined in the pioneer work [CBN96]. Based on this classification, SWMR are of the Type $(1, 2)$ having one degree of mobility (DOM) and two degrees of steerability. The available DOM corresponds to the rotation about the

Figure 1.3: EZ10 bi-steerable vehicle used in [NTLM16].

instantaneous center of rotation (ICR) point. In the literature we can find two types of steerable mobile robots.

The first type utilizes 2 sets of mechanically coupled steerable wheels, the front and rear steering systems [MT90,AS93,AS03,HS03]. This type of robots are called bi-steerable vehicles, and an example is shown in Fig. 1.3. They are most commonly used for outdoor applications like autonomous cars [NTLM16], and agriculture [CLTB09, LTCM06], since their mechanical structure is simplified and these applications do not require high maneuverability.

The second type of steerable mobile robots utilizes 2 (but most practically a minimum of 3) or more fully steerable wheels, with no hardware coupling among the wheels (an example is shown in Fig. 1.2). Although both types have the same DOM, the second is more challenging since coordination between the independently controlled steering axes is needed, to achieve proper operation with least possible actuator fighting. Furthermore, the second type has almost no constraints on the location of the ICR, and as such uses the maneuver-ability of the mobile robot structure at its full.

Steerable mobile robots are expected to be smoother in performance at lower cost for the same load capacity, as compared to Mecanum wheel based robots (at the cost of having less mobility). As such, they are more convenient for the application in hand (kitting for the automotive industry). The objective of this work is to develop a SWMR kinematic controller that is suited for discontinuous trajectories, while respecting the robot kinematic constraints and actuation limits.

1.5 Thesis Outline

In this section, we present the general outline of the chapters of this thesis. The motivation behind each chapter will be presented in the sequel (Chapter 2), after exploring the state of the art for better understanding of the latest development in the subject and its defeciency with respect to the problem in hand. This thesis is organized as follows:

- **Chapter 2** presents the current state-of-art in mobile robotics and in particular in steerable wheeled mobile robots.
- **Chapter 3** presents the kinematic modeling of SWMR based on the $3D$ planar Cartesian velocity formalism. In addition, a kinematic singularity treatment is introduced and shown to have a damping effect on the joint space velocities at singular configurations.

- **Chapter 4** introduces our new algorithmic ICR-based controller that determines the "*next sample time*" steering coordinates in the presence of velocity commands discontinuity. This controller also respects the maximum steer performance limits in terms of velocity and acceleration.

- **Chapter 5** presents our new idea of complementary ICR route, as opposed to the direct ICR route used in the literature. We also present a methodology to evaluate both routes in terms of the expected time efficiency (convergence to desired ICR point), so that the shortest route is utilized.

- **Chapter 6** summarizes the main contributions of this thesis on SWMR. Conclusions are drawn and recommendations for future research directions are highlighted.

Chapter 2

State of the Art

Contents

2.1 SWMR Platforms 8
2.2 Gaps in the Literature 12
2.3 Thesis Contributions 14
2.3.1 Singularity-free $3D$ Cartesian Space Model 14
2.3.2 Discontinuity Robust ICR Based Controller 14
2.3.3 Complementary ICR Route 15

This chapter reviews the state of the art in hardware and control of steerable wheeled mobile robots. It shows the gaps and downsides in the literature, that motivate the contributions of this thesis. In SWMR, two types of singularities are most common in the literature (and will be mentioned frequently throughout this Chapter), designated as kinematic and representation singularities. The former refers to the state of the robot when the steer joint configuration corresponds to an ICR point at one of the steering axes, while the latter depends on the mathematical model used to represent the robot behavior which at some velocity values can be singular.

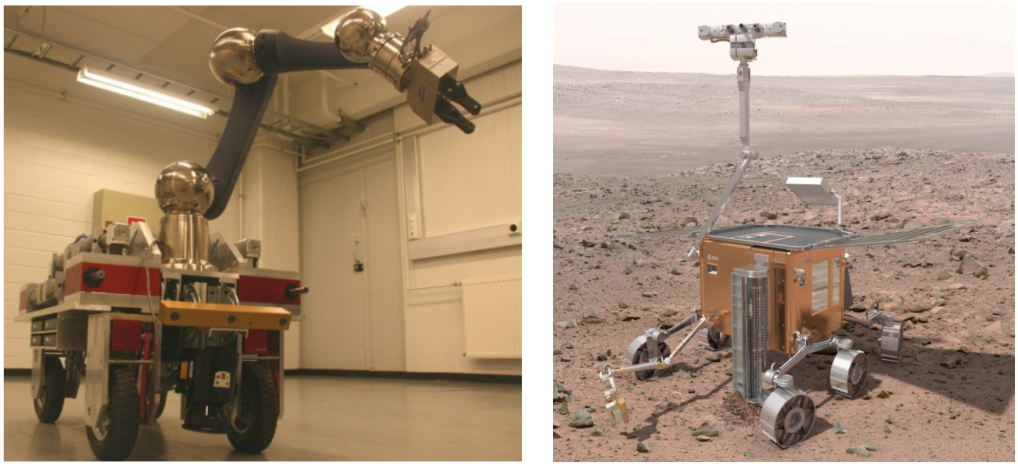
2.1 SWMR Platforms

The authors of [RCF+09] have developed the mobile service robot "Care-O-bot-3", shown in Fig. 2.1. This applies a two side concept, where the manipulation part is away from the user when s/he is being served, offered a drink, etc, and only the tray and the drink are facing the user during the interaction phase of operation. The "Care-O-bot-3" employs a 4 wheeled pseudo omni-directional structure for its mobility.

Figure 2.1: The Care-O-bot-3 service robot in [RCF+09] employs a pseudo omni-directional mobile structure.

In [CPHV08], the authors present their controller for the mobile base of the same robot, using a new representation for the ICR point, based on spherical coordinates to avoid singularities of the ICR representation in the Cartesian coordinates. In [CPHV09], they have added repulsive potential fields around the steering axes as a solution to the kinematic singularity and actuator saturation. In [CHB+10, CPHV10], the authors have added model predictive control (MPC) to address the sensitivity to local minima problems that arise due to the potential fields employed in [CPHV09]. However, later on in [CHV12], the authors revealed the existence of representational singularities in their spherical coordinate representation, leading to the need for coordinate switching to avoid such singularity.

Authors in [OGM13, OAGM13] developed the iMoro mobile manipulator (shown in Fig. 2.2a), employing a Pseudo omni-directional mobile robot structure. They



(a) (b)

Figure 2.2: The iMoro steerable wheeled mobile manipulator [OGM13] (a), and (b) the ExoMars rover 6 wheeled SWMR of the European Space Agency [SPS12].

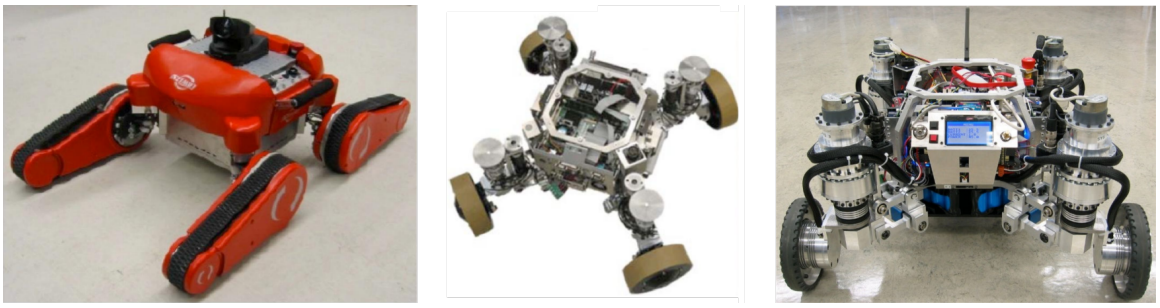


Figure 2.3: Azimut SWMR development. Azimut-1 of [LNL+06a] (left), Azimut-2 of [LNL+06b] (middle) and Azimut-3 of [CLM10] (right).

propose a non linear controller, which acts on the robot velocity and the path followed by the robot, so that it respects the steering and driving velocity bounds. The results have shown that the steering velocity is well bounded at kinematic singularity, although the singular steer joint keeps operating at the maximum limit. The authors have extended the controller to take into account the steer acceleration bounds also in [OGM14], which was further generalized to other mobile robot structures in [OGM15]. Figure 2.2b shows the ExoMars rover, a 6 wheeled SWMR of the European Space Agency which was the experimental platform used in [SPS12]. The authors use special spherical coordinates of the ICR to develop the kinematic model. In the study, the authors implement repulsive potential field at singular kinematic configurations and attractive ones at the target ICR point. Maximum acceleration limits were taken into account, however handling joint position limits is ambiguous although stated as a critical problem in the ExoMars rover structure.

The authors in [LNL+06a, LNL+06b, CLM10] have developed three versions of the SWMR named Azimut1-3 (showin in Fig. 2.3), where the ICR space is divided into 7 zones with 7 modes of operation. In [CBC+10], the authors have developed a motion planner based on rapidly exploring random trees algorithm in order to find robot paths avoiding steer joint limits/restrictions.

In [SN11], steering angle and drive rate constraints are developed based on the information of path geometry to be followed and on the current robot velocity. Then, a path tracking controller is developed to take into account the same information in addition to such constraints. However, critical problems as singularity avoidance/handling and the maximum joint performance limits are not tackled in that study.

The team at the Institute of Robotics and Mechatronics, DLR, has developed the dual arm research platform for mobile and dexterous manipulation named Rollin' Justin, shown in Fig. 2.4. It employs a unique steerable wheeled mobile base with movable legs. The kinematic model and control of both the conventional SWMR,

Figure 2.4: Rollin' Justin variable footprint SWMR. The dual arm mobile manipulation system [BOW+07, FBG+09] (left), legged SWMR extended [GFASH09] (middle) and folded (right)

and the additional movable leg structures has been introduced in [GFASH09] using the $3D$ Cartesian robot velocity formulation. However, this work clearly avoided the kinematic singularity case. In [DWASH11], the same team switched to an ICR based controller, and tackled the problem of kinematic singularity by constraining the ICR space, applying a repulsive potential field around the steering axes.

In [GH14], a motion controller based on static input-output linearization is developed and experimentally tested on the 2 wheeled SWMR shown in Fig. 2.5a. In this work, however, both kinematic singularity and maximum steering performance limits were neglected. In [SMG15], robustness against kinematic singularity is enhanced by developing a second order model with regular parameterization depending on the robot path. The downside of such controller, implemented on the robot shown in Fig. 2.5b, are the assumptions that the robot will not stop at singular configuration, and that the steer joint velocity/acceleration limits are negligible.

Two more wheeled steerable robot named HERMIES-III (shown in Fig. 2.6a) have been developed in [WBE+90]. Their control was introduced in [RU93]. In that work, constraint force control has been introduced and was claimed to improve the odometry error by a factor of more than 20 times. Again, handling singularities nor actuation maximum limits was considered. In [TNM96], kinematic singularity has been treated as a non-allowed constraint to robot motion. Instead, in [BC96], representational singularities have been treated by switching between the choice of the two master wheels used in computing the kinematic model. The other wheels are steered depending on the master wheels. In both [TNM96, BC96], the maximum joint performance limit is not taken into account.

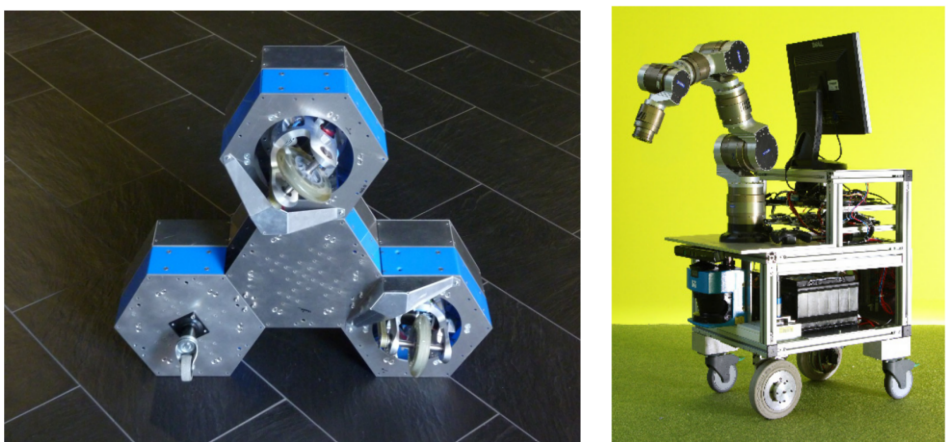
A backstepping controller is used in the field steerable mobile robots in [CLTB09] together with predictive control to account for time delays. The developed kinematic model accounts for sliding effects. The mobile robot, shown in Fig. 2.7, weighs 600Kg, and in such case, respecting the maximum steering limit or even respecting the wheel lateral skidding constraint can be completely out of question. The same applies for the 6 wheeled field SWMR of [MF00], shown in Fig. 2.6b. Nevertheless, these SWMR are provided here for the sake of completeness, and to show the versatility of applications for such mobile robot structure.

2.2 Gaps in the Literature

Problems with SWMR can be summarized as follows:

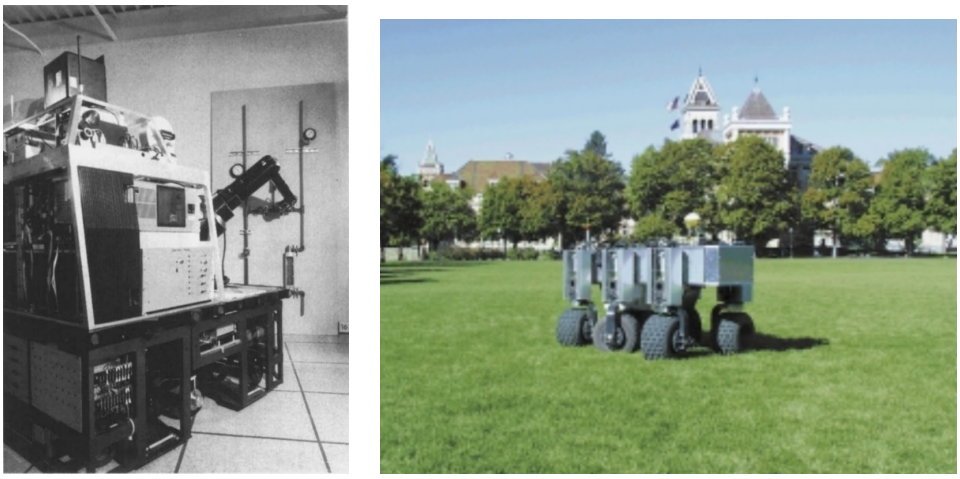
- representation singularity,
- kinematic singularity,
- respecting maximum steering limits,
- handling discontinuity in velocity commands.

Representation singularity is present in all methods for developing the mathematical (kinematic) model. In [CPHV08, CPHV09], attempts were made to develop a representation singularity-free model (no matter how complex its formulation) based on the ICR point formalism. This turned out to have the same problem later in [CHV12] requiring switching between different models. The same policy was adopted in [BC96]



(a) (b)

Figure 2.5: The experimental 2 wheeled SWMR used in [GH14] (a), and (b) another 2 wheeled steerable mobile manipulator used in [SMG15].



(a) (b)

Figure 2.6: HERMIES-III robot, 2 wheeled SWMR used in [RU93] (a), and (b) six wheeled field SWMR of [MF00].

using two master wheels in developing the kinematic model, and switching between each pair of wheels depending on the current steer configuration.

Kinematic singularity usually refers to the case where the ICR point approaches (and eventually coincides with) one of the steering axes. At such steer configuration, the singular steer joint value is undefined, and usually the velocity output by the kinematic model for such joint is unbounded. This configuration has been treated as a restricted area in the $3D$ robot velocity space in [DWASH11, CPHV09, SPS12] by applying repulsive potential fields around the steering axes, or as a constraint to robot motion in [TNM96]. Obviously these methods limit the robot maneuverability. In [OGM13, OGM14], the velocity and acceleration of the singular steer joint were bounded which makes use of the full velocity space of the robot, but shortens the hardware lifespan by allowing the steer joints to work at maximum limit during such

Figure 2.7: Field Steerable mobile vehicle used in [CLTB09].

configurations.

Discontinuity in robot velocity commands is a new challenge for SWMR due to the nature of the application in hand, where human worker and robot are sharing the same workspace. Unfortunately, the current state of art will fail to provide proper solutions.

2.3 Thesis Contributions

In this thesis, given a SWMR with maximum steering velocity and acceleration bounds, an algorithmic controller is designed to allow following - as faithfully as possible - a desired non-smooth (discontinuous), $3D$ Cartesian velocity trajectory assuming the no-slip, no-skid kinematic constraints are well respected, while being robust against kinematic and representation singularities. Our findings has been validated experimentally on an industrial mobile base. In what follows, we give a short summary of the main contributions of the thesis, that should cover the gaps presented above. Details are presented in the respective chapters.

2.3.1 Singularity-free $3D$ Cartesian Space Model

The problem is usually solved either by taking into account the representational and kinematic singularities of this kind of mobile robot structure at either the motion (trajectory) planning phase or the control phase. In Chapter 3, as a first contribution we propose a solution in the kinematic modeling phase. It is known that kinematic model using the $3D$ Cartesian space robot velocity formalism employing 3 or more steerable wheels is free from representation singularity [DWASH11]. We then provide numeric treatment for the kinematic singularity by adding a damping term to the denominator of such kinematic model, providing an opposite behavior to the singular one, where the steering velocity slows down to zero at kinematic singularity.

2.3.2 Discontinuity Robust ICR Based Controller

To the best of the authors knowledge, no thorough investigation has been conducted on the issue of reorienting the wheels, when discontinuities occur in the robot trajectory. Usually, steer reconfiguration is performed in a *manual* fashion, depending on the test trajectory, and the initial condition is set in an initial phase. This is found

in the literature under various designations, e.g. "*initial phase*" in [GFASH09], "*open-loop starting procedure*" in [TNM96], and "perfect matching condition" in [GFASH09]. Although an ICR-based controller is the most suited to handle such cases, the work in [OGM13] and [OGM14] is limited by the assumption of continuous and differentiable desired signal, whereas in [SPS12] 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 art methods will fail to provide proper solutions. In chapter 4, 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).

2.3.3 Complementary ICR Route

In applications requiring that some heading angle is to be maintained, that some object or feature stays in the field of view (e.g., for vision-based tasks), or that the translation verse changes, the ICR point is required to move long distances from one extreme of the workspace to the other, usually passing by the robot geometric center, where the feasible robot velocity is limited. In such scenarios, the state-of-art ICR based controllers will lead to unsatisfactory behavior/results.

To solve this problem, in chapter 5 we propose a comparison between the direct and complementary ICR routes, the former (state-of-art approach) being the shortest straight line connecting the current and desired ICR points, while the latter (proposed here) connecting the ICR extreme borders via a border ICR point that is chosen to minimize the total ICR distance moved across borders. The 4 borders define the maximum values that the ICR point is allowed to take on the x and y axes of the geometrically centered robot frame. Instead of moving directly to the desired ICR,

the complementary route will move first to the optimum border point lying on the nearest border line, switch borders, *for example: from the $+y$ to the $-y$ border line* in one sample period, then move to the desired ICR point.

Chapter 3

Steerable Wheeled Robot Kinematic Modeling

Contents

3.1 Introduction 17
3.2 Kinematic Modeling 18
3.2.1 Cartesian Space Model Formulation 18
3.2.2 Odometry Model 21
3.2.3 $2D$ ICR Space to $3D$ Cartesian Space Mapping 21
3.3 Kinematic Singularity Treatment 22
3.4 Benchmark Test Trajectory 24
3.4.1 Parabolic ICR position profile 24
3.4.2 ICR point at kinematic singularity 27
3.4.3 Pure linear motion 27
3.4.4 Straight line motion of the ICR between steering axes . . . 27
3.4.5 Zero Velocity 28
3.5 Experiments and Results 28
3.6 Drawbacks 31
3.7 Conclusion 33
3.8 Link with Next Chapter 33

3.1 Introduction

This chapter presents the kinematic model of steerable wheeled mobile robots (SWMR). Section 3.2.1 gives the $3D$ Cartesian space model formulation, that outputs the inverse actuation kinematic model (IAKM) giving the joint space velocities (steer and drive)

in function of the task space velocities. Then, the forward kinematics model (used for robot odometry) is formulated in Section 3.2.2. In Section 3.3, we propose a numeric treatment of the kinematic model, that robustifies it with regards to kinematic singularities. In Section 3.4, a benchmark test trajectory is proposed to evaluate the numerically treated kinematic model against all kinds of singularities. Experiments are carried out on the industrial mobile robot Neobotix-MPO700 and results of the proposed treatment on the benchmark trajectory are provided in Section 3.5. In the same section, we compare the performance of the Neobotix-MPO700 embedded model/controller against the same benchmark. Conclusions are finally given in Section 3.7.

3.2 Kinematic Modeling

The kinematic model presented hereby is inspired by the pioneer work of Muir [MN87b], Campion [CBN96], Betourne [BC96] and Low *et al.* [LL05]. The schematic of a SWMR is shown in Fig. 3.1 for a 4 wheeled robot. However, the model is generic for SWMR with number of wheels $N \geq 3$.

3.2.1 Cartesian Space Model Formulation

Let $\boldsymbol{\mathcal{F}}_I = (o_I \mid \boldsymbol{x}_I, \boldsymbol{y}_I, \boldsymbol{z}_I)$ be the inertial frame, $\boldsymbol{\mathcal{F}}_b = (o_b \mid \boldsymbol{x}_b, \boldsymbol{y}_b, \boldsymbol{z}_b)$ the mobile base frame, with origin o_b located at the base geometric center (the mid point between all steering axes), $\boldsymbol{\mathcal{F}}_{hi} = (o_{hi} \mid \boldsymbol{x}_{hi}, \boldsymbol{y}_{hi}, \boldsymbol{z}_{hi})$ the i^{th} hip frame ($i = 1, \ldots, N$), attached to the fixed part of the steering joint, and related to the base frame by a fixed transformation matrix, and $\boldsymbol{\mathcal{F}}_{si} = (o_{si} \mid \boldsymbol{x}_{si}, \boldsymbol{y}_{si}, \boldsymbol{z}_{si})$ the steering frame, attached to the movable part. The hip and steering frames share the same origin, with relative orientation β_i (the steering angle). Frame $\boldsymbol{\mathcal{F}}_{wi} = (o_{wi} \mid \boldsymbol{x}_{wi}, \boldsymbol{y}_{wi}, \boldsymbol{z}_{wi})$ is attached to (but not rotating with) the i^{th} wheel, assigned such that $\boldsymbol{x}_{wi}$ points along the heading of the wheel, which rotates about $\boldsymbol{y}_{wi}$ by the driving angle ϕ_i. All the frames have the $\boldsymbol{z}$ axis pointing upwards. Let the mobile base pose w.r.t. the inertial frame define the $3D$ task space coordinates $\boldsymbol{\xi} = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$. A left superscript is added to indicate the frame in which the pose is expressed, for instance ${}^I\boldsymbol{\xi}$ and ${}^b\boldsymbol{\xi}$ 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 $\boldsymbol{v}_{ci} = \begin{bmatrix} v_{ti} & v_{ni} & 0 \end{bmatrix}^T$ at the ground contact point o_{ci} (expressed in the wheel frame $\boldsymbol{\mathcal{F}}_{wi}$), with v_{ti} and v_{ni} respectively the i^{th} tangential

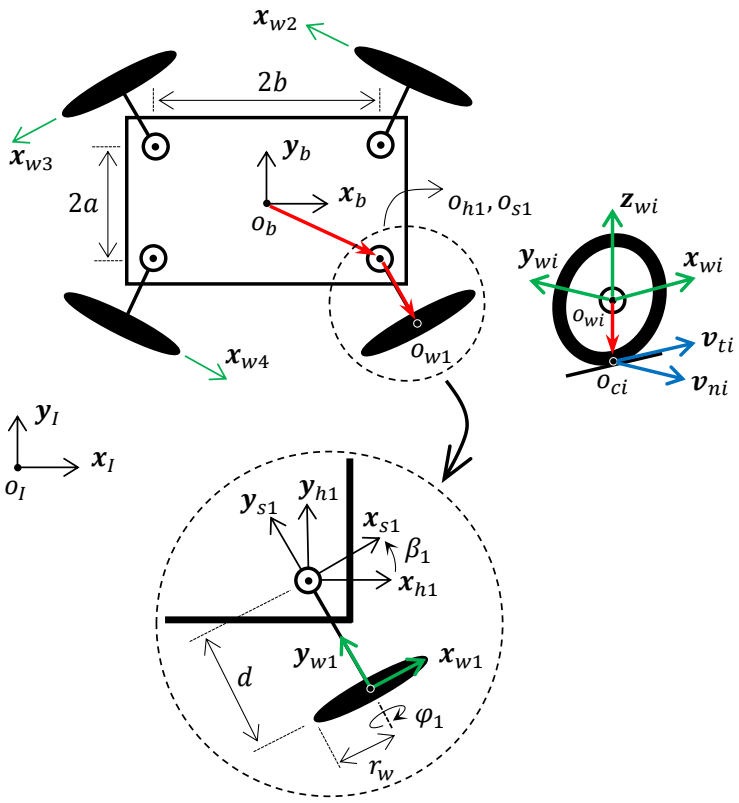


Figure 3.1: Schematic model of a four wheeled SWMR.

and normal velocities, we can obtain:

$$\boldsymbol{v}_{ci} = \dot{\boldsymbol{\xi}} + \dot{\theta}\hat{\boldsymbol{k}}_b \times \overrightarrow{o_b o_{ci}} + \dot{\beta}_i \hat{\boldsymbol{k}}_{si} \times \overrightarrow{o_{si} o_{ci}} + \dot{\phi}_i \hat{\boldsymbol{j}}_{wi} \times \overrightarrow{o_{wi} o_{ci}}, \tag{3.1}$$

where $\hat{\boldsymbol{k}}_b$, $\hat{\boldsymbol{k}}_{si}$, and $\hat{\boldsymbol{j}}_{wi}$ are unit vectors along the $\boldsymbol{z}_b$, $\boldsymbol{z}_{si}$, and $\boldsymbol{y}_{wi}$ axes respectively. Equation (3.1) is then expressed in the i^{th} wheel frame $\boldsymbol{\mathcal{F}}_{wi}$ as follows:

$$\boldsymbol{v}_{ci} = {}^{wi}\boldsymbol{R}_b(\dot{\boldsymbol{\xi}} + \dot{\theta}\hat{\boldsymbol{k}}_b \times {}^{b}\overrightarrow{o_b o_{ci}} + \dot{\beta}_i {}^{b}\boldsymbol{R}_{si}\hat{\boldsymbol{k}}_{si} \times {}^{si}\overrightarrow{o_{si} o_{ci}}) + \dot{\phi}_i \hat{\boldsymbol{j}}_{wi} \times {}^{wi}\overrightarrow{o_{wi} o_{ci}}, \tag{3.2}$$

where ${}^{wi}\boldsymbol{R}_b = rot(\boldsymbol{z}_b, -\beta_i)$ and ${}^{b}\boldsymbol{R}_{si} = rot(\boldsymbol{z}_b, \beta_i)$ are $3D$ rotation matrices representing the orientation of $\boldsymbol{\mathcal{F}}_b$ with respect to (wrt) $\boldsymbol{\mathcal{F}}_{wi}$, and $\boldsymbol{\mathcal{F}}_{si}$ wrt $\boldsymbol{\mathcal{F}}_b$ respectively, ${}^{wi}\boldsymbol{R}_b, {}^{b}\boldsymbol{R}_{si} \in SO(3)$, ${}^{b}\overrightarrow{o_b o_{ci}}$, ${}^{si}\overrightarrow{o_{si} o_{ci}}$, and ${}^{wi}\overrightarrow{o_{wi} o_{ci}}$ are 3×1 position vectors (colored red in Fig. 3.1) detailed as follows:

$${}^{wi}\overrightarrow{o_{wi} o_{ci}} = \begin{bmatrix} 0 \\ 0 \\ -r_w \end{bmatrix},$$

$${}^{si}\overrightarrow{o_{si} o_{ci}} = {}^{si}\overrightarrow{o_{si} o_{wi}} + {}^{si}\boldsymbol{R}_{wi}\,{}^{wi}\overrightarrow{o_{wi} o_{ci}} = \begin{bmatrix} 0 \\ -d \\ -r_w \end{bmatrix},$$

$$ {}^{b}o_b\overrightarrow{o}_{ci} = {}^{b}o_b\overrightarrow{o}_{hi} + {}^{b}\boldsymbol{R}_{si}\,{}^{si}o_{si}\overrightarrow{o}_{ci} = \begin{bmatrix} h_{xi} + d\sin(\beta_i) \\ h_{yi} - d\cos(\beta_i) \\ -r_w \end{bmatrix}. $$

In the above equations, $h_{xi} = \pm b$ and $h_{yi} = \pm a$ denote the position of the i^{th} hip frame origin o_{hi} in the base frame $\boldsymbol{\mathcal{F}}_b$, $a, b \in \mathbb{R}^+$. Parameters d and r_w are respectively the steerable wheel offset and wheel radius. Substituting by these position vectors in (3.2) we get the following expressions for the contact point velocities:

$$ \begin{aligned} v_{ti} &= \boldsymbol{f}(\beta_i)\dot{\boldsymbol{\xi}} + d\dot{\beta}_i - r_w\dot{\phi}_i, \\ \boldsymbol{f}(\beta_i) &= \begin{bmatrix} \cos(\beta_i) & \sin(\beta_i) & d - h_{yi}\cos(\beta_i) + h_{xi}\sin(\beta_i) \end{bmatrix}, \end{aligned} \tag{3.3} $$

$$ \begin{aligned} v_{ni} &= \boldsymbol{g}(\beta_i)\dot{\boldsymbol{\xi}}, \\ \boldsymbol{g}(\beta_i) &= \begin{bmatrix} -\sin(\beta_i) & \cos(\beta_i) & h_{xi}\cos(\beta_i) + h_{yi}\sin(\beta_i) \end{bmatrix}. \end{aligned} \tag{3.4} $$

Setting $v_{ti} = 0$ and $v_{ni} = 0$, (3.3) and (3.4) represent the *rolling with no slipping* and the *no lateral skidding* kinematic constraint assumptions, 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 $\boldsymbol{G}(\boldsymbol{\beta})$ that will be used in the sequel as:

$$ \boldsymbol{G}(\boldsymbol{\beta})_{N\times 3} = \begin{bmatrix} \boldsymbol{g}(\beta_1)^T & \dots & \boldsymbol{g}(\beta_N)^T \end{bmatrix}^T. \tag{3.5} $$

For joint position controlled robots, the steering angles corresponding to Cartesian velocity commands can be evaluated using (3.4) when $v_{ni} = 0$ as:

$$ \beta_i = \arctan\Big(\frac{\dot{y} + h_{xi}\dot{\theta}}{\dot{x} - h_{yi}\dot{\theta}}\Big). \tag{3.6} $$

For joint rate controlled robots, (3.4) should differentiated w.r.t time, and then rearranged to evaluate the required steer joint rate when $v_{ni} = 0$ as:

$$ \dot{\beta}_i = \frac{-\boldsymbol{g}(\beta_i)\ddot{\boldsymbol{\xi}}}{\frac{d\boldsymbol{g}(\beta_i)}{d(\beta_i)}\dot{\boldsymbol{\xi}}}. \tag{3.7} $$

Then, the wheel drive rate is obtained from (3.3), when $v_{ti} = 0$:

$$ \dot{\phi}_i = \frac{1}{r_w}\boldsymbol{f}(\beta_i)\dot{\boldsymbol{\xi}} + \frac{d}{r_w}\dot{\beta}_i. \tag{3.8} $$

Equations (3.7) and (3.8) represent the SWMR inverse actuation kinematic model (IAKM).

3.2.2 Odometry Model

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). This can be written as:

$$\hat{\dot{\boldsymbol{\xi}}} = \boldsymbol{F}^{+}_{(d)}(\hat{\boldsymbol{\beta}})(r_w\hat{\dot{\boldsymbol{\phi}}} - d\hat{\dot{\boldsymbol{\beta}}}), \tag{3.9}$$

$$\boldsymbol{F}(\hat{\boldsymbol{\beta}})_{N\times 3} = \begin{bmatrix}\boldsymbol{f}(\hat{\beta}_1)^T & \dots & \boldsymbol{f}(\hat{\beta}_N)^T\end{bmatrix}^T,$$

where the hat symbol $\hat{}$ indicates a measurement value and $\boldsymbol{F}^{+}_{(d)}(*)$ denotes the damped pseudo-inverse of $\boldsymbol{F}(*)$, evaluated using [NH86] and [Wam86]:

$$\boldsymbol{F}^{+}_{(d)}(*) = (\boldsymbol{F}^T(*)\boldsymbol{F}(*) + \delta^2\boldsymbol{I}_{3\times 3})^{-1}\boldsymbol{F}^T(*).$$

Parameter $\delta \in \mathbb{R}^*$ is 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) since, in such case, $\boldsymbol{F}(\hat{\boldsymbol{\beta}})$ loses rank.

3.2.3 $2D$ ICR Space to $3D$ Cartesian Space Mapping

The conversion from the $2D$ ICR point space to the $3D$ Cartesian velocity space (and vice versa), will be used in the experiments, here we provide such relation. We use the laws of rigid body mechanics. Recall that an object rotating at an angular velocity $\boldsymbol{\omega}$ about a fixed ICR point located at a distance $\boldsymbol{r}$, will have the linear velocity:

$$\boldsymbol{v} = \boldsymbol{\omega} \times \boldsymbol{r}, \tag{3.10}$$

where,

$$\boldsymbol{v} = \begin{bmatrix}\dot{x}\\ \dot{y}\\ 0\end{bmatrix}, \quad \boldsymbol{\omega} = \begin{bmatrix}0\\ 0\\ \dot{\theta}\end{bmatrix}, \quad \boldsymbol{r} = \begin{bmatrix}X_{ICR}\\ Y_{ICR}\\ 0\end{bmatrix}.$$

Using (3.10), we can easily deduce:

$$\begin{aligned}\dot{x} &= -Y_{ICR}\,\dot{\theta},\\ \dot{y} &= X_{ICR}\,\dot{\theta}.\end{aligned} \tag{3.11}$$

The relation in (3.11) is non-injective, where we still need one more information to retrieve the Cartesian robot velocity corresponding to a given ICR point, where the robot can rotate about such point at any arbitrary angular velocity $\dot{\theta}$. We use (3.11) in the experiments in this chapter and the next one only, where we know the angular

velocity command sent to the robot. In the developed controllers in chapters 4, and 5, we make sure not to use (3.11). The opposite (injective) relation is biven by:

$$X_{ICR} = \frac{-\dot{y}}{\dot{\theta}}, \quad Y_{ICR} = \frac{\dot{x}}{\dot{\theta}}, \tag{3.12}$$

which is singular for $\dot{\theta} = 0$.

3.3 Kinematic Singularity Treatment

Singularities related to this kind of mobile robots are divided into two categories: kinematic and representational. While the latter (for 3 or more steerable WMR) is solved using the Cartesian coordinates of the base frame [DWASH11], the former still presents a challenging problem. Kinematic singularity refers usually to the case where the ICR passes by (or near) any of the steering axes. In such case, the steer angle β_i in not uniquely defined, as there exist infinite values respecting the kinematic constraints. While passing this configuration, the steering rate shall grow unbounded from the "nearby" to the "at" singular configuration. This is shown mathematically in (3.6), where the denominator of the argument of the arctan() function decreases to zero at this singularity.

In our work, we fix this problem with a very simple, yet effective, numerical treatment, by adding the damping parameters $\delta_1 \in \mathbb{R}^+$ and $\delta_2 \in \mathbb{R}$ to the denominator of (3.6), and (3.7) respectively. This is explained hereby. For the steering rate, we use a formulation alternative to (3.7) by partial differentiation of (3.6). Then, the damping term is added, while evaluating the partial derivatives:

$$\begin{gathered} \dot{\beta}_i = \frac{\partial \beta_i}{\partial \dot{x}}\ddot{x} + \frac{\partial \beta_i}{\partial \dot{y}}\ddot{y} + \frac{\partial \beta_i}{\partial \dot{\theta}}\ddot{\theta}, \\ \frac{\partial \beta_i}{\partial \dot{x}} = \frac{-\tilde{y}_i}{\tilde{x}_i^2 + \tilde{y}_i^2 + \delta_2}, \quad \frac{\partial \beta_i}{\partial \dot{y}} = \frac{\tilde{x}_i}{\tilde{x}_i^2 + \tilde{y}_i^2 + \delta_2}, \quad \frac{\partial \beta_i}{\partial \dot{\theta}} = \frac{h_{xi}\dot{x} + h_{yi}\dot{y}}{\tilde{x}_i^2 + \tilde{y}_i^2 + \delta_2}, \end{gathered} \tag{3.13}$$

with $\tilde{x}_i = \dot{x} - h_{yi}\dot{\theta}$, $\tilde{y}_i = \dot{y} + h_{xi}\dot{\theta}$. The steering angle is then evaluated as:

$$\beta_i = \arctan\Big(\frac{\tilde{y}_i}{\tilde{x}_i + \delta_1 sign(\tilde{x}_i)}\Big). \tag{3.14}$$

The value of δ_1 can be chosen arbitrarily small. Instead, δ_2 must respect the acceleration limits of the steer joints (in case of centered WMRs) or of the steer/drive joints (in case of off-centered WMRs). Such effect is shown in Fig. 3.2. Once tuned, δ_2 provides a damping effect that is completely opposite to the singular behavior. Using (3.13) the steering rate will decrease to zero from the "nearby" to the "at"

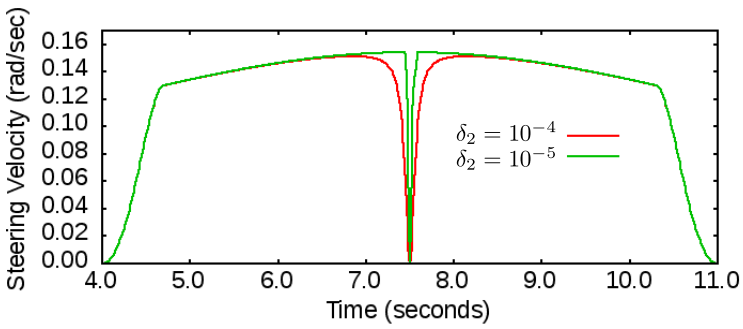


Figure 3.2: Effect of different values of δ_2 on the steering velocity while passing kinematic singularity. The lower the value, the more the required acceleration.

singular configuration. To tune δ_2 in order to respect the steer acceleration limits, we differentiate (3.13) with respect to time, neglecting higher order derivatives, to obtain:

$$\ddot{\beta}_i = \frac{k_i(\dot{\xi}, \ddot{\xi})}{(\tilde{x}_i^2 + \tilde{y}_i^2 + \delta_2)^2}, \tag{3.15}$$

where

$$k_i(\dot{\xi}, \ddot{\xi}) = 2(\tilde{y}_i\dot{\tilde{x}}_i - \tilde{x}_i\dot{\tilde{y}}_i)(\tilde{x}_i\dot{\tilde{x}}_i + \tilde{y}_i\dot{\tilde{y}}_i) \tag{3.16}$$

is a function of the robot trajectory $\dot{\xi}, \ddot{\xi}$, that is assumed to be continuous.

Proposition 3.3.1. *Using (3.15) we define a candidate value for δ_2 (called δ_{2c}) as:*

$$\delta_{2c} = \sqrt{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}}} - (\tilde{x}_i^2 + \tilde{y}_i^2), \tag{3.17}$$

where $\ddot{\beta}_{i(max)}$ is the i^{th} steer joint acceleration limit in centered steerable robots, or the smallest acceleration limit of the steer/drive joint in off-centered steerable robots (since in this robot structure, steering requires moving the corresponding wheel as well, see (3.8)) and $|*|$ is the Euclidean norm. Although the value obtained in (3.17) depends on the trajectory, to respect the acceleration limits at all times, we can conservatively use, for δ_2, the maximum value over all possible trajectories in a bounded domain around the singularity. In applications involving human-robot collaboration, the trajectory is generally unknown a priori. For such applications, we therefore need to evaluate the maximum δ_{2c} online. To this end, we define the following max() function:

$$\max(x_k) = \begin{cases} x_k, & \text{if } x_k > x_{k-1} \\ x_{k-1}, & \text{otherwise} \end{cases},$$

with $(*)_k$ and $(*)_{k-1}$ the values at current and previous sampling instances, respectively. The max() function will store the input argument with highest magnitude. Furthermore, (3.17) was shown (empirically) to evaluate to a positive value in the neighborhood of kinematic singularity. In this region of interest, where respecting the acceleration limits becomes critical, we set

$$\delta_2 = \max(\delta_{2c}). \tag{3.18}$$

Let us point out that this is the minimum value that can be used in (3.13), while respecting the joint acceleration limit. Indeed, we need the minimum value to obtain the highest accuracy in evaluating the steer rate command. Finally, the online evaluation of δ_2 is done using:

$$\delta_2 = \begin{cases} \delta_1, & \forall \delta_{2c} \leq 0 \\ \max(\delta_{2c}), & \forall \delta_{2c} > 0 \end{cases}. \tag{3.19}$$

With (3.19), the smallest δ_2 value can be obtained online and set only in the vicinity of singular configurations at which the acceleration limits are to be monitored. Away from the critical zone (around singularity), this formula provides negligibly small value (δ_1) for δ_2, and consequently much higher accuracy for (3.13). The proof that (3.19) guarantees respecting the joint acceleration limits is provided in Appendix A.

3.4 Benchmark Test Trajectory

In this section, we propose a set of Cartesian velocity trajectories (commands) to benchmark test the model and controller performances w.r.t all representational and kinematic singularities for this mobile robot structure. The test is composed of five distinct motion profiles, all defined in the robot frame and depicted in Fig. 3.3. In what follows, we detail each one of them. Since each set of velocity commands requires a different initial steering configuration, there is a transition phase between each two successive sets of commands, during which the required initial steering reconfiguration is realized.

3.4.1 Parabolic ICR position profile

This first test excites steer joint motion in the vicinity of kinematic singularity, to check if it will respect the joint limits. It comprises a set of Cartesian velocity commands corresponding to a parabolic ICR-point motion profile with vertex at one of

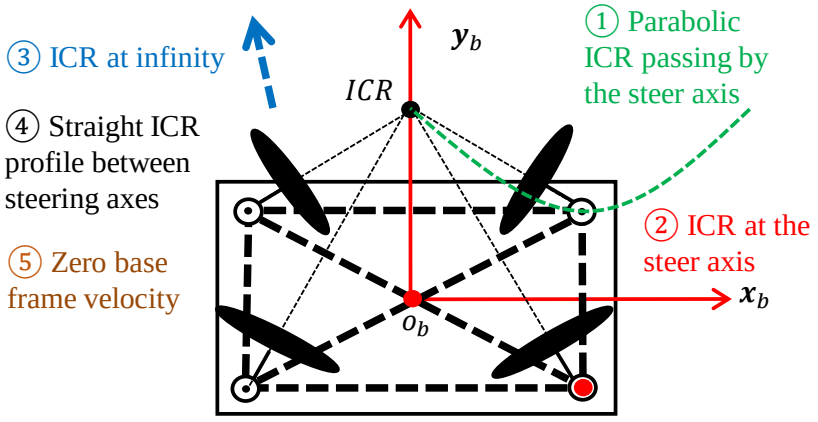


Figure 3.3: Schematic model of the 5 benchmark test trajectories.

the steering axes. Note that a straight line motion would impose no motion on the steering joint.

The parabolic ICR path shown in Fig. 3.3 (in green) has its x-axis component X_{ICR} ranging from 0 to $2h_{x2}$. We convert it to a motion profile by assigning it the following time law:

$$X_{ICR} = T_{l5b}(X_{ICR(i)} = 0,\ X_{ICR(f)} = 2h_{x2},\ t_i,\ t_f,\ t), \tag{3.20}$$

In (3.20), T_{l5b} is an online - linear with 5^{th} order blends - trajectory generator function, taking the initial position $X_{ICR(i)}$, the final position $X_{ICR(f)}$, the initial trajectory time t_i, the final trajectory time t_f, and the current time instant t as arguments. It is shown in Fig. 3.4, computed as:

$$T_{l5b} = \begin{cases} a_0 + a_1\delta t_1 + a_2\delta t_1^2 + a_3\delta t_1^3 + a_4\delta t_1^4 + a_5\delta t_1^5, & t_i \le t < t_1 \\ a_6 + a_7\delta t_2, & t_1 \le t < t_2 \\ a_0 + a_1\delta t_3 + a_2\delta t_3^2 + a_3\delta t_3^3 + a_4\delta t_3^4 + a_5\delta t_3^5, & t_2 \le t < t_f. \end{cases}$$

where,

$$\begin{aligned} \delta t_1 &= t - t_i, \\ \delta t_2 &= t - t_1, \\ \delta t_3 &= t - t_2, \\ t_1 &= t_i + 0.1 * (t_f - t_i), \\ t_2 &= t_f - 0.1 * (t_f - t_i). \end{aligned}$$

Coefficients a_0 ... a_7, are computed by setting the initial and final positions to $X_{ICR(i)}$ and $X_{ICR(f)}$ respectively, and the velocity/acceleration/jerk *(boundary conditions)* to zero. The y-axis component of the ICR motion profile is then computed

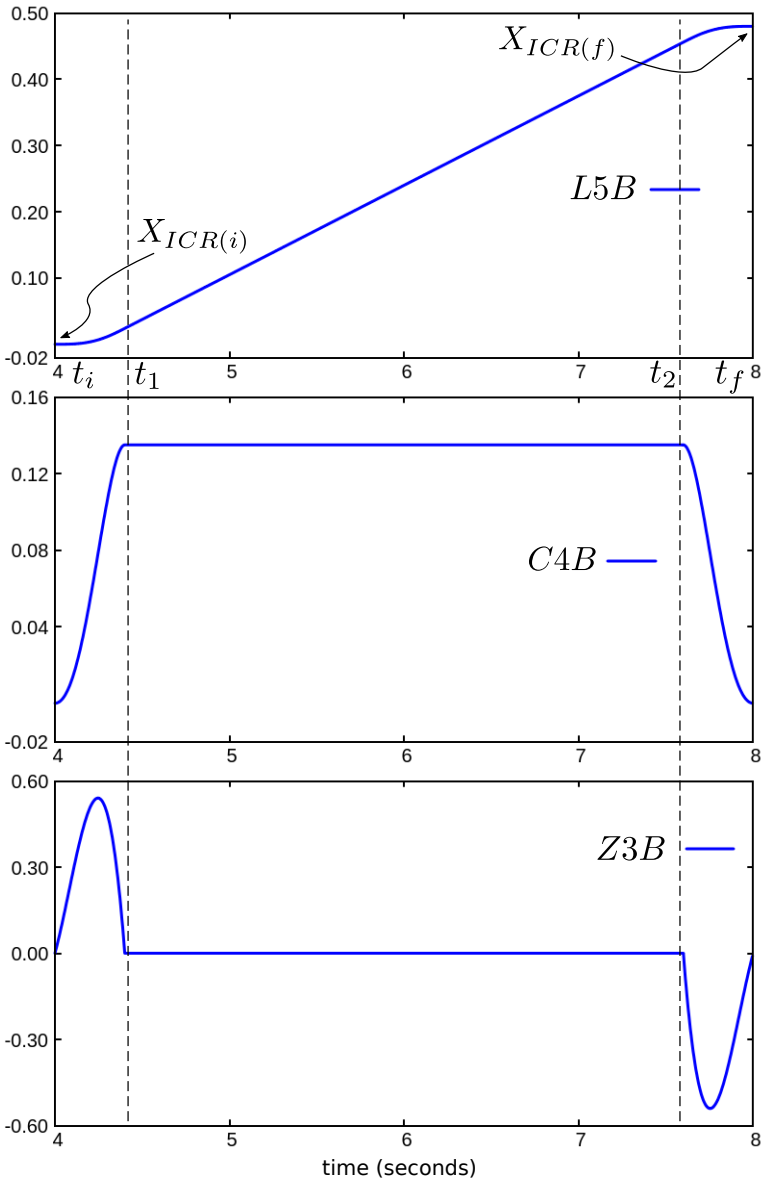


Figure 3.4: Linear with fifth order blends ICR point motion trajectory in first experiment (upper), its first derivative being a constant with fourth order blends (middle), and the second derivative (lower).

using:

$$Y_{ICR} = (X_{ICR} + h_{x2})^2 + h_{y2}. \tag{3.21}$$

With (3.21), (X_{ICR}, Y_{ICR}) is a parabola with vertex located at the 2^{nd} steer joint axis (kinematic singularity). From (3.11), (3.4.1), and (3.21), by setting $\dot{\theta} = -1$, we finally obtain the Cartesian velocity commands for this test:

$$\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} -(\dot{y} - h_{x2})^2 - h_{y2} \\ T_{l5b}(\dot{y}_i = 0, \dot{y}_f = 2h_{x2}, t_i, t_f, t) \\ -1 \end{bmatrix}. \tag{3.22}$$

3.4.2 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.

The following velocity command corresponds to an ICR point at the second steering axis (red in Fig. 3.3):

$$\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} h_{y2} & -h_{x2} & 1 \end{bmatrix}^T. \tag{3.23}$$

3.4.3 Pure linear motion

This trajectory is needed to test robustness against representational singularity in ICR-based models and controllers, that usually employ $\dot{\theta}$ in the denominator, by sending *"zero angular velocity"* and arbitrary linear velocity (v_{ar1}, v_{ar2}):

$$\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} v_{ar1} & v_{ar2} & 0 \end{bmatrix}^T. \tag{3.24}$$

3.4.4 Straight line motion of the ICR between steering axes

This test consists in moving the ICR along a straight line connecting each two steering axes *(i.e., six straight lines for a 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. The test is realized using the following command sequence:

$$\begin{aligned}
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y1} & -h_{x1} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y2} & -h_{x2} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y3} & -h_{x3} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y4} & -h_{x4} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y2} & -h_{x2} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y1} & -h_{x1} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y4} & -h_{x4} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y3} & -h_{x3} & 1 \end{bmatrix}^T, \\
\dot{\boldsymbol{\xi}}^* &= \begin{bmatrix} h_{y1} & -h_{x1} & 1 \end{bmatrix}^T.
\end{aligned} \tag{3.25}$$

3.4.5 Zero Velocity

This trajectory is used to test robustness against representation singularity in Cartesian velocity based models employing $\dot{x}$, $\dot{y}$, and $\dot{\theta}$ in the denominator of (3.7). We simply use:

$$\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T . \tag{3.26}$$

3.5 Experiments and Results

In this section, the benchmark test trajectory presented in 3.4 is used to evaluate the proposed singularity robust methods developed in 3.2.1 and 3.3. We also compare the performance of our method with that of the embedded controller/model. The experiments are carried out on the industrial mobile robot Neobotix-MPO700 (shown in Fig. 3.5). All parameters used in the experiments are provided in Table 3.1. A video of the experiment can be found on the IDH YouTube channel [1]. In this experiment, the 4^{th} benchmark test (3.25) is not implemented since the developed kinematic model uses the 4 wheels.

Figure 3.6 (top) shows the $3D$ Cartesian robot velocity commands corresponding to the benchmark trajectory, where the encircled numbers denote:

1. the parabolic ICR motion profile (3.22) during period $t = [4, 8[s$,
2. ICR point at kinematic singularity (3.23) during period $t = [12, 15[s$,
3. pure linear motion (3.24) during period $t = [18, 20[s$,
4. zero velocity command (3.26) during period $t = [20, 22[s$.

In Fig. 3.6 (middle, bottom), the corresponding steer joint space velocities and accelerations are provided. The curves clearly show that using (3.19) results in respecting the maximum steer acceleration limit. The initialization phase (denoted by the encircled letter I in Fig. 3.6, bottom) during periods $t = [0, 3[s$, $t = [9, 12[s$, and $t = [15, 18[s$, is dedicated to reconfiguring the steer joints, to meet the required initial conditions before implementing the following test. In the same figure, it can be seen that such phase corresponds to null robot velocity, but still results in some steering action as shown in Fig. 3.6 (middle, bottom). During intervals $t = [3, 4[s$ and $t = [8, 9[s$ (respectively) the robot velocity increases from 0 to the initial value of test 1, and decreases from the final value of test 1 to 0.

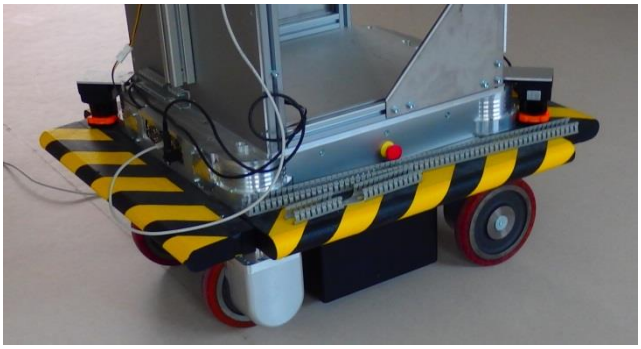


Figure 3.5: Photograph of the Neobotix MPO700 industrial SWMR.

The robot driver allows two command modes: joint velocity or robot velocity modes. We use the former to test the proposed model, while the latter (employing the embedded model/controller) is used in the comparative study. In a first experiment, the test trajectory shown in Fig. 3.6 (top) is sent in robot velocity mode, to evaluate the joint commands output by the embedded model. In a second experiment, the output of our inverse kinematic model shown in Fig. 3.6 (middle) is used to control the robot in joint velocity mode.

Figures 3.7 and 3.8 respectively show the results for the embedded and developed controllers. For test 1 (parabolic motion of the ICR), the embedded controller gives a velocity peak of the steering joint in the neighborhood of the kinematic singularity (green curve, at $6.3s$ on Fig. 3.7c). This corresponds to a steer angle change of 180° in a very short time, implying that the embedded model probably employs steer position, rather than steer velocity control. The corresponding wheel velocity response can be seen in Fig. 3.7d, and the effect of such behavior on the velocity error is evident in Fig. 3.7b. The Cartesian robot velocity estimate shown in 3.7a is obtained utilizing the odometry model presented in (3.9). The other steer velocity peaks in Fig. 3.7a are due to the fact that the embedded controller moves to each initial steer joint space configuration (corresponding to each test) as quickly as possible.

[1]https://www.youtube.com/watch?v=9RCOUp24Gvs

Table 3.1: Robot and model parameters used in the singularity treatment experiments.

a	$0.19m$	b	$0.24m$	d	$0.045m$
r_w	$0.09m$	δ	0.001	δ_1	1^{-9}
$\ddot{\beta}_{max}$	$5rad./s^2$	t_s	$25ms$		

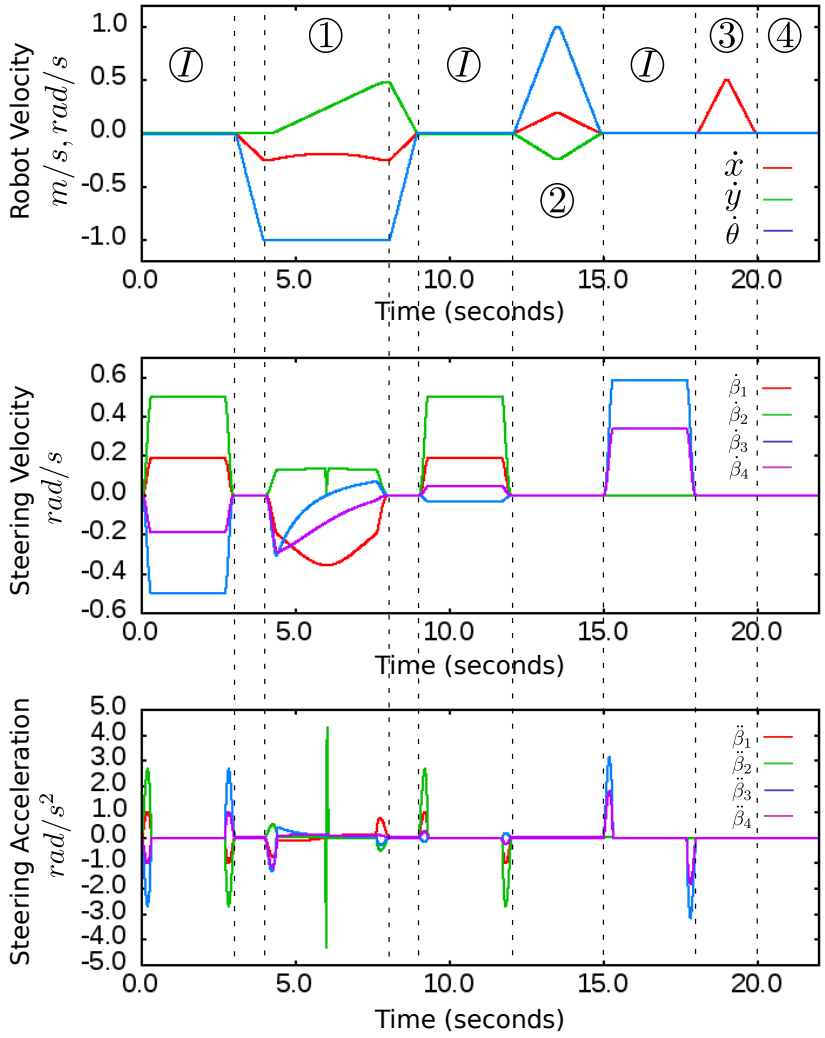


Figure 3.6: Output commands of the kinematic model employing the proposed singularity treatment, and excited with the benchmark test trajectory.

The effect of the proposed model on the steering joint velocity is shown in Fig. 3.8c. As shown, when approaching kinematic singularity, the velocity of the singular steer joint slows down to $0 rad/s$ at the kinematic singularity, with damped steer/drive rates in vicinity. Such effect is favorable, as it makes all the velocity space of the robot accessible, with minimum power consumption and with no risk of damage to motors performing at their top limits. Such damped behavior in the steering and driving (3.8d) results in much smaller velocity error w.r.t the desired trajectory as shown in Fig. 3.8a,b. Both models show satisfactory response in the 3 other tests.

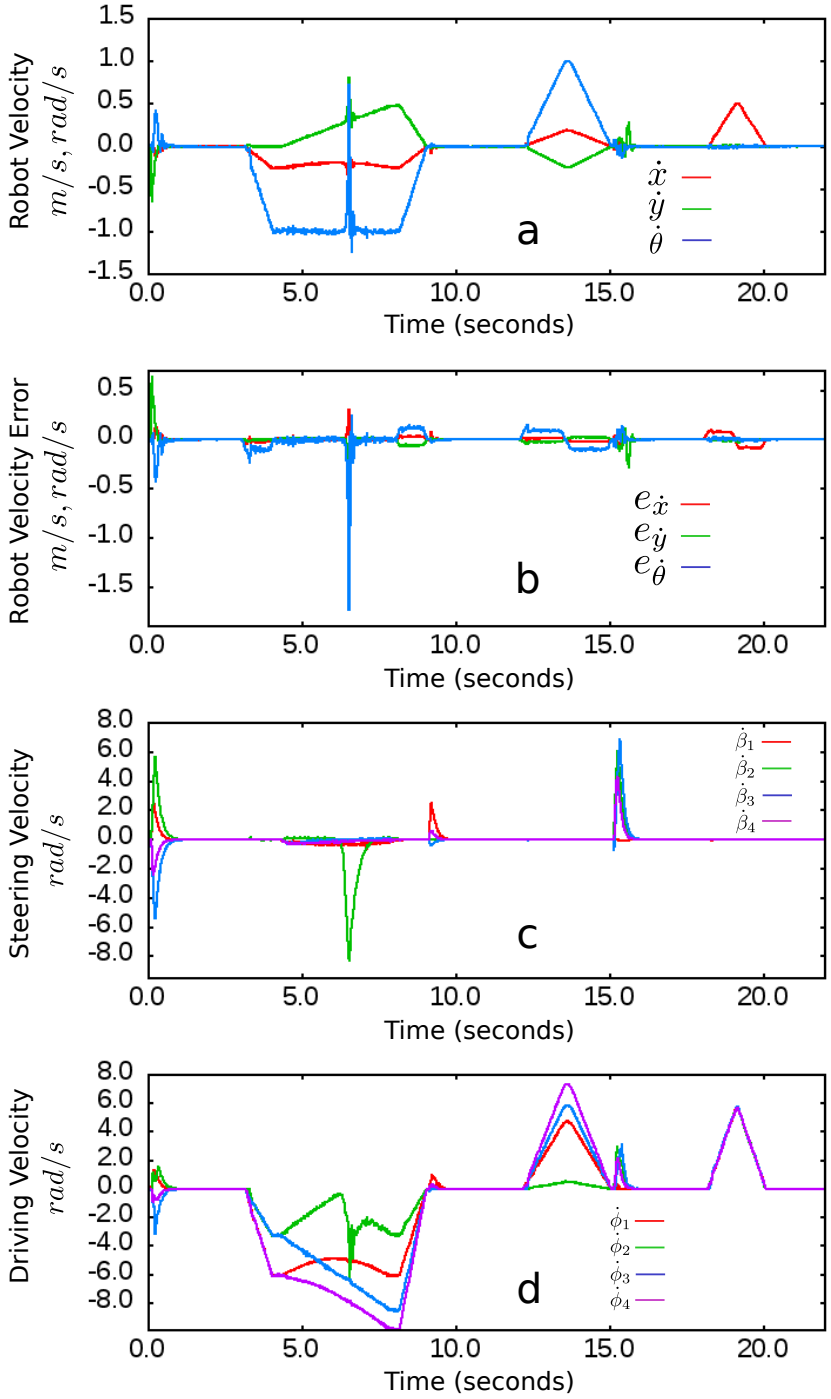


Figure 3.7: Measured response of the experiments employing the embedded controller, while executing the benchmark test trajectory.

3.6 Drawbacks

Although the numeric treatment for the kinematic singularity provides the desirable damping effect, it affects the accuracy of the steering commands. As such, after

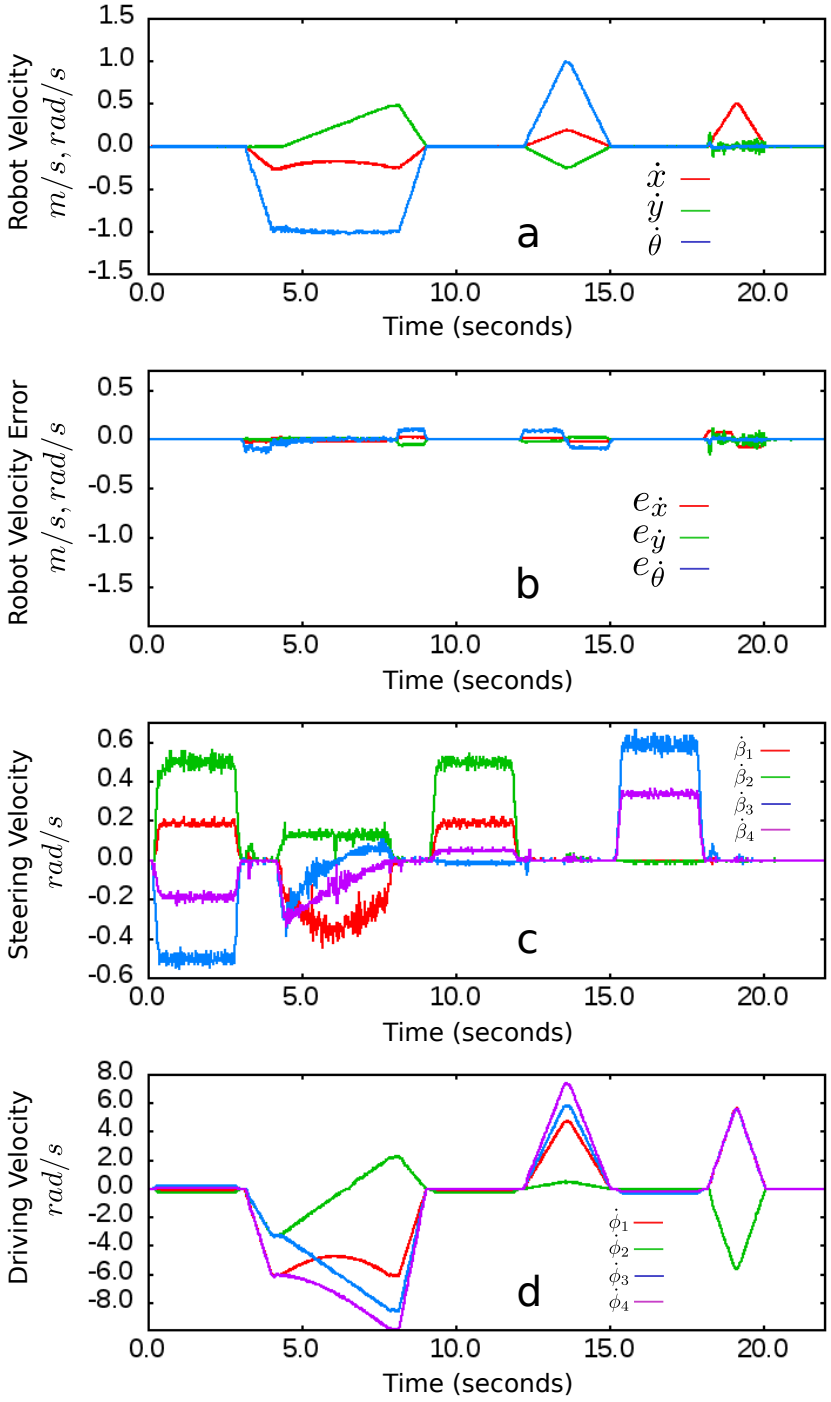


Figure 3.8: Measured response of the experiments employing the kinematic model plus singularity treatment, while executing the benchmark test trajectory.

extended periods of robot operation, significant steer joint error will appear due to accumulation of steering velocity errors. Another major drawback is that it is not easy to extend the kinematic model formulation in the $3D$ Cartesian space to handle

discontinuity in the robot velocity commands. Where it will be specially difficult to take into account both the maximum steering velocity and acceleration. Additionally, since both the steering and driving velocities are computed using the Cartesian space velocity command, this will result in limiting the maximum potential of either the steering or the driving. Usually the drive joints are the ones carrying all the load, and as such will have lower maximum performance limit as compared to the steering joints. This will result in lower performance of the steering system below its true potential. This is highly unacceptable, since the ability to perform steering as fast as possible, makes such mobile robot structure as close as possible towards being omnidirectional, which is the major merit of such structure.

3.7 Conclusion

A complete kinematic model for steerable wheeled mobile robots has been presented in terms of forward (odometry) and inverse (actuation) kinematics. The inverse kinematic model presented is free of representational singularities. Additionally, a numeric treatment is provided to solve the kinematic singularity, providing a completely singular-free model. A benchmark velocity trajectory is proposed, to evaluate the performance of the developed model in all singular conditions that can be encountered by such systems. Using the proposed benchmark test, the developed model has been tested on the Neobotix-MPO700 industrial mobile robot, to show superior results, as compared to the embedded controller.

3.8 Link with Next Chapter

This chapter assumes continuous velocity trajectory commands are sent to the robot, however, such assumption is not valid in the application in hand. We require a control framework that is robust against discontinuous commands provisioned by sharing workspace with humans, this motivates the next chapter.

Chapter 4

Motion Discontinuity Robust ICR Controller

Contents

4.1 Introduction 34
4.1.1 Motivation 35
4.1.2 Proposed control framework 36
4.2 ICR Velocity Controller 36
4.3 Handling Kinematic Singularity 39
4.4 Optimal next sample time ICR Point 40
4.5 Fixing Numeric Issues 43
4.6 The Driving Rate 43
4.7 Experiments 44
4.7.1 Benchmark trajectory 44
4.7.2 Results and Discussion 46
4.8 Conclusion 48
4.9 Link with Next Chapter 48

4.1 Introduction

In this chapter, we present a 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.

4.1.1 Motivation

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 with the robots. This imposes the need for human safe trajectory generators, that can lead to frequent halts in motion, re-planning, and sudden, discontinuous changes in the position of the robot's instantaneous center of rotation (ICR). All this requires steer joint reconfiguration to the newly computed trajectory. This issue is almost ignored in the literature, and motivates the work presented in this chapter.

Furthermore, to the best of our 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 [GFASH09], and "*open-loop starting procedure*" in [TNM96]. Although an ICR-based controller is the most suited to handle such cases, the work in [OGM13] and [OGM14] is limited by the assumption of continuous and differentiable desired trajectory, whereas in [SPS12] the singularities imposed by the ICR-based model are handled by reducing the robot maneuverability. As such, the state of the art will fail to provide proper solutions.
In this chapter, we propose an ICR based kinematic control framework that is:

1. robust against trajectory discontinuity,

2. capable of handling kinematic singularity in the ICR space (as opposed to the Cartesian velocity space method developed in Chapter 3),

3. compliant with the maximum steer joint limits in terms of velocity and acceleration (or seamlessly, jerk).

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 benefit of using separate

kinematic controllers for the drive and steering rates, is that it is not necessary to map the $2D$ ICR-coordinate space to the $3D$ Cartesian space, so associated singularities and inconveniences (non injectiveness) are avoided. The $3D$ Cartesian space kinematic model developed in Section 3.2.1 is free from representational singularity, while the kinematic singularities are being handled in the $2D$ ICR space controller.

4.1.2 Proposed control framework

The proposed control framework is depicted in Fig. 4.1. The desired $3D$ Cartesian space robot motion ($\dot{\boldsymbol{\xi}}^*$, and $\ddot{\boldsymbol{\xi}}^*$), is generated by a high level perception controller (out of the scope of this thesis). This is then mapped to the $2D$ ICR space, and the output desired ICR motion ($\boldsymbol{ICR}^*$, and $\boldsymbol{I\dot{C}R}^*$) is fed to the ICR velocity controller, along with the current ICR coordinates $\boldsymbol{ICR}_{curr}$. The output reference signal of the contoller $\boldsymbol{ICR}_{ref}$ is then fed to an optimization algorithm that determines the *"next sample time"* ICR coordinates $\boldsymbol{ICR}_{next}$ that will minimize the quadratic cost error: $\|\boldsymbol{ICR}_{ref} - \boldsymbol{ICR}_{next}\|_2^2$ while respecting the joint performance limits formulated as linear constraints. We use the $\boldsymbol{ICR}_{ref}$ rather than $\boldsymbol{ICR}^*$ in the cost function in order to obtain smooth behavior since it is error dependant, otherwise the steering joint will keep operating at its maximum limit, either velocity or acceleration, to go from the current steering angle to the desired one. When the ICR point reaches the desired value (while operating at maximum velocity), it cannot stop since it has to respect the maximum deceleration, so it keeps oscillating about the desired steering angle and eventually stops, this can result in instability, and as such it is more adequate to use an error based signal such as $\boldsymbol{ICR}_{ref}$. The corresponding steer joint reference signal $\boldsymbol{\beta}_{ref}$ is then evaluated (while fixing the numeric issues involved) and differentiated, to obtain the $\dot{\boldsymbol{\beta}}_{ref}$ that is sent to the motor drivers. At the same time, a decoupled robot velocity controller is implemented, the initial output of which $\dot{\boldsymbol{\xi}}_{ref(init)}$ is projected onto the null space of the *"next sample time"* robot configuration (represented by the kinematic constraint matrix $\boldsymbol{G}(\beta_{ref})$ defined in (3.5)), to obtain the feasible control signal $\dot{\boldsymbol{\xi}}_{ref}$. The reference wheel rate $\dot{\boldsymbol{\phi}}_{ref}$ is then obtained using the IAKM in (3.8). The blocks in Fig. 4.1 are detailed in each of the following sections.

4.2 ICR Velocity Controller

In Fig. 4.1, the desired base frame motion is the output of the perception (or motion planning) block. As shown in Fig. 4.2, the corresponding desired ICR coordinates,

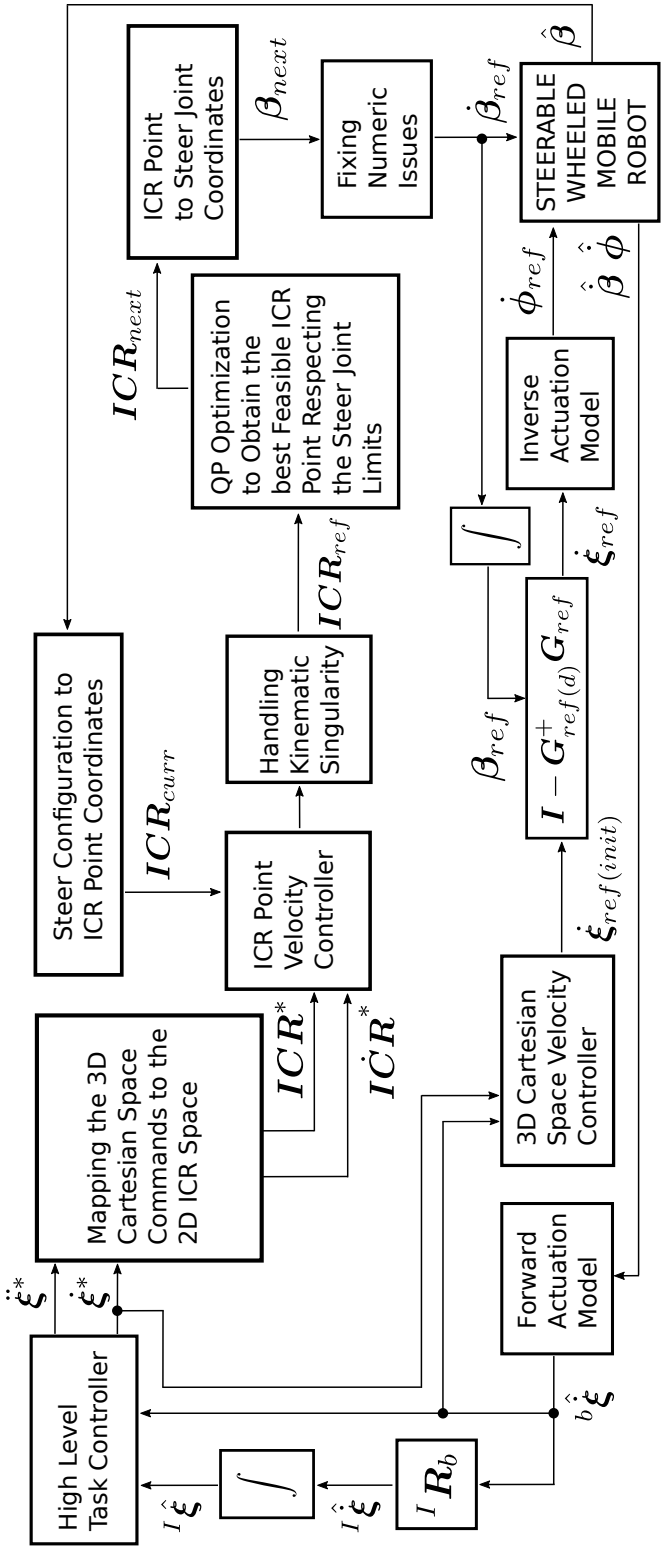


Figure 4.1: Proposed motion discontinuity-robust control framework.

expressed in base frame, $\boldsymbol{ICR}^* = \begin{bmatrix} X^* & Y^* \end{bmatrix}^T$, are usually evaluated using (3.12):

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

However, this formula is singular in pure translation motion (when $\dot{\theta}^* = 0$). Alternatively, here we propose to use:

$$\begin{aligned} X^* &= R_\infty \tanh\left(\left(\frac{-\dot{y}^*}{\dot{\theta}^* + sign(\dot{\theta}^*)\delta_1}\right)/R_\infty\right), \\ Y^* &= R_\infty \tanh\left(\left(\frac{\dot{x}^*}{\dot{\theta}^* + sign(\dot{\theta}^*)\delta_1}\right)/R_\infty\right), \end{aligned} \tag{4.1}$$

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

with R_∞ a large positive scalar that should be chosen carefully, to avoid numerical instability when computing the corresponding steer joint values using the arctan() function. On the other hand, δ_1 is an infinitesimally small positive scalar value.

Definition 4.2.1. *$R_\infty \in]0, \infty[$ is a parameter representing the radius of curvature of the path followed by the robot, corresponding to pure translation motion. It serves as an upper limit to the values allowed for the X and Y components of the $\boldsymbol{ICR}$.*

Remark 4.2.2. *Setting R_∞ to a value other than ∞ will result in non pure translation motion: there will always exist a rotation velocity component. Shall the desired command correspond to pure translation, the controller proposed in this work will provide the necessary corrective action.*

Equation (4.1) is both singularity free, and provides bounded $\boldsymbol{ICR}^*$ components thanks to the tanh() function. Similarly, the desired ICR velocity, $\dot{\boldsymbol{ICR}}^*$ can be computed using:

$$\begin{aligned} \dot{X}^* &= \dot{V}_{max} \tanh\left(\left(\frac{-\ddot{y}^*\dot{\theta}^* + \dot{y}^*\ddot{\theta}^*}{\left(\dot{\theta}^*\right)^2 + \delta_1}\right)/\dot{V}_{max}\right), \\ \dot{Y}^* &= \dot{V}_{max} \tanh\left(\left(\frac{\ddot{x}^*\dot{\theta}^* - \dot{x}^*\ddot{\theta}^*}{\left(\dot{\theta}^*\right)^2 + \delta_1}\right)/\dot{V}_{max}\right), \end{aligned} \tag{4.2}$$

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

On the other hand, the current ICR coordinates $\boldsymbol{ICR}_{curr} = \begin{bmatrix} X_{curr} & Y_{curr} \end{bmatrix}^T$ are computed using the current *"measured"* steer joint configuration, $\hat{\boldsymbol{\beta}}$. We assume that this configuration corresponds to a unique valid ICR point, i.e., that the motor

driver properly tracks the reference steer rate $\dot{\boldsymbol{\beta}}_{ref}$, and as such the measured steer joint values correspond to a *"unique"* $\boldsymbol{ICR}_{curr}$. For numeric robustness, we use the pair of steer joints with the biggest angular difference (see Fig. 4.2). Their indexes $i, j \in \{1, \ldots, N\}$ are given by:

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

then:

$$\begin{aligned} Y_{curr} &= \begin{cases} R_\infty \cos(\hat{\beta}_i), & \forall \hat{\beta}_i < \beta_{th} \\ \frac{h_{xi} - h_{xj} + h_{yi} \tan(\hat{\beta}_i) - h_{yj} \tan(\hat{\beta}_j)}{\tan(\hat{\beta}_i) - \tan(\hat{\beta}_j)}, & \text{otherwise} \end{cases}, \\ X_{curr} &= \begin{cases} R_\infty \sin(\hat{\beta}_i), & \forall \hat{\beta}_i < \beta_{th} \\ h_{xi} - (Y_{curr} - h_{yi}) \tan(\hat{\beta}_i), & \text{otherwise} \end{cases}, \end{aligned} \tag{4.3}$$

where β_{th} is a small threshold value. The result of (4.3) is saturated to $\pm R_\infty$ to obtain X_{curr} and Y_{curr}. Finally, the *ICR-Point Velocity Controller* is formulated as:

$$\begin{aligned} \dot{\boldsymbol{ICR}}_{ref} &= \dot{\boldsymbol{ICR}}^* + \lambda \boldsymbol{ICR}_{err}, \\ \boldsymbol{ICR}_{err} &= \boldsymbol{ICR}^* - \boldsymbol{ICR}_{curr}, \end{aligned} \tag{4.4}$$

where λ is a positive scalar proportional gain, and $\boldsymbol{ICR}_{err}$ is the error between the desired and current ICR coordinates. Such controller is known to guarantee asymptotic convergence.

4.3 Handling Kinematic Singularity

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:

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

will give an undefined value. In such case, the steering rate grows unbounded. This is handled in the literature either by constraining the robot velocity space [TNM96, DWASH11, CPHV09, SPS12], so that the ICR never passes by any steering axis, or by saturating the steer rate at singularity [OGM13]. In Section 3.3, 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 [SCPF16]. However, that method cannot be applied here, as we control steer joints in the $2D$ ICR space, rather than in Cartesian space.

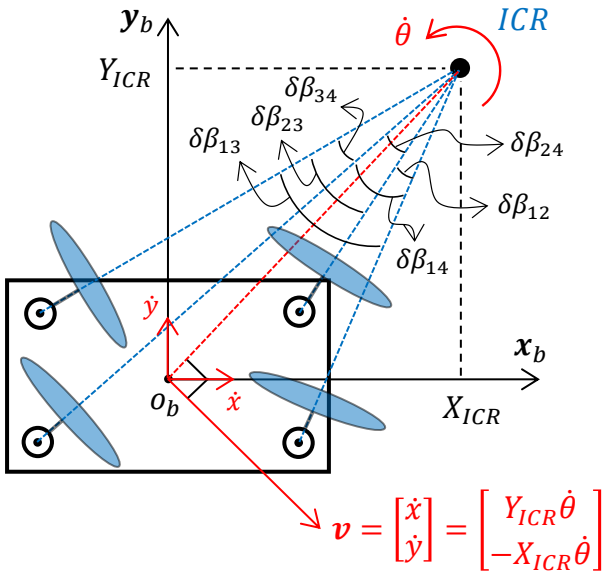


Figure 4.2: Relationship between the 3D Cartesian velocity space and the 2D ICR coordinate space. The steer joint pair with the biggest angular difference $\delta\beta_{ij} = |\beta_i - \beta_j|$ is used to compute $\boldsymbol{ICR}_{curr}$.

To handle this kinematic singularity in the $2D$ ICR space, we construct a singularity zone (circle with fixed radius R_{zone}) around each steering axis (the 2^{nd} steer joint zone, in Fig. 4.3). If the ICR enters the circle, $\boldsymbol{ICR}_{ref}$ 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 $\boldsymbol{ICR}_{ref}$ is modified as:

$$\begin{aligned} X_{ref} &= X_{curr} + 2R_{zone}\cos(\beta_s + \pi/2), \\ Y_{ref} &= Y_{curr} + 2R_{zone}\sin(\beta_s + \pi/2), \end{aligned} \tag{4.5}$$

with R_{zone}, the zone radius, and β_s, the singular steer angle.

4.4 Optimal next sample time ICR Point

The reference control signal $\boldsymbol{ICR}_{ref}$ obtained from (4.4) is error driven. When trajectory discontinuity occurs or the initial motion commands (with high error) are being sent, excessive joint velocity/acceleration is needed. This drives the ICR reference point far from the current one. In this section, a quadratic programming optimization problem is formulated to determine the best feasible *"next sample time"* ICR coordinates $\boldsymbol{ICR}_{next}$ 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,

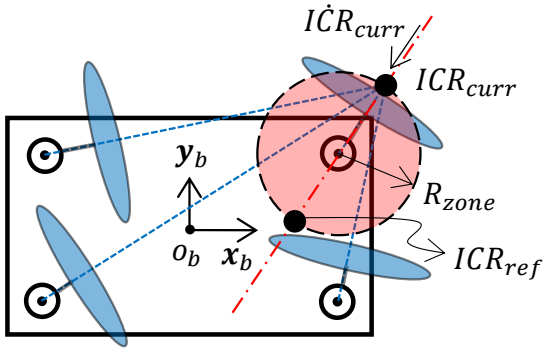


Figure 4.3: ICR point approaching the singularity zone (in red) of the 2^{nd} steer joint. ICR motion direction is indicated by the top arrow.

velocity and acceleration) limits. Then, $\boldsymbol{ICR}_{next}$ will replace $\boldsymbol{ICR}_{ref}$ in computing the steer joint commands.

Fig. 4.4 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 position of $\boldsymbol{ICR}_{curr}$,

2. the maximum steer joint velocity and acceleration.

Indeed, the feasible set of $\boldsymbol{ICR}_{next}$ is within the intersection of the extreme feasible changes of all steer joints. The feasible region for $\boldsymbol{ICR}_{next}$ shown in Fig. 4.4.a corresponds to a stationary $\boldsymbol{ICR}_{curr}$, which can move arbitrarily in any direction within the indicated feasible set. Instead, Fig. 4.4.b depicts the case of moving $\boldsymbol{ICR}_{curr}$ (i.e., $\dot{\boldsymbol{ICR}}_{curr} \neq 0$). In such case, the maximum steering deceleration constraints (shown in Fig. 4.4.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 4.4.c shows a particular case of Fig. 4.4.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 $\boldsymbol{ICR}_{ref}$ is required, while $\boldsymbol{ICR}_{curr}$ is following (and converging towards) an old one $\boldsymbol{ICR}_{ref(old)}$. Thanks to this approach, discontinuity in velocity commands 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 $\boldsymbol{ICR}_{ref}$ and the best feasible ICR at the next sample time $\boldsymbol{ICR}_{next} = \begin{bmatrix} X_{next} & Y_{next} \end{bmatrix}^T$.

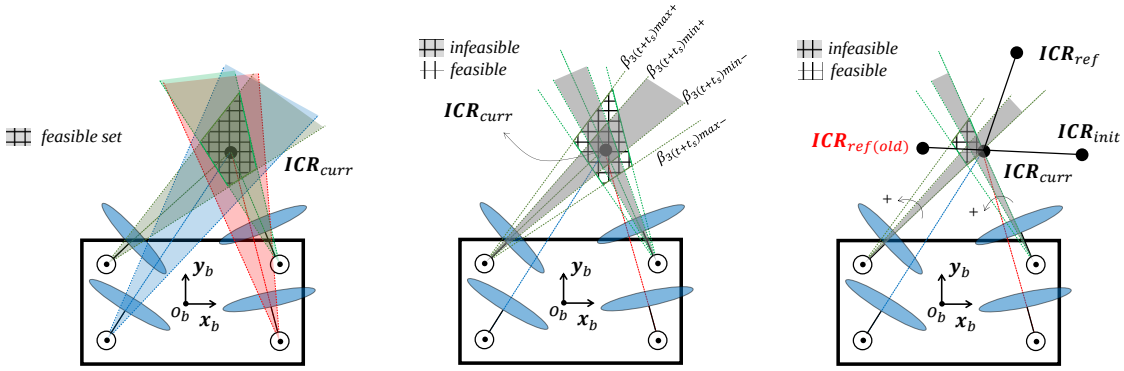


Figure 4.4: Feasible set of solutions for the ICR coordinates, respecting the maximum steer joint performance with: (a) stationary ICR, (b) moving ICR (general case), and (c) moving ICR (particular case) with $\dot{\beta}_2 > 0$ and $\dot{\beta}_3 > 0$, t_s is the sample time.

The latter is also the decision variable:

$$\begin{aligned} \underset{\boldsymbol{ICR}_{next}}{\text{minimize}} \quad & \|\boldsymbol{ICR}_{ref} - \boldsymbol{ICR}_{next}\|_2^2 \\ \text{subject to} \quad & (-1)^{q_{i(min)}} \boldsymbol{A}_{i(min)} (\boldsymbol{ICR}_{next} - \boldsymbol{h}_i) \geq 0, \\ & (-1)^{q_{i(max)}+1} \boldsymbol{A}_{i(max)} (\boldsymbol{ICR}_{next} - \boldsymbol{h}_i) \leq 0, \end{aligned}$$

where $\| * \|_2^2$ is the squared Euclidean norm, $\boldsymbol{A}_{i(max/min)} = \begin{bmatrix} \cot(\beta_{i(max/min)}) & 1 \end{bmatrix}$, $\boldsymbol{h}_i = \begin{bmatrix} h_{xi} & h_{yi} \end{bmatrix}^T$, and $q_{i(max/min)}$ is a parameter indicating the $\beta_{i(max/min)}$ quadrature:

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

The value of $\boldsymbol{ICR}_{ref}$ is obtained either from (4.4) by numeric integration, or from (4.5) in case of kinematic singularity. Using $\boldsymbol{ICR}_{ref}$ instead of $\boldsymbol{ICR}^*$ provides smooth damped behavior, since its computation depends on $\boldsymbol{ICR}_{err}$ (refer again to (4.4)). The constraints are the straight lines of slopes $\beta_{i(max)}$ and $\beta_{i(min)}$:

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

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

$$\beta_{i(max/min)} = \beta_i + \dot{\beta}_{(max/min)(ref)} * t_s. \tag{4.6}$$

Finally, using the optimal $\boldsymbol{ICR}_{next}$, the next best steer angles are computed:

$$\beta_{i(next)} = \arctan 2\Big(Y_{next} - h_{yi}, X_{next} - h_{xi}\Big) - \frac{\pi}{2}. \tag{4.7}$$

4.5 Fixing Numeric Issues

Two numeric issues can be observed, when employing the previously described controller. The first originates from the $\arctan 2()$ function in (4.7) whenever $\beta_{i(next)}$ moves between 4^{th} and 1^{st}, or between 2^{nd} and 3^{rd} quadrants. This results in 2π 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 π jump in $\beta_{i(next)}$. This problem has been avoided previously in Section 3.3, by employing the steer joint rate. However, a different method is necessary here, since we compute the joint angle rather than its rate in (4.7). To handle these two issues, we use $\Delta\beta_i = \beta_{i(next)} - \beta_i$ to detect and fix the jumps:

$$\dot{\beta}_{i(ref)} = \begin{cases} (\Delta\beta_i + l\pi)/t_s, & \Delta\beta_i < -l\pi + 2\dot{\beta}_{max}t_s \\ (\Delta\beta_i - l\pi)/t_s, & \Delta\beta_i > l\pi - 2\dot{\beta}_{max}t_s \\ \Delta\beta_i/t_s, & \text{otherwise,} \end{cases} \tag{4.8}$$

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

4.6 The Driving Rate

To compute the wheel drive rate $\dot{\phi}_{i(ref)}$ from (3.8), the reference velocity vector $\dot{\boldsymbol{\xi}}_{ref}$ equivalent to $\boldsymbol{ICR}_{next}$ is needed. However, this is not feasible, since one information is missing when mapping from the $2D$ space of $\boldsymbol{ICR}_{next}$ to the $3D$ space of $\dot{\boldsymbol{\xi}}_{ref}$. To avoid such inconvenience, we propose a separate *3D Cartesian Space Velocity Controller* (see Fig. 4.1), to compute the driving rate. First, a simple proportional controller is used to evaluate the reference signal $\ddot{\boldsymbol{\xi}}_{ref(init)}$ (the subscript *init* indicates that this is an initial value, not yet the final one to be applied to the robot):

$$\ddot{\boldsymbol{\xi}}_{ref(init)} = K_p(\dot{\boldsymbol{\xi}}^* - \dot{\boldsymbol{\xi}}), \tag{4.9}$$

with K_p a positive scalar gain, and $\dot{\boldsymbol{\xi}}^*$ the desired robot velocity output by the task controller. The corresponding $\dot{\boldsymbol{\xi}}_{ref(init)}$ (obtained by numeric integration of (4.9)) may be incompatible with the "*next sample time*" robot configuration obtained by substituting β_{ref} (obtained by integrating (4.8)) in the kinematic constraint matrix (3.5), where $\boldsymbol{G}_{ref} = \boldsymbol{G}(\boldsymbol{\beta}_{ref})$. Hence, $\dot{\boldsymbol{\xi}}_{ref}$ is obtained by projecting $\dot{\boldsymbol{\xi}}_{ref(init)}$ onto the null-space of $\boldsymbol{G}_{ref}$:

$$\dot{\boldsymbol{\xi}}_{ref} = (\boldsymbol{I} - \boldsymbol{G}^+_{ref(d)}\boldsymbol{G}_{ref})\dot{\boldsymbol{\xi}}_{ref(init)}. \tag{4.10}$$

Substituting by $\dot{\boldsymbol{\xi}}_{ref}$ in (3.8), we finally obtain $\dot{\phi}_{i(ref)}$.

4.7 Experiments

In this section, the benchmark test trajectory presented in Section 3.4 is used again, to evaluate our discontinuity robust control framework on the industrial mobile robot Neobotix MPO700 (shown in Fig. 3.5). All parameters used in the experiments are provided in Table 5.1. A video of the experiment can be found on the IDH YouTube channel [1]

4.7.1 Benchmark trajectory

The benchmark trajectory proposed in Section 3.4 and depicted in Fig. 3.3, 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).

4.7.1.1 Parabolic ICR position profile

A parabolic ICR-point motion with vertex at one of the steering axes is used, and implemented smoothly using a constant velocity profile:

$$\begin{cases} \dot{x}^* = 0.5 * (2\dot{y} + h_{x2})^2 + h_{y2} \\ \dot{y}^* = T_{c4b}(\dot{y}_i = 0, \dot{y}_f = -h_{x2}, t_i = 2, t_f = 6, t) \\ \dot{\theta}^* = 0.5. \end{cases} \tag{4.11}$$

[1] https://youtu.be/P9aIIwz6qLE

Table 4.1: Robot and Controller parameters used in the singularity-robust control framework experiments.

δ	0.001	$\dot{V}_{max}$	$10m/s$	β_{th}	$0.005rad.$
λ	3.7	R_{zone}	$0.015m$	$\dot{\beta}_{max}$	$2rad./s$
$\ddot{\beta}_{max}$	$25rad./s^2$	t_s	$25ms$	K_p	2

Here, T_{c4b} is a 4^{th} 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 ... 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 (4.11), the following velocity command is applied, to guarantee that the parabolic ICR profile starts at the correct initial condition:

$$\dot{\boldsymbol{\xi}}^* = 0.5 * \begin{bmatrix} h_{x2}^2 + h_{y2} & 0 & 1 \end{bmatrix}^T, \quad 0 \leq t < 2.$$

4.7.1.2 ICR point at kinematic singularity

Cartesian robot velocity used:

$$\dot{\boldsymbol{\xi}}^* = 0.5 * \begin{bmatrix} h_{y1} & -h_{x1} & 1 \end{bmatrix}^T, \quad 6 \leq t < 8.$$

4.7.1.3 Pure linear motion

Done by sending *"zero angular velocity"* and arbitrary linear velocity v_{ar}:

$$\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} v_{ar1} & v_{ar2} & 0 \end{bmatrix}^T, \quad 8 \leq t < 10.$$

4.7.1.4 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)*, realized by the following command sequence:

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

4.7.1.5 Zero Velocity

Cartesian robot velocity command used:

$$\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T, \quad 28 \leq t < 30.$$

4.7.2 Results and Discussion

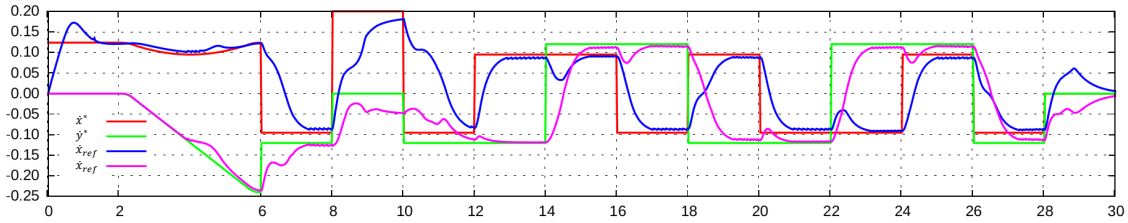
Applying the benchmark test trajectory shown in Fig. 4.5a and Fig. 4.5b to the proposed controller, we obtain the steer joint velocity and acceleration commands in Fig. 4.5c and Fig. 4.5d, respectively. As shown, these commands respect the maximum joint performance limits. The corresponding Cartesian space velocity is also shown in Fig. 4.5a and Fig. 4.5b, 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. 4.6a, showing some overshoot due to the embedded controller. However, 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. 4.5e and Fig. 4.6b, respectively. In Fig. 4.5c, 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:

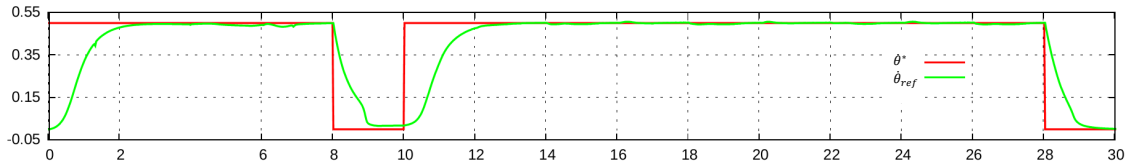
1. The singular joint does not rotate at the maximum limit, which is favorable as noted earlier,

2. low amplitude vibration (chattering) exists in the steering joints other than the singular one (see also test 4).

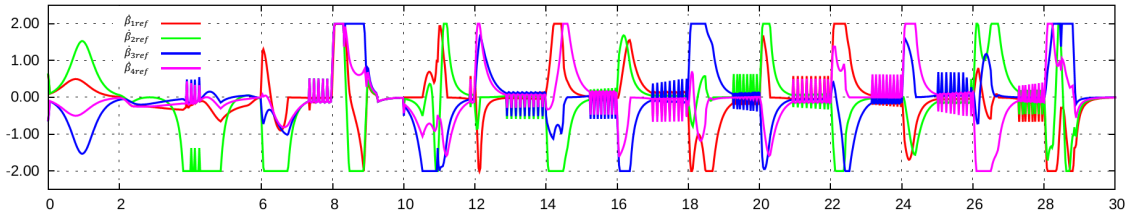
The second result is due to the singularity treatment algorithm, which keeps driving $\boldsymbol{ICR}_{ref}$ from one side of the singular zone circle to the other. Where the benchmark test explicitly requires the ICR point to be "at" kinematic singularity, which is not allowed in the way kinematic singularity is handled in section 4.3. Although such chattering behavior looks ugly and can result in vibration of the stter joint mechanical system, it can be a promising solution by decreasing the controller sample time t_s, below the response time of the mechanical system (the sample time can be as low as 1 milliseconds on a CORE i7 processor). However, this was not achievable without rewriting the embedded actuation controller, a task that we plan to do in future work.



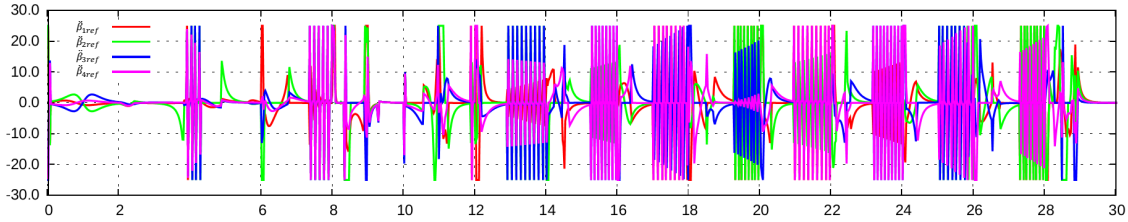
(a) Linear velocity: desired and output by the proposed controller.



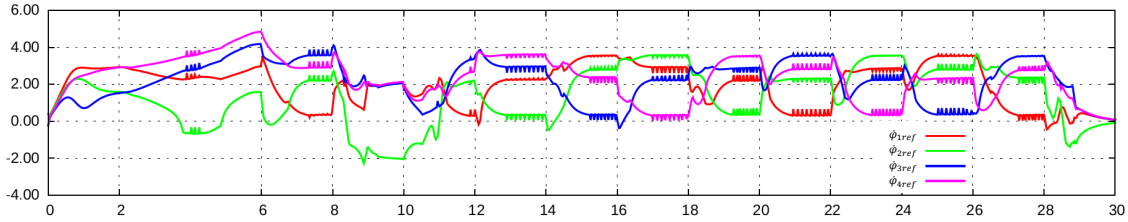
(b) Angular velocity: desired and output by the proposed controller.



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



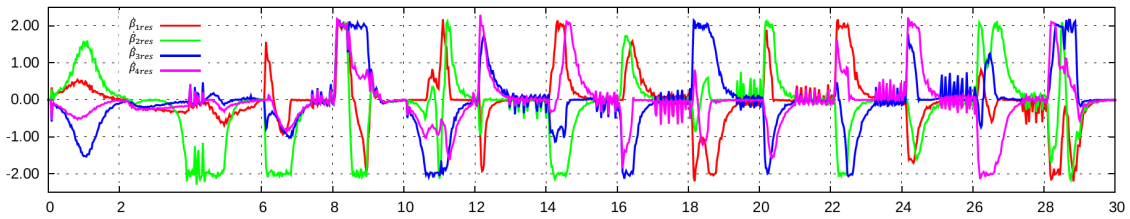
(d) Output steering acceleration command of the proposed controller.



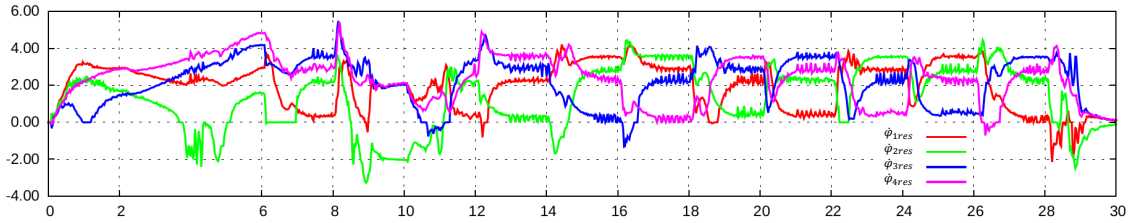
(e) Output drive speed command of the proposed controller.

Figure 4.5: Results of the proposed controller along the "discontinuous" benchmark trajectory.

The vibration frequency can also be attenuated by taking into account the maximum steering jerk in the QP formulation. Finally, it is worth noting that our kinematic controller assumes a high performance joint space dynamic controller implemented on the mobile robot in hand (very small joint space position/velocity error is assumed).



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



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

Figure 4.6: Joint space response results of the proposed controller, applied to the MPO700 industrial mobile robot.

4.8 Conclusion

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 successfully performed, using the proposed controller. Maximum steer joint performance limits are taken into account and have shown to be respected throughout the experiments. In future work, the drive joint maximum performance limits will be added to the proposed framework.

4.9 Link with Next Chapter

Chapter 3 introduced a kinematic model that is robust to singularities, that can be used in controlling the SWMR in $3D$ Cartesian velocity space assuming continuous velocity commands. In contrast to this approach, this Chapter employs an ICR based controller in order to deal with discontinuity in the robot velocity commands. However, for heading control applications where a constant robot orientation is desired (quite common in visual servoing tasks), the reference ICR (output of the ICR controller) will keep changing the sign of the angular velocity, which in turn requires moving the ICR point from one extreme of the ICR space to the other (refer to (3.12)). The required ICR movement is long and passes by the robot geometric center, at which ICR position, the feasible robot velocity is rotation in place, and as such

robot motion is neither efficient nor satisfactory. Providing a solution to such issue motivates the next Chapter.

Chapter 5

Complementary Route Based ICR Controller

Contents

5.1 Introduction 50
5.2 Complementary ICR Route 53
5.2.1 Complementary Versus Direct Routes 54
5.2.2 One Sample Period ICR Border Switching 57
5.2.3 Route Planning 58
5.2.4 Automatic Joint Space Border Switching 60
5.3 Direct OR Complementary 62
5.4 Experiments 65
5.4.1 Command Fulfillment Index 66
5.4.2 Discontinuous Velocity Commands Simulation 66
5.4.3 Practical Application Scenario and Results 72
5.5 Conclusion 78

5.1 Introduction

Kinematic modeling and control of SWMR is usually done either in the $2D$ ICR space [CPHV08, CHV12, SPS12, CBC+10] or in the $3D$ Cartesian space [GFASH09, BC96, SCPF16, TNM96]. In [GH14, DWASH11, SCFP17], both are combined, since ICR space is best suited for steering coordination (to avoid actuator fighting and wheel slippage) as it ensures the existence of a unique ICR point, while Cartesian space is utilized for robot speed control. We also use ICR-based steer coordination,

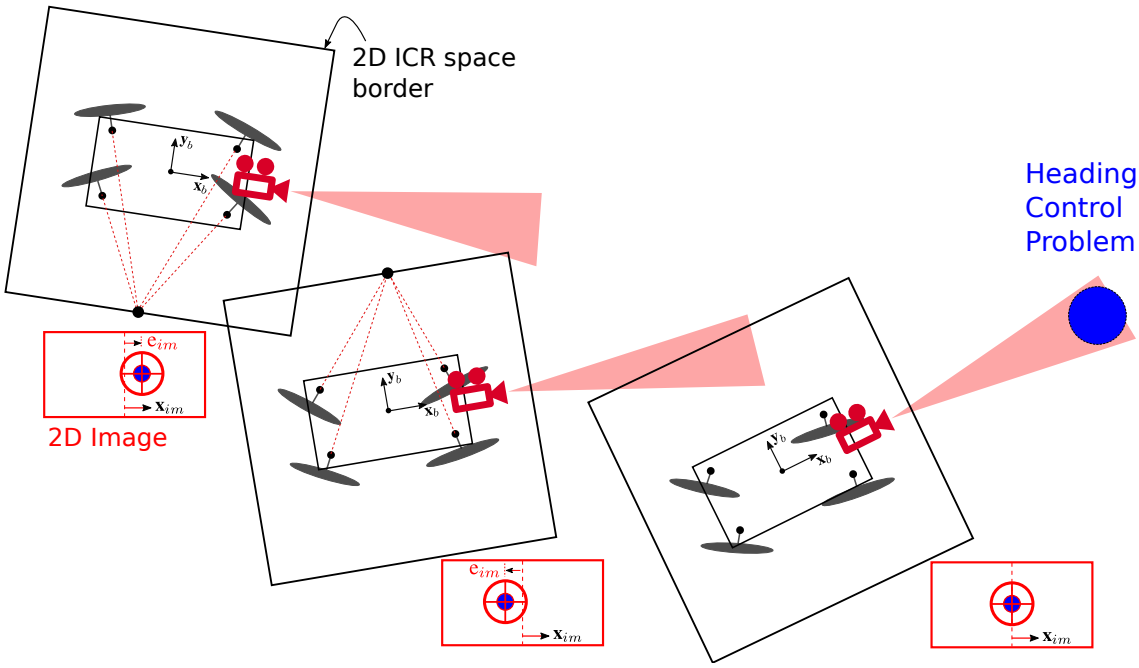


Figure 5.1: An example of heading control application, where changing the error sign will result in vast ICR point motion requirement.

and the focus of this chapter is precisely to enhance ICR based controllers, through the design of a complementary route strategy.

In applications requiring that some heading angle is to be maintained, some object or feature stays in the field of view (e.g., for vision-based tasks), or the translation verse changes, the ICR point is required to move long distances from one extreme of the workspace to the other, usually passing by the robot geometric center, where the feasible robot velocity is limited. An illustration of such application is shown in Fig. 5.1, where the blue object is to be maintained in the vision cone (light red colored), while the robot is moving from an initial pose to a final relative pose with the object. A visual servoing controller based on the error between the current and desired object horizontal location in the image frame (e_{im}) is a common approach. In such case the error sign fluctuation, will result in frequent sign change in the desired robot angular velocity $\dot{\theta}^*$, that would in turn result in sign change for at least one component of the desired ICR vector (refer to Fig. 5.2), which means large ICR motion requirement. In such scenarios, the state-of-art ICR based controllers will lead to unsatisfactory behavior/results.

To solve this problem, here we propose a comparison between the direct and complementary ICR routes, the former (state-of-art approach) being the shortest straight line connecting the current and desired ICR points, while the latter (proposed

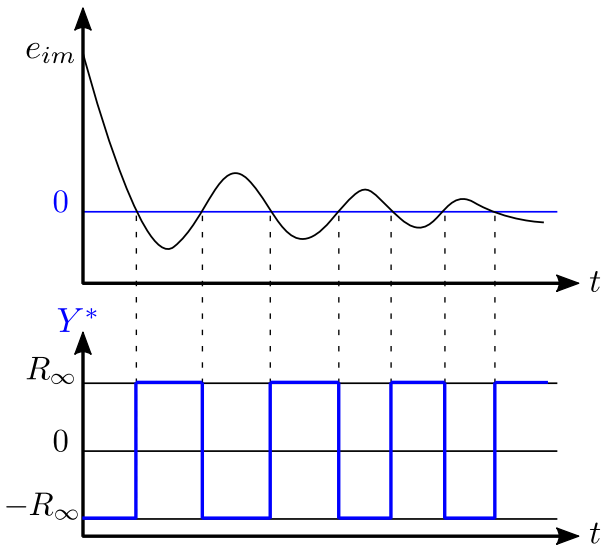


Figure 5.2: An example of heading control application, where changing the error sign will result in vast ICR point motion requirement.

here) connecting the ICR extreme borders via a border ICR point that is chosen to minimize the total ICR distance moved across borders. The 4 borders here define the maximum values that the ICR point is allowed to take on the x and y axes of the geometrically centered robot frame. Instead of moving directly to the desired ICR, the complementary route will move first to the optimum border point lying on the nearest border line, switch borders, *for example: from the $+y$ to the $-y$ border line* in one sample period, then move to the desired ICR point. A graphical representation of such process is shown in Fig. 5.3.

In [CHV12], the direct ICR route is stereographic projected onto a unit sphere, where moving between complementary borders can be done at the pole of the unit sphere. However, no investigation on border switching nor simulations showing the joint-space performance are provided. To the best of our knowledge, the work presented in this chapter is the first connecting ICR borders for more time efficient ICR controllers.

One of the contributions of the method adopted here is to provide a decoupled solution from the high level command/perception controller, where the trajectory planning doesn't need prior knowledge of the robot structure: only the current robot velocity and pose will suffice to close the feedback loop. Error between the command provided and the actual capability of the robot is handled locally. In order to assess the enhanced performance quantitatively, here, we introduce the command fulfillment index (CFI) that is based on the robot velocity error vector, and use it to compare

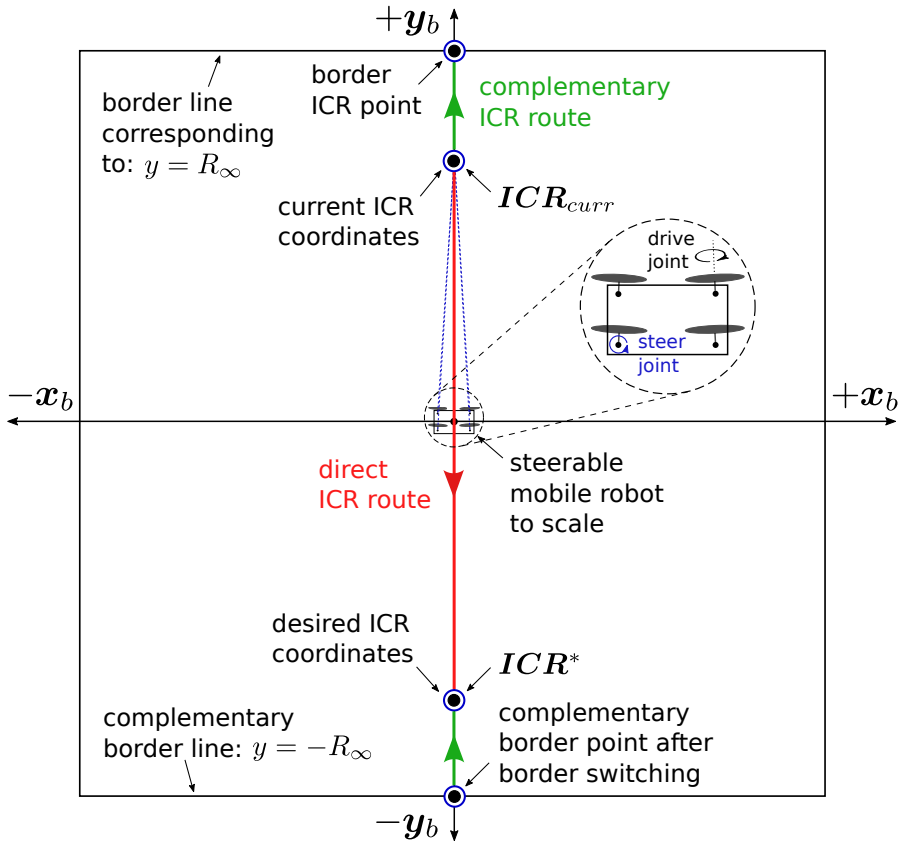


Figure 5.3: Steerable mobile robot $2D$ ICR space is bounded in this work by 4 border lines corresponding to $x = \pm R_\infty$ and $y = \pm R_\infty$ defined in the robot frame $\boldsymbol{\mathcal{F}}_b$ with origin at the robot center. An ICR point moving from the current to the desired value can follow a direct route (in red), or a complementary one (in green). Here the parameter $R_\infty = 10m$.

the direct and complementary route controllers.

In this chapter, we extend the kinematic control framework of described in chapter 4 so that it:

1. can handle sign variations in rotational speed commands, that frequently occurr in orientation (heading) control applications,

2. has better command fulfillment index (responsiveness),

3. results in less steering motion demand (of both velocity and acceleration).

5.2 Complementary ICR Route

In this section, the complementary ICR route algorithm is described/formulated. First we show the difference with respect to the direct route and that in some sit-

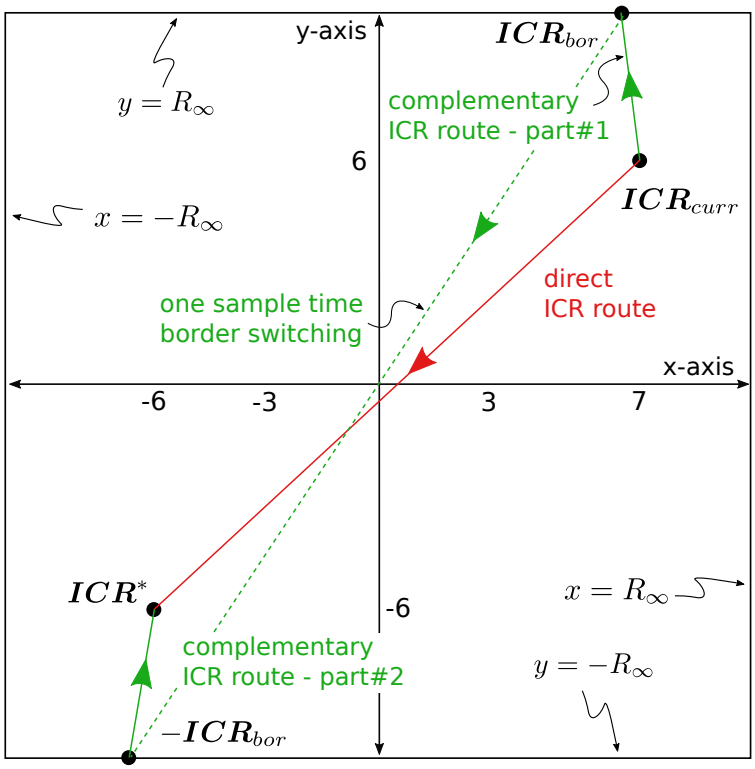


Figure 5.4: Illustrative schematic showing the complementary versus the direct ICR routes.

uations it can be shorter in length. Then we formulate a QP optimization problem to find the shortest complementary route to the desired ICR point from the current one. We then describe in detail how to implement the algorithm in conjunction with the discontinuity-robust ICR controller developed in Chapter 4.

5.2.1 Complementary Versus Direct Routes

The direct and complementary ICR routes are depicted in Fig. 5.4 in red and green colors respectively. The direct is the shortest direct straight line connection between the current point $\boldsymbol{ICR}_{curr}$, and the desired one $\boldsymbol{ICR}^*$. The complementary, on the other hand, is a concatenation of two successive straight lines interconnected *(detailed in next section)* via the complementary border lines, defined below.

Definition 5.2.1. *A border line defines an upper limit to the accessible workspace of the ICR point in any of the principle axes of the Cartesian space coordinates in the mobile base frame. As such there exist* 4 *border lines:* $x = \pm R_\infty$ *and* $y = \pm R_\infty$.

Definition 5.2.2. *A border point is any point belonging to a border line.*

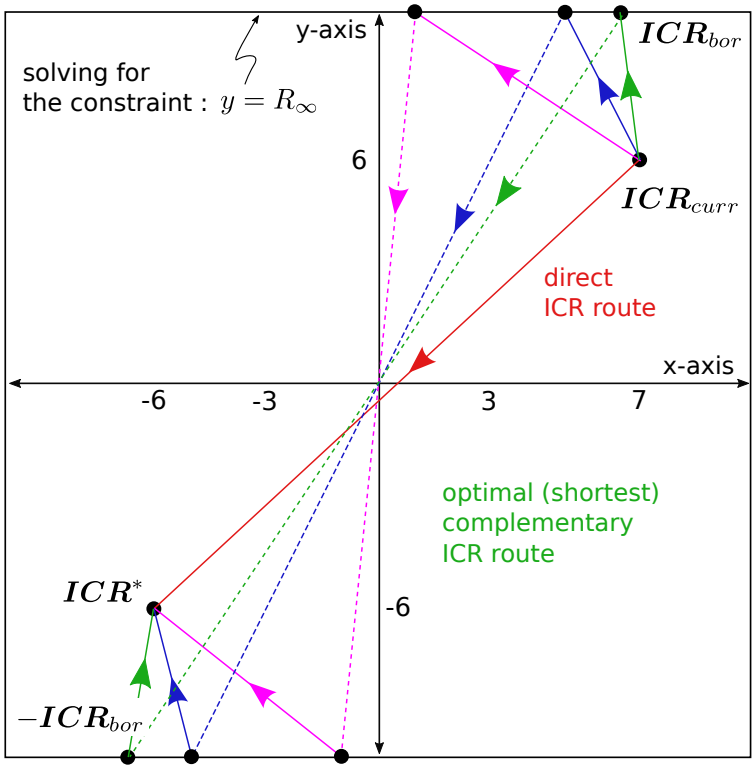


Figure 5.5: The locally optimal complementary ICR route (in green) among other possible routes (magenta and blue) satisfying one constraint $y = R_\infty$.

Definition 5.2.3. *The border is the perimeter of the square constructed from the 4 border lines.*

For the particular pair of $\boldsymbol{ICR}_{curr}$ and $\boldsymbol{ICR}^*$ shown in Fig. 5.4, the complementary route is the shortest and as such is expected to be time optimal. The ICR border point $\boldsymbol{ICR}_{bor} = \begin{bmatrix} X_{bor} & Y_{bor} \end{bmatrix}^T$ used to construct the complementary route belongs to the border line $y = R_\infty$, although any $\boldsymbol{ICR}_{bor}$ belonging to the other 3 border lines can be used as well. Furthermore, there exist infinite points satisfying $y = R_\infty$, each resulting in a complementary route, as shown in Fig. 5.5, where the blue and magenta lines represent alternative solutions. This reflects the need to choose the optimal (shortest) complementary route, which in this example (and for this particular border line), is represented by the green line. In order to find the $\boldsymbol{ICR}_{bor}$ corresponding to the shortest complementary route (refer to Fig. 5.6), we use quadratic programming. Since the feasible domain is the perimeter of the square border (constructed from the 4 border lines) as depicted in Fig. 5.6, the optimization problem is non convex.

Property 5.2.4. *The non convex feasible domain (the border) of the* $\boldsymbol{ICR}_{bor}$ *is composed of 4 convex sub-domains (the 4 border lines).*

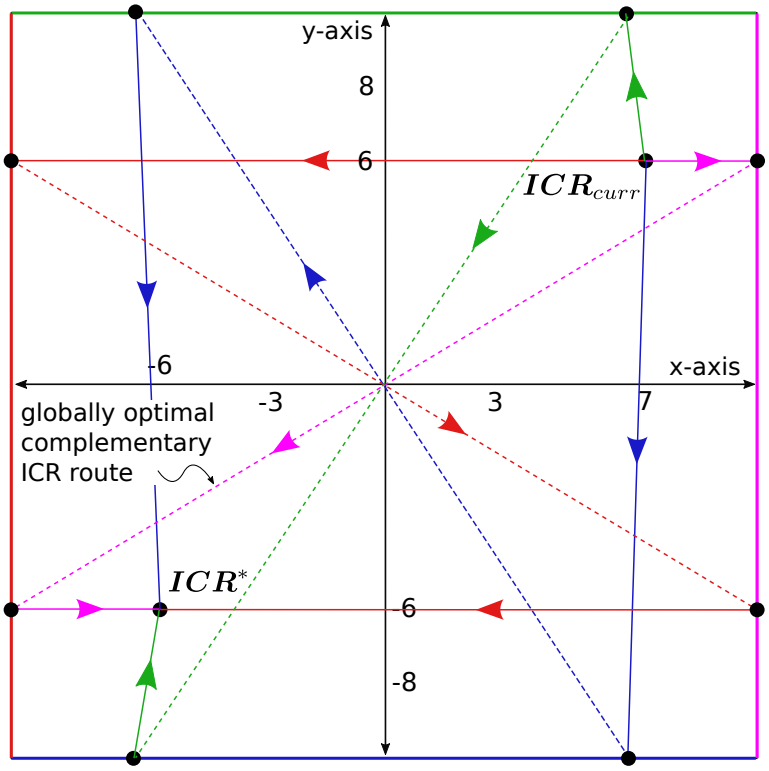


Figure 5.6: The global optimum complementary route (magenta) among the four local optima, each satisfying one straight line constraint.

To simplify the problem, we divide it into four separate QP formulations having the same cost function but different constraints, each representing one of the border lines, shown in Fig. 5.6 with different colors. Now each formulation is convex and can be solved easily. The general formulation is given by:

$$\begin{aligned} \underset{\boldsymbol{ICR}_{i(bor)}}{\text{minimize}} \quad & f(\boldsymbol{ICR}_{i(bor)}) = \\ & \|\boldsymbol{ICR}_{curr} - \boldsymbol{ICR}_{i(bor)}\|_2^2 \\ & + \|\boldsymbol{ICR}^* - \boldsymbol{ICR}'_{i(bor)}\|_2^2, \\ \underset{i=[1,2,3,4]}{\text{subject to}} \quad & \boldsymbol{c}_i \boldsymbol{ICR}_{i(bor)} = (-1)^i R_\infty, \end{aligned} \tag{5.1}$$

where $\| * \|_2^2$ is the squared Euclidean norm, $\boldsymbol{ICR}_{i(bor)}$ is the ICR border point corresponding to the shortest (local minimum) complementary route using the i^{th} border line, $\boldsymbol{ICR}'_{i(bor)} = -\boldsymbol{ICR}_{i(bor)}$, $\boldsymbol{c}_{1,2} = \begin{bmatrix} 1 & 0 \end{bmatrix}$, and $\boldsymbol{c}_{3,4} = \begin{bmatrix} 0 & 1 \end{bmatrix}$ indicate the constraint vectors. This QP optimization has been implemented using the QuadProg++ library [Gas]. Finally the index j corresponding to lower cost f_{min} is computed to obtain the global minimum:

$$\begin{aligned} \begin{bmatrix} f_{min} | j \end{bmatrix} &= \min_{i=[1,2,3,4]} (\{ \begin{bmatrix} f(\boldsymbol{ICR}_{i(bor)}) | i \end{bmatrix} \}) \\ \boldsymbol{ICR}_{bor} &= \boldsymbol{ICR}_{j(bor)}. \end{aligned} \tag{5.2}$$

Using (5.2), the $\boldsymbol{ICR}_{bor}$ corresponding to the shortest complementary route is found (colored magenta in Fig. 5.6) among the infinite points lying on the square shaped border.

5.2.2 One Sample Period ICR Border Switching

To maintain unique ICR existence (in accordance with the steer joint configuration) at all times, the ICR border switching has to be done in one sample period. To be able to do this, restrictions on the value of R_∞ have to be respected, depending on the maximum change in steering angle that can be done in one sample time ($\delta\beta_{max}$ in Fig. 5.7) which is in turn related to the robot hardware (actuator) constraints. On the other hand, R_∞ is limited to certain range of upper values since the optimization algorithm run by the motion discontinuity-robust controller will have limited freedom in the feasible region at ICR points far from the robot center moving at high ICR velocity $\dot{\boldsymbol{ICR}}_{ref}$ (refer to Fig. 5.10). In such case, the feasible region for the $\boldsymbol{ICR}_{next}$ will be confined to move towards the ICR space borders, a favorable behavior for the complementary route motion, but not for the normal (most usual) operating conditions. To avoid such inconvenience, we introduce the extended version $R_{\infty(e)}$ that is only used for complementary route execution.

Definition 5.2.5. *$R_{\infty(e)} = \{x \in \mathbb{R}^+ : R_\infty < x < \infty\}$, is an extended value of the upper limit allowed for the components of the $\boldsymbol{ICR}$, such that border switching can be performed in one sample period.*

Definition 5.2.6. *Border switching is a joint space operation where the steer joints are reconfigured from $\mathcal{M}(\boldsymbol{ICR}_{bor})$ to $\mathcal{M}(-\boldsymbol{ICR}_{bor})$, where $\mathcal{M} : \boldsymbol{ICR} \mapsto \boldsymbol{\beta}$ is a coordinate mapping from the ICR space to the steer joint space.*

From the geometry of the green triangle in Fig. 5.7, $R_{\infty(e)}$ can be evaluated in the worst case scenario as:

$$\begin{aligned} R_{\infty(e)} &\geq b * \tan\left(\frac{\pi - \delta\beta_{max}}{2}\right) + o_{band} \\ \delta\beta_{max} &= \min(\{\dot{\beta}_{max} * t_s, \ddot{\beta}_{max} * t_s^2\}), \end{aligned} \tag{5.3}$$

where $\dot{\beta}_{max}$, and $\ddot{\beta}_{max}$ denote the maximum steering velocity and acceleration characteristics of the robot respectively, o_{band} is a positive scalar (hand tuned) band offset. For further clarification, R_∞ is usually selected in the range of $10 \sim 20m$, whereas $R_{\infty(e)}$ as computed using (5.3) along with the robot-controller parameters provided in Table 5.1, evaluates to $\approx 31m$ at sampling time $t_s = 25ms$ (or $\approx 192m$ at $t_s = 10ms$).

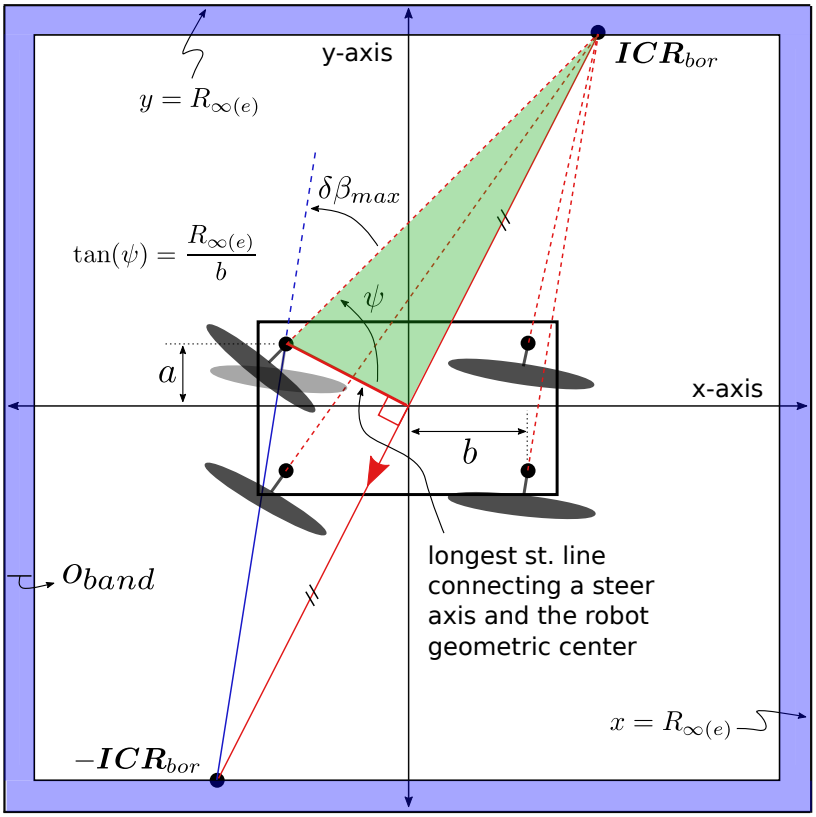


Figure 5.7: Evaluating the minimum value of $R_{\infty(e)}$ at the worst case scenario (biggest change in steer joint $\delta\beta$) to be able to perform border switching in one sample period. The mobile base schematic is magnified w.r.t. the ICR border lines for illustrative purposes.

5.2.3 Route Planning

To implement the complementary ICR route, its time efficiency with respect to the direct one is first evaluated using a quantitative metric (detailed in the next subsection). Once the route is verified as being more time efficient, the complementary enable flag F_{ce} is set to 1 allowing its execution. Such route requires the following set of 4 consecutive steps $S = \{S_1, \ldots, S_4\}$ with, S_1: moving to the optimum border point $\boldsymbol{ICR}_{bor}$ at R_∞ (Fig. 5.8a), S_2: extending the border lines to $R_{\infty(e)}$ (Fig. 5.8b), S_3: one sample period border switching (Fig. 5.8c), S_4: retracting the border lines back to R_∞ (Fig. 5.8d), at which step, the complementary route algorithm is disabled ($F_{ce} = 0$) and the direct route is then utilized finally to the desired $\boldsymbol{ICR}^*$. The elements of the Boolean set S are initialized to 1 at the beginning of the complementary route execution. These steps are implemented by manipulating the modified ICR

desired point $\boldsymbol{ICR}^*_{mod}$, and the ICR controller gain λ as follows:

$$\begin{aligned}
&\forall S_1 \wedge F_{ce} \wedge (|\boldsymbol{ICR}_{err(c)}| > c_{th}) : \\
&\quad \begin{cases} \boldsymbol{ICR}^*_{mod} = \boldsymbol{ICR}_{bor}, \\ \lambda = \lambda_n, \end{cases} \\
&Otherwise : S_1 = 0,
\end{aligned} \tag{5.4}$$

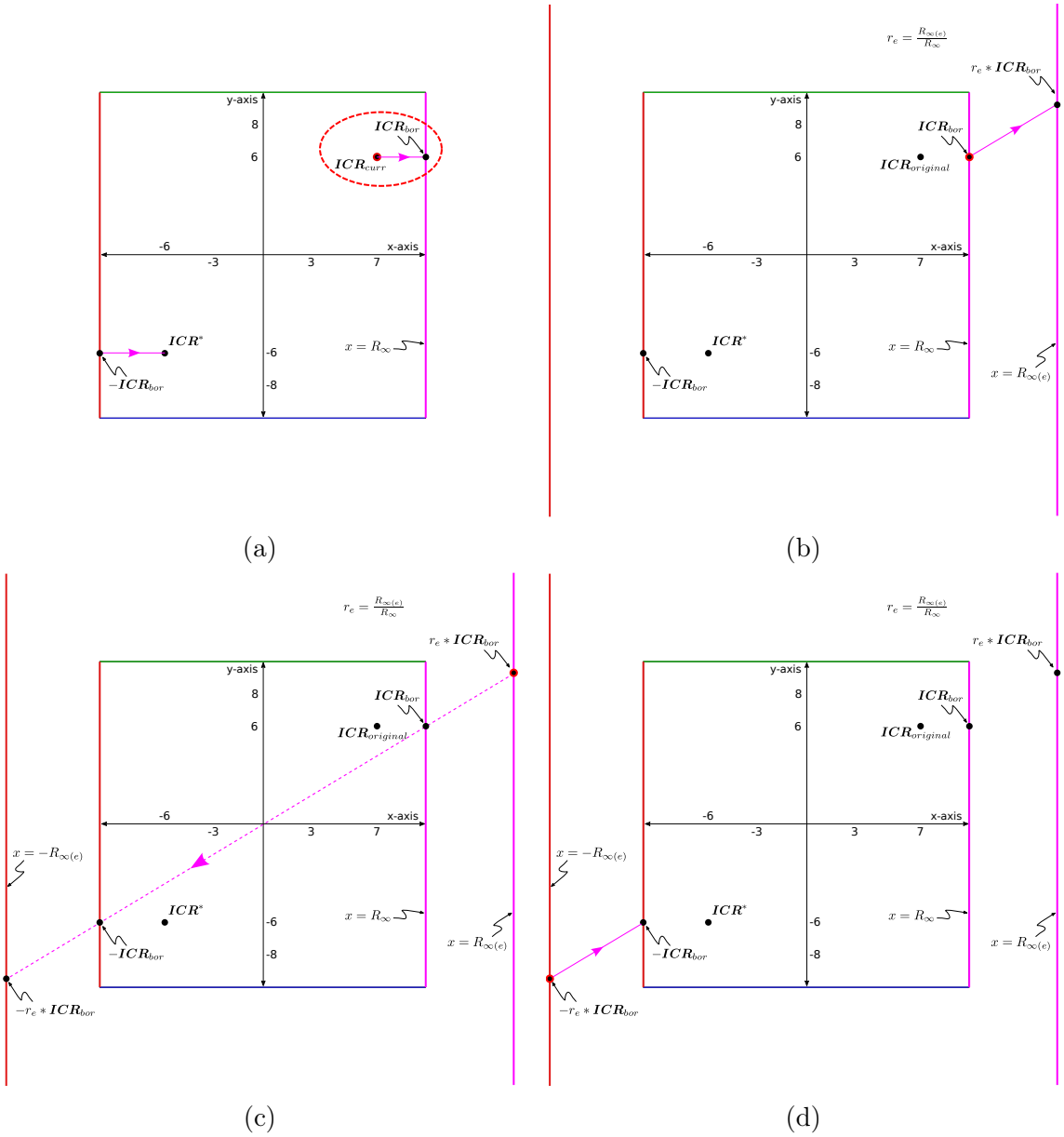


Figure 5.8: Implementation of the complementary route; (a) step1: moving to the optimum border point, (b) step2: extending the border lines, (c) border switching in one sample period, and (d) retracting border line, then completion as direct route.

$$\forall S_1 \oplus S_2 \wedge F_{ce} \wedge (|\boldsymbol{ICR}_{err(e)}| > e_{th}) :$$
$$\begin{cases} \boldsymbol{ICR}^*_{mod} = \boldsymbol{ICR}_{bor} * r_e, \\ \lambda = \lambda_e, \end{cases} \tag{5.5}$$
$$Otherwise : S_2 = 0,$$

$$\forall S_1 \oplus S_2 \oplus S_3 \wedge F_{ce} \wedge (\boldsymbol{ICR}_{curr} \neq \boldsymbol{ICR}'_{bor}) :$$
$$\begin{cases} \boldsymbol{ICR}^*_{mod} = \boldsymbol{ICR}'_{bor} * r_e, \end{cases} \tag{5.6}$$
$$Otherwise : S_3 = 0,$$

$$\forall S_1 \oplus S_2 \oplus S_3 \oplus S_4 \wedge F_{ce} \wedge (|\boldsymbol{ICR}'_{err(c)}| > c_{th}) :$$
$$\begin{cases} \boldsymbol{ICR}^*_{mod} = \boldsymbol{ICR}'_{bor}, \\ \lambda = \lambda_n, \end{cases}$$
$$Otherwise : \tag{5.7}$$
$$\begin{cases} \boldsymbol{ICR}^*_{mod} = \boldsymbol{ICR}^*, \\ S_4 = 0, \\ F_{ce} = 0. \end{cases}$$

In the above, $r_e = R_{\infty(e)}/R_{\infty}$ is the extension ratio, $\boldsymbol{ICR}_{err(c)} = \boldsymbol{ICR}_{bor} - \boldsymbol{ICR}_{curr}$, and $\boldsymbol{ICR}_{err(e)} = \boldsymbol{ICR}_{bor} * r_e - \boldsymbol{ICR}_{curr}$ are the complementary and extended error vectors respectively, e_{th}, and c_{th} are the corresponding positive scalar thresholds. Two positive scalar gains λ_e, and λ_n are used for the extended and normal ICR motion controllers respectively. Finally, $\boldsymbol{ICR}'_{err(c)} = \boldsymbol{ICR}'_{bor} - \boldsymbol{ICR}_{curr}$. Equations (5.4 - 5.7) represent the semantic core (planning) of the complementary route algorithm, where the high level desired signal $\boldsymbol{ICR}^*$ is modified to $\boldsymbol{ICR}^*_{mod}$ based on the choice of ICR route to follow. The modified desired signal will be used by the ICR controller [SCFP17] (refer to Fig. 5.9):

$$\begin{aligned} \dot{\boldsymbol{ICR}}_{ref} &= \dot{\boldsymbol{ICR}}^* + \lambda \boldsymbol{ICR}_{err}, \\ \boldsymbol{ICR}_{err} &= \boldsymbol{ICR}^*_{mod} - \boldsymbol{ICR}_{curr}. \end{aligned} \tag{5.8}$$

5.2.4 Automatic Joint Space Border Switching

In (5.4), we modify the desired ICR so that it is on the complementary border, but the actual border switching is done in the steering joint space. The switching is executed by first checking if the $\boldsymbol{ICR}^*_{mod}$ and the $\boldsymbol{ICR}_{curr}$ satisfy:

$$(|X^*_{mod} \pm R_{\infty(e)}| \wedge |X_{curr} \mp R_{\infty(e)}|) \vee (|Y^*_{mod} \pm R_{\infty(e)}| \wedge |Y_{curr} \mp R_{\infty(e)}|) \leq o_{band}. \tag{5.9}$$

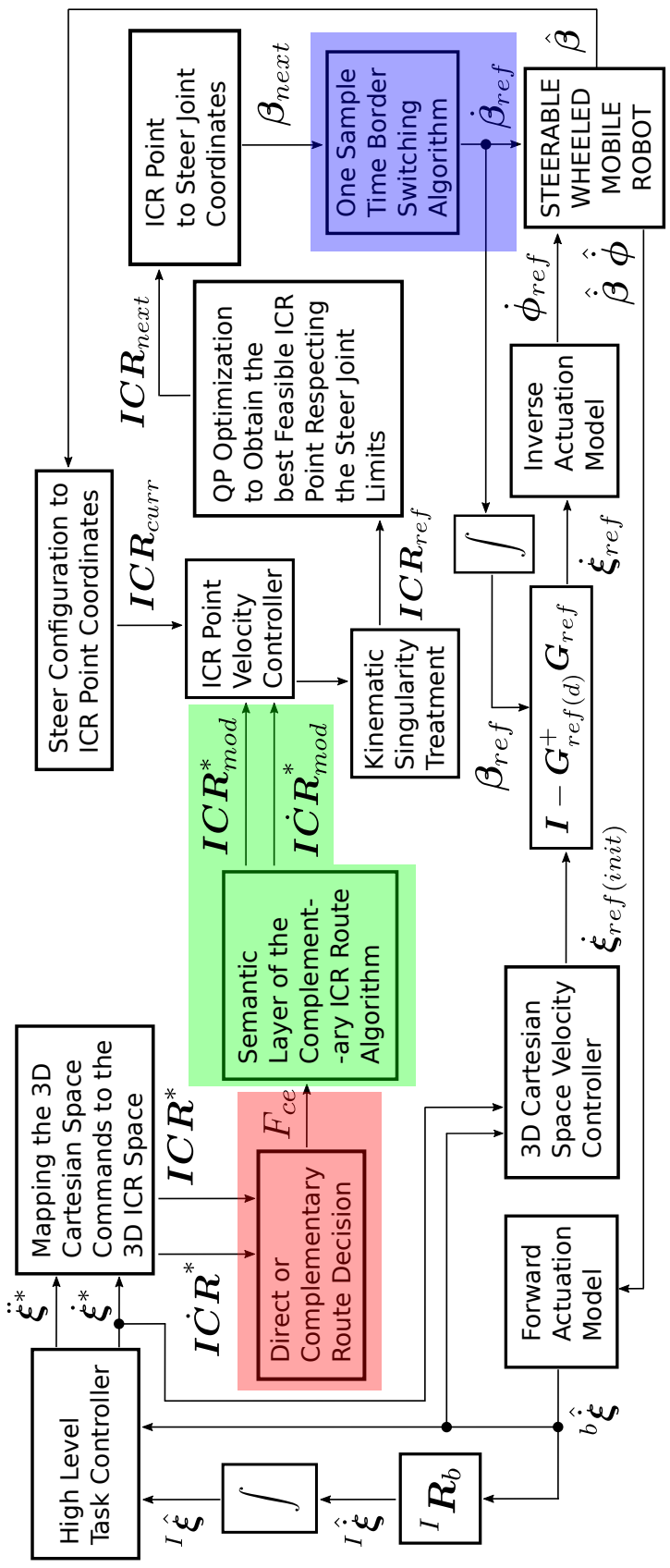


Figure 5.9: Motion-discontinuity robust controller architecture with the complementary controller addition shown in colored blocks.

This simply indicates that at least one component (being x or y) of both the desired and current ICR points is in the opposite border band. Then the complementary steer joint variables corresponding to $\boldsymbol{ICR}'_{bor} = -\boldsymbol{ICR}_{bor}$ are evaluated using the SWMR geometric model:

$$\beta'_{i(bor)} = \arctan 2\Big(-Y_{bor} - h_{yi}, -X_{bor} - h_{xi}\Big) - \frac{\pi}{2}. \tag{5.10}$$

However we are interested in the minimum change in steering angle to achieve the border switch, which can be obtained by first computing the required steering change $\delta\beta_{i(bs)}$ using (5.10) and the current steer joint coordinates:

$$\delta\beta_{i(bs)} = PosB(\beta'_{i(bor)}) - PosB(\beta_{i(curr)}), \tag{5.11}$$

$$PosB(\delta\beta_{i(bs)}) = \begin{cases} \delta\beta_{i(bs)} + 2\pi, & \forall\delta\beta_{i(bs)} < 0 \\ \delta\beta_{i(bs)} - 2\pi, & \forall\delta\beta_{i(bs)} \geq 2\pi \end{cases},$$

where $PosB(*)$ is a function that ensures the output angles are positive and bounded in the range $[0, 2\pi[$. The minimum change is then obtained from (5.11) as follows:

$$\delta\beta_{i(bs)(min)} = \begin{cases} \delta\beta_{i(bs)} - 2\pi, & \text{if } \delta\beta_{i(bs)} > 3\pi/2 \\ \delta\beta_{i(bs)} - \pi, & \text{if } \delta\beta_{i(bs)} > \pi/2 \\ \delta\beta_{i(bs)} + 2\pi, & \text{if } \delta\beta_{i(bs)} < -3\pi/2 \\ \delta\beta_{i(bs)} + \pi, & \text{if } \delta\beta_{i(bs)} < -\pi/2 \end{cases}. \tag{5.12}$$

Finally, the $4D$ vector $\delta\boldsymbol{\beta}_{(bs)(min)}$ is added to the current steer position vector to conclude the border switching phase:

$$\boldsymbol{\beta}_{ref} = \boldsymbol{\beta}_{curr} + \delta\boldsymbol{\beta}_{(bs)(min)}. \tag{5.13}$$

5.3 Direct OR Complementary

In order to select the best route, here defined as the one that is more time efficient, we need an estimate of the total time required by each. To do this we use the Jacobian relating the steering velocity to that of the ICR point:

$$\dot{\boldsymbol{ICR}}_{max} = \boldsymbol{J}^{+}_{(\rho)}(\boldsymbol{ICR}_{curr}) * \dot{\boldsymbol{\beta}}_{max}, \tag{5.14}$$

$$\boldsymbol{J}(\boldsymbol{ICR}) = \begin{bmatrix} \boldsymbol{J}_1 & \dots & \boldsymbol{J}_N \end{bmatrix}^T, \tag{5.15}$$

$$\boldsymbol{J}_i = \begin{bmatrix} \dfrac{-\widetilde{Y}}{\widetilde{X}^2 + \widetilde{Y}^2 + \delta_3} & \dfrac{\widetilde{X}}{\widetilde{X}^2 + \widetilde{Y}^2 + \delta_3} \end{bmatrix},$$

where $\widetilde{Y} = (Y - h_{yi})$, $\widetilde{X} = (X - h_{xi})$, with $\delta_3 \in \mathbb{R}^+$ a damping factor. The Jacobian matrix in (5.14) is obtained by substituting by $\boldsymbol{ICR}_{curr}$ in the general relation (5.15). Fig. 5.10 shows the evolution of $\|\dot{\boldsymbol{ICR}}_{max}\|_2$ over the entire R_∞ bounded ICR space, computed using (5.14) with the parameters of the Neobotix MPO-700 mobile robot provided in Table 5.1. As depicted, among equal length direct and complementary routes, the latter will be more time efficient. Both routes are then divided into an equal, even number of samples $2 * n_p$, $n_p \in \mathbb{Z}^+$, and at each, $\dot{\boldsymbol{ICR}}_{max}$ is computed using (5.14) and used to estimate the time required by each route as follows:

$$
\begin{aligned}
T_d &= \sum_{k=1}^{2*n_p} \frac{S_d}{\|\boldsymbol{J}^+_{(\rho)}(\boldsymbol{ICR}_{k(d)}) * \dot{\boldsymbol{\beta}}_{max}\|}, \\
T_c &= \sum_{k=1}^{n_p} \frac{S_c}{\|\boldsymbol{J}^+_{(\rho)}(\boldsymbol{ICR}_{k(c)}) * \dot{\boldsymbol{\beta}}_{max}\|} \\
&\quad + \frac{S'_c}{\|\boldsymbol{J}^+_{(\rho)}(\boldsymbol{ICR}'_{k(c)}) * \dot{\boldsymbol{\beta}}_{max}\|},
\end{aligned}
\tag{5.16}
$$

$$
\begin{aligned}
\boldsymbol{ICR}_{k(d)} &= \boldsymbol{ICR}_{curr} + \frac{k}{2 * n_p} * \boldsymbol{ICR}_{err}, \\
\boldsymbol{ICR}_{k(c)} &= \boldsymbol{ICR}_{curr} + \frac{k}{n_p} * \boldsymbol{ICR}_{err(c)}, \\
\boldsymbol{ICR}'_{k(c)} &= \boldsymbol{ICR}'_{bor} + \frac{k}{n_p} * (\boldsymbol{ICR}^* - \boldsymbol{ICR}'_{bor}),
\end{aligned}
$$

where T_d, and T_c are the time duration estimates, $S_d = 0.5 * \|\boldsymbol{ICR}_{err}\|/n_p$, $S_c = \|\boldsymbol{ICR}_{err(c)}\|/n_p$, and $S'_c = \|\boldsymbol{ICR}^* - \boldsymbol{ICR}'_{bor}\|/n_p$ are the distance increments for the direct, complementary part#1, and part#2 routes respectively (refer to Fig. **??**, upper). It is worth noting that the estimates provided by (5.16) are not exact and are only useful for comparison purposes, since the error driven ICR controller will not provide maximum ICR velocity during the whole route.

In some cases, even if the complementary ICR route is found more time consuming than its direct counterpart, it is preferable to still use it. In particular when the direct route will pass by/near singular configurations, triggering the singularity avoidance algorithms [SCFP17], that in turn will consume more time by manipulating the direct route. Moreover, in the neighbourhood of singular configurations or of the robot geometric center, usually the feasible region for the ICR_{next} is very small, and it might happen (although rarely if controller parameters are tuned well) that no feasible solution is found. In practice, such situation occurs when the direct route passes by

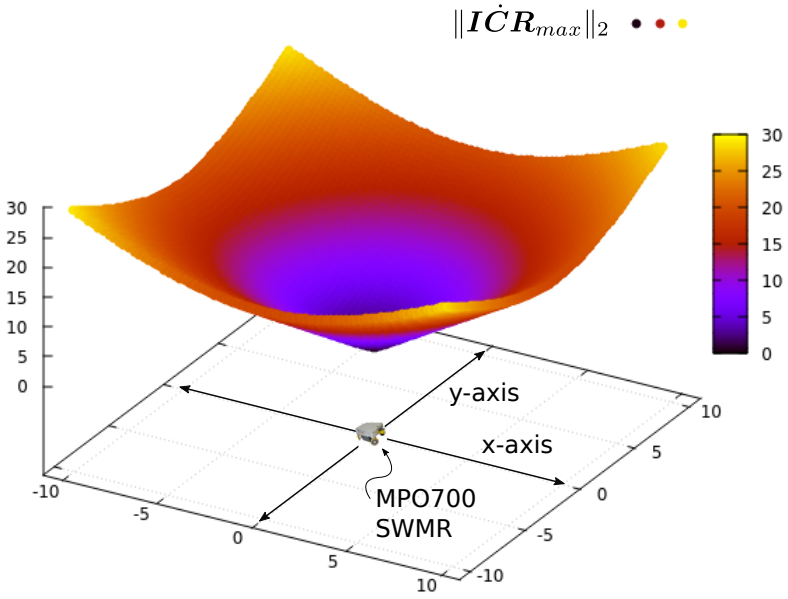


Figure 5.10: Variation of the maximum ICR velocity with robot's ICR position, featuring the Neobotix-MPO700 industrial mobile robot approximately to scale, with $R_\infty = 10m$.

the robot footprint. To bias the decision in such cases, we construct an elliptic footprint, light red colored in Fig. 5.11, whose geometric center coincides with the robot frame, and semiaxes lengths w, and h (values chosen arbitrarily), formulated as:

$$x^2/w^2 + y^2/h^2 = 1. \tag{5.17}$$

The intersection points $\boldsymbol{P}_{int(1,2)} = \begin{bmatrix} x_{int(1,2)} & y_{int(1,2)} \end{bmatrix}^T$ of such footprint with the direct route formulated as $y = m_d * x + b_d$, would be:

$$x_{int(1,2)} = \frac{-w^2 m_d b_d \pm wh\sqrt{K_1 - b_d^2}}{K_1},$$

$$y_{int(1,2)} = \frac{h^2 b_d \pm whm_d\sqrt{K_1 - b_d^2}}{K_1},$$

where $K_1 = w^2 m_d^2 + h^2$. Then we use the discriminant $D = K_1 - b_d^2$ to determine whether the direct route intersects the designed footprint or not ($D \geq 0$ if intersection occurs). Afterwards, we add a biasing factor T_b that is activated only if both the current and desired ICR points are outside the ellipse. Such biasing condition is checked by substituting by $\boldsymbol{ICR}_{curr}$, and $\boldsymbol{ICR}^*$ in (5.17) and formulated as:

$$C_b = \begin{cases} 1, & \text{if } \left(\frac{X_{curr}^2}{w^2} + \frac{Y_{curr}^2}{h^2} \geq 1\right) \wedge \left(\frac{X^{*2}}{w^2} + \frac{Y^{*2}}{h^2} \geq 1\right) \\ 0, & \text{Otherwise} \end{cases}.$$

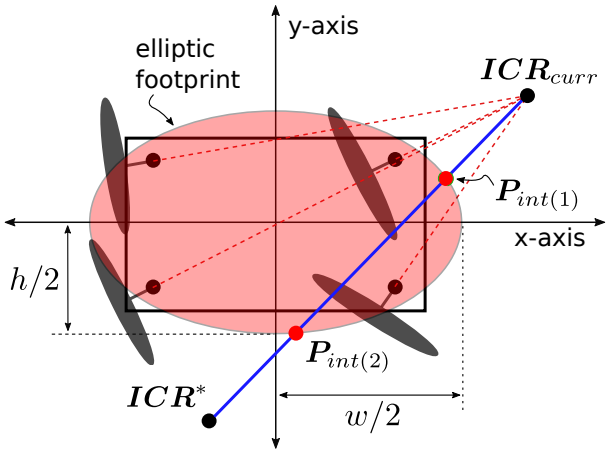


Figure 5.11: Intersection of the direct ICR route with the elliptic footprint of the mobile robot should be in favor of using the complementary route.

Finally, the total time duration based route comparison is done using:

$$F_{ce} = \begin{cases} 1, & \text{if } (T_c \leq T_d + l(D)C_bT_b) \wedge \dot{\boldsymbol{ICR}}^* \neq 0 \\ 0, & \text{Otherwise} \end{cases}, \tag{5.18}$$

where $l(*)$ is a logic function:

$$l(D) = \begin{cases} 1, & \forall D > 0 \\ 0, & \forall D \leq 0 \end{cases}.$$

In (5.18), condition $\dot{\boldsymbol{ICR}}^* \neq 0$, ensures that we evaluate both routes and update the flag F_{ce} automatically only if a change in the desired ICR motion is detected, triggering the "direct or complementary" decision making algorithm (red colored block in Fig. 5.9).

5.4 Experiments

In this section, we describe/perform two experiments, the first (simulation) will show how the proposed complementary route controller can enhance the performance of the SWMR in response to diverse discontinuous velocity commands. To this end we introduce the *command fulfillment index* as performance (quantitative) evaluation metric. The second experiment will feature a particular (common) application where the mobile base is required to reach certain relative pose with respect to a target object, avoiding obstacles while maintaining a particular heading angle so that the target is always in the vision cone. We show that the general performance is greatly enhanced with the proposed controller. Both, the vision based controller and the

obstacle avoidance algorithm are out of the scope of this work and consequently not detailed here, and the reader is referred to [KCAP17] for more information on these.

5.4.1 Command Fulfillment Index

To better assess the quality of the required commands $\dot{\boldsymbol{\xi}}_{ref}$ we introduce the CFI (dimensionless quantity) as metric for evaluation, formulated as:

$$CFI = 1 - \frac{\|\dot{\boldsymbol{\xi}}^* - \dot{\boldsymbol{\xi}}_{ref}\|_2}{2\|\dot{\boldsymbol{\xi}}_{max}\|_2}, \tag{5.19}$$

where $\dot{\boldsymbol{\xi}}_{max} = \begin{bmatrix} \dot{x}_{max} & \dot{y}_{max} & \dot{\theta}_{max} \end{bmatrix}^T$ is the maximum robot velocity allowed via linear thresholding the reference signal $\dot{\boldsymbol{\xi}}_{ref}$. In (5.19), the CFI metric does not depend on the estimated (from joint space measurements) robot velocity, since the low level (joint dynamics) control is out of the scope of this work. Instead we aim at evaluating how much the generated, feasible velocity commands will follow the desired one, where in case of SWMR, the value $0 < CFI < 1$ will be closer to 1 as the quality of the controller increases.

It is worth noting that the best values for CFI are expected for a fully omnidirectional robot. However, it will not maintain $CFI = 1$ at all times being subject to discontinuous velocity commands: a sudden drop in the CFI value will happen once a new (discontinuous) desired ICR point is commanded (see Fig. 5.13c, and Fig. 5.15c). It is generally expected that a FOMR will converge faster than a SWMR. Note that the CFI formulation in (5.19) depends on the response time of both the steering and driving joints. The FOMR structure has an advantage since it does not need steering. On the other hand, a SWMR structure has an advantage in the driving joint thanks to its conventional wheels that are generally expected to respond faster (to reach a discontinuously commanded drive speed) than omnidirectional (or Swedish) ones, due to their limited velocity/acceleration capabilities [WCGR02], and frequent slippage [NTST00]. This argument implies that for applications subject to frequent discontinuous velocity commands, the SWMR structure may be equivalent or even out-perform the FOMR one.

5.4.2 Discontinuous Velocity Commands Simulation

In order to evaluate the proposed controller against discontinuous robot velocity commands in different situations, we should be able to set those commands in the ICR

space. To do this we use the kinematic relation between the $3D$ Cartesian velocity space and the $2D$ ICR space (refer to Fig. 3.1):

$$\boldsymbol{v}^* = \begin{bmatrix} \dot{x}^* \\ \dot{y}^* \end{bmatrix} = \begin{bmatrix} Y^*\dot{\theta}^* \\ -X^*\dot{\theta}^* \end{bmatrix}, \tag{5.20}$$

where X^*, and Y^* are the components of each of the nine desired ICR points shown in Fig. 5.12, and a constant rotational velocity $\dot{\theta}^* = 0.05 rad/s$ is used throughout the simulation. Each of the 9 velocity commands is applied for a duration of 5 seconds. The $3D$ Cartesian velocity vector $\dot{\boldsymbol{\xi}}^* = \begin{bmatrix} \boldsymbol{v}^{*T} & \dot{\theta}^* \end{bmatrix}^T$ is then sent to the controller architecture shown in Fig. 5.9.

Simulation results for the complementary route controller and the conventional one are shown in Fig. 5.13, 5.14, and Fig. 5.15, 5.16 respectively. The first command $\boldsymbol{ICR}^* = \begin{bmatrix} 0 & R_\infty \end{bmatrix}^T$, applied during $t = [0, 5[s$ (t is the simulation time), shown in Fig. 5.12 requires no steering as it conforms with the initial steer joint configuration. Among numerous reasons, the motivation behind the first command is to show that even if no steering is required (the same case as in FOMR), the CFI is not equal to 1 (Fig. 5.13c, 5.15c) since it depends on how fast the required driving velocity can be reached, and in that spirit it is expected that the SWMR might out-perform the FOMR structure at some motion requirements, due to the limited performance capabilities [WCGR02, NTST00] of such non conventional wheel types.

The second command $\boldsymbol{ICR}^* = \begin{bmatrix} 0 & -R_\infty \end{bmatrix}^T$, applied during $t = [5, 10[s$, requires ICR motion from one extreme of the ICR space to the opposite in the conventional (direct) controller. The ability to perform such discontinuous velocity command in short time is the core motivation behind the complementary route development, since it is very common in heading control applications, where a small variation about the

Table 5.1: Robot and Controller parameters used in the complementary route experiments.

δ_1	1^{-9}	$\dot{V}_{max}$	$10m/s$	β_{th}	$0.005 rad.$
λ_n	7.7	R_{zone}	$0.015m$	$\dot{\beta}_{max}$	$2 rad./s$
$\ddot{\beta}_{max}$	$25 rad./s^2$	t_s	$25ms$	K_p	2
w	$0.4m$	h	$0.3m$	λ_e	30
$\dot{x}_{max}$	$0.5m/s$	$\dot{y}_{max}$	$0.5m/s$	$\dot{\theta}_{max}$	$0.5 rad./s$
$\ddot{x}_{max}$	$0.5m/s^2$	$\ddot{y}_{max}$	$0.5m/s^2$	$\ddot{\theta}_{max}$	$0.5 rad./s^2$
$R_{\infty(e)}$	$33m$	e_{th}	$0.1R_\infty$	c_{th}	$0.05R_{\infty(e)}$
δ_3	0.01	n_p	50	T_b	$3s$

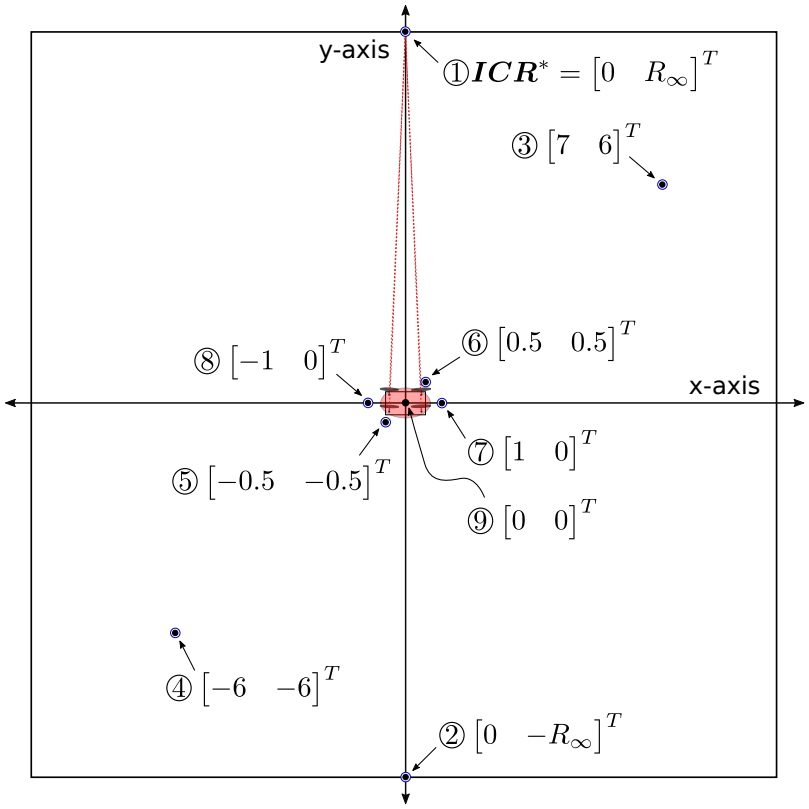


Figure 5.12: Desired (discontinuous) ICR points used in simulation. The relative dimensions of the mobile base schematic as well as the points are drawn at approximately 1 : 1 scale for clarification. The initial steer joint configuration corresponds to the equivalent ICR point of the first velocity command vector.

desired heading will result in sign change in the rotational velocity command, that in turn cause big jumps in the corresponding ICR from one extreme to the other. The X component of the desired and reference ICR points, for the complementary and direct routes is shown in Fig. 5.13a, 5.15a respectively. Similarly for the Y component in Fig. 5.13b, 5.15b. Recall that the reference ICR point is a feasible signal that can be performed by the robot taking into account its maximum performance limits. By comparing the Y_{ref} output of both controllers in Fig. 5.13b, 5.15b we clearly see the enhanced performance using the complementary route, where converging to the Y^* took only fraction of a second ($0.2s$), whereas in Fig. 5.15b, the direct route took much longer to converge ($2.27s$).

The third $\boldsymbol{ICR}^* = \begin{bmatrix}7 & 6\end{bmatrix}^T$, and fourth $\boldsymbol{ICR}^* = \begin{bmatrix}-6 & -6\end{bmatrix}^T$ commands applied during $t = [10, 15[s$, $t = [15, 20[s$ respectively, correspond to the example given earlier in Fig. 5.4. These commands confirm the superior performance of the complementary route (refer to Fig. 5.13a, 5.15a, 5.13b, 5.15b) in terms of the convergence time of both the X and Y components of $\boldsymbol{ICR}_{ref}$. The 4 commands just described feature

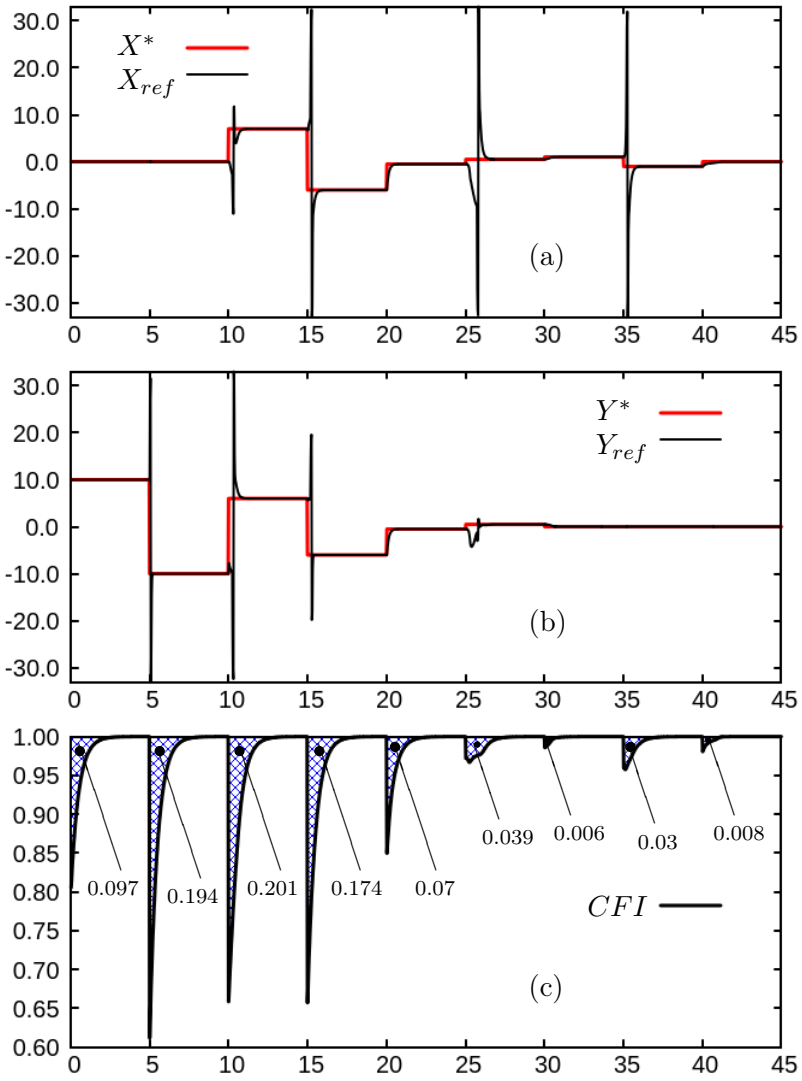


Figure 5.13: Simulation results using the complementary route controller with (a), (b) the $x-axis$, and $y-axis$ components respectively of the desired and reference (output of the controller) ICR points (in *meters*), (c) the command fulfillment index (*dimensionless*). In all plots, the abscissa represents the simulation time instant (in *seconds*).

large discontinuous jumps in the desired ICR point, in which cases the performance of the complementary route clearly surpasses that of the direct route in terms of lowering both the time of convergence and the total Cartesian velocity error. The latter is evident from the smaller area above curve for the CFI plots in Fig. 5.13c as compared to Fig. 5.15c.

The fifth command $\boldsymbol{ICR}^* = \begin{bmatrix} -0.5 & -0.5 \end{bmatrix}^T$, applied during $t = [20, 25[s$, lies in the same quadrant as the fourth. Consequently there exists no advantage in using the complementary route; this is automatically detected using (5.16), and the direct route

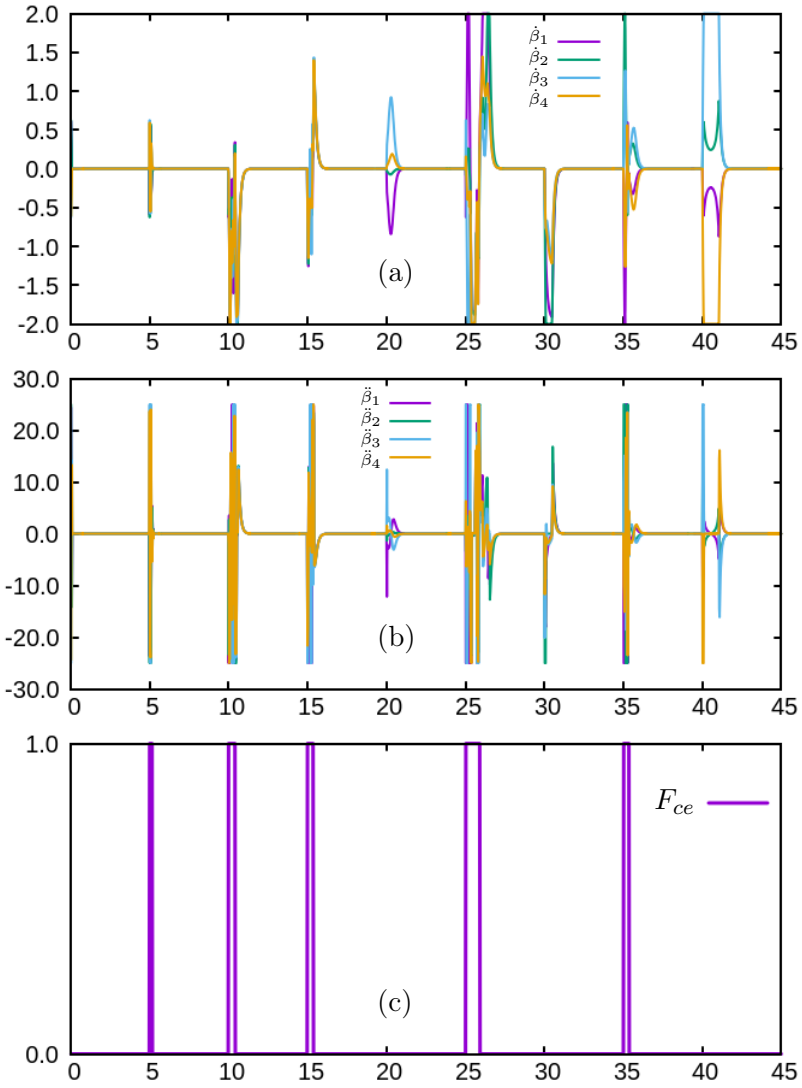


Figure 5.14: Simulation results using the complementary route controller with (a), (b) the steering velocity (in rad/s) and acceleration (in rad/s^2) respectively, and (c) the time instants at which the complementary route algorithm is active. In all plots, the abscissa represents the simulation time instant (in *seconds*).

is used. The sixth command $\boldsymbol{ICR}^* = \begin{bmatrix}0.5 & 0.5\end{bmatrix}^T$, applied during $t = [25, 30[s$, aims at testing the elliptic footprint avoiding algorithm. By comparing Fig. 5.15, and Fig. 5.13 we see that the $\boldsymbol{ICR}_{ref}$ converging time for both is almost the same, although the CFI is lower in case of the direct route. This can be adjusted by lowering the value of the penalty factor T_b in (5.18). Instead, here, its value is intentionally left high enough ($3s$) to bias the decision in favor of the complementary route for two reasons. First: in practice, when the desired velocity is near the current one (especially near the robot geometric center), a small increase in the error value does not affect the

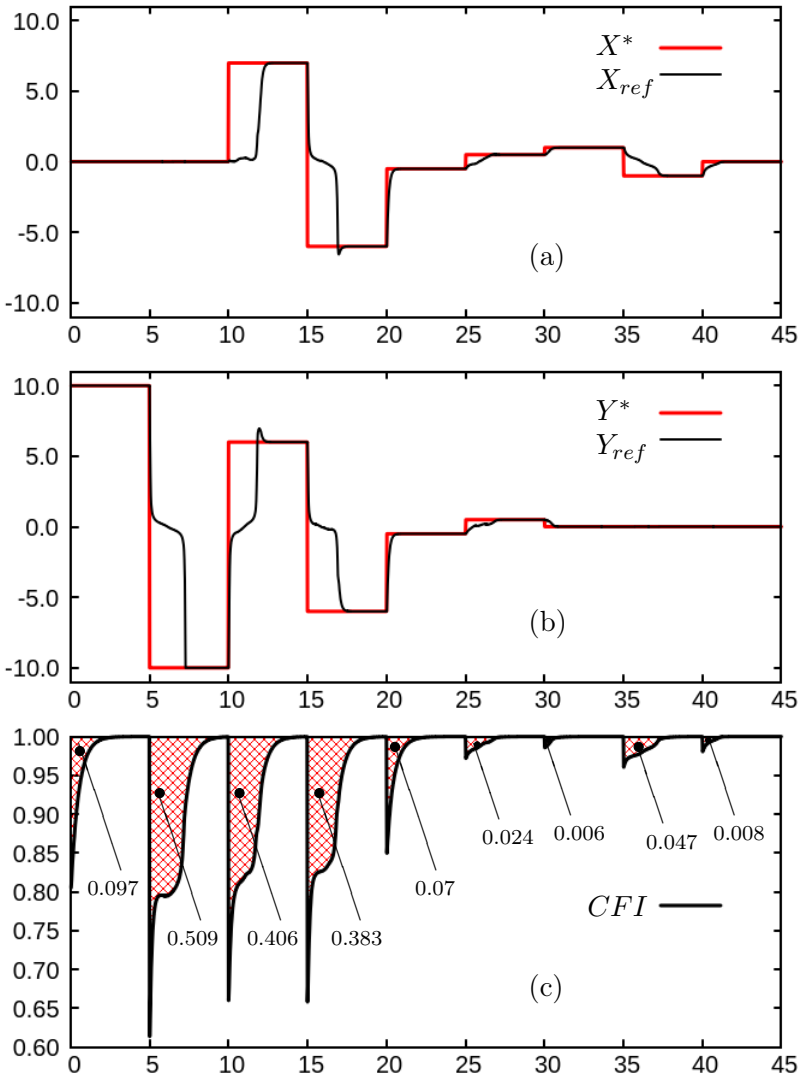


Figure 5.15: Simulation results using direct route (conventional) controller with (a), (b) the $x-axis$, and $y-axis$ components respectively of the desired and reference (output of the controller) ICR points (in *meters*), (c) the command fulfillment index (*dimensionless*). In all plots, the abscissa represents the simulation time instant (in *seconds*).

application in hand tragically. Second: avoiding ICR motion in the robot footprint region lowers the chances of triggering singularity avoidance algorithms, keeping the steering to minimum as shown by comparing Fig. 5.14a, 5.14b with Fig. 5.16a, 5.16b, which is generally favorable.

The seventh $\boldsymbol{ICR}^* = \begin{bmatrix}1 & 0\end{bmatrix}^T$, and eighth $\boldsymbol{ICR}^* = \begin{bmatrix}-1 & 0\end{bmatrix}^T$ commands applied during $t = [30, 35[s$, $t = [35, 40[s$ respectively, show that the complementary route can be as efficient for small ICR jumps near the geometric center as it is for the far points in the ICR space. In moving from the seventh to the eighth, the conversion

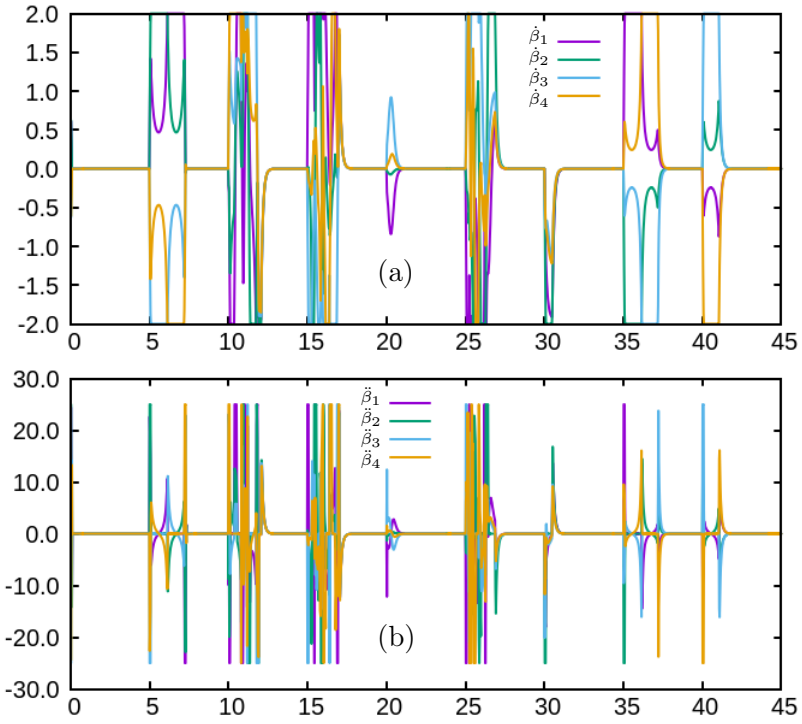


Figure 5.16: Simulation results using direct route (conventional) controller with (a), (b) the steering velocity (in rad/s) and acceleration (in rad/s^2) respectively. In all plots, the abscissa represents the simulation time instant (in *seconds*).

time duration is approximately $2.5s$ and $1s$ for the direct and complementary routes respectively. The corresponding areas above the CFI curve are $0.047s$, and $0.03s$ in the same order. Finally a pure rotation velocity command $\boldsymbol{ICR}^* = \begin{bmatrix}0 & 0\end{bmatrix}^T$ is applied during $t = [40, 45[s$. In Fig. 5.14c, the periods in which the complementary route algorithm is active are shown. During the rest of the simulation period, the direct route is active, better explaining the proposed algorithm as a correcting module to enhance the overall performance. Aside better CFI values and shorter convergence time of $\boldsymbol{ICR}_{ref}$ to the $\boldsymbol{ICR}^*$, the proposed controller reduces the required steering velocity and acceleration, as clearly shown in Fig. 5.14a, 5.14b, 5.16a, and 5.16b.

5.4.3 Practical Application Scenario and Results

The performance evaluation detailed in previous subsection (Fig. 5.13, 5.14, and Fig. 5.15, 5.16) is a mere quantitative comparison. It shows that the complementary route controller outperforms the direct route one. However in practice, the direct route controller might fail in several applications or will provide unsatisfactory behavior at best. Here, we describe/perform one of these applications, where a mobile robot is required to move to a relative pose with respect to a visible target, while

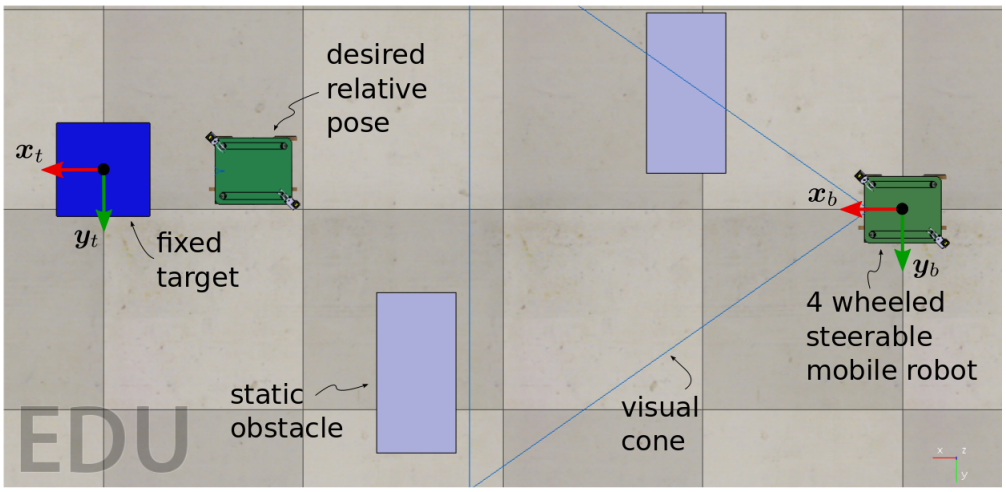


Figure 5.17: V-rep scene of the practical scenario, featuring the Neobotix-MPO700 mobile robot navigating to a target while performing static obstacle avoidance. The simulated visual cone applies a constraint on the heading of the robot so that the target is always seen by the robot, as is common in vision-based navigation. Here we apply such constraint to show the utility of the proposed controller. $\boldsymbol{\mathcal{F}}_t = (o_t, \boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{z}_t)$ denotes the target object frame.

avoiding several static obstacles on its way. During this task, the robot has to maintain a fixed relative heading angle with respect to the target, to maintain it within its field of view. The V-rep simulation scene of such application is shown in Fig. 5.17. The desired velocity commands $\dot{\boldsymbol{\xi}}^*$ output by the visual navigation controller (detailed in [KCAP17]), are highly discontinuous. Real experiments have been conducted successfully using the complementary route controller.

The desired velocity (output by the navigation controller) is shown in Fig. 5.18a and Fig. 5.20a for the complementary and direct route controllers, respectively. In Figures 5.18b, and 5.20b, the output of the respective controllers is shown. As seen, the complementary route approach is more responsive and converges faster to the desired signal. However during the time period $t = [2, 5[s$ in Fig. 5.18b, the complementary route algorithm is less satisfactory in terms of following the desired robot velocity. Hence, during this period, the direct route is activated. The reason for this can be found in Fig. 5.18c, and 5.18d showing the X and Y components of the $\boldsymbol{ICR}^*$, and $\boldsymbol{ICR}_{ref}$ signals respectively. During this time period, the $\boldsymbol{ICR}^*$ is near the robot geometric center (values of X^*, and Y^* are closer to 0), consequently the direct route is computed to be more time efficient using (5.16), and since the signal is highly discontinuous, the direct route controller output $\boldsymbol{ICR}_{ref}$, did not have enough time to converge to $\boldsymbol{ICR}^*$. It is worth noting that whenever a new discontinuous velocity command is requested, all robot velocity components are affected until they

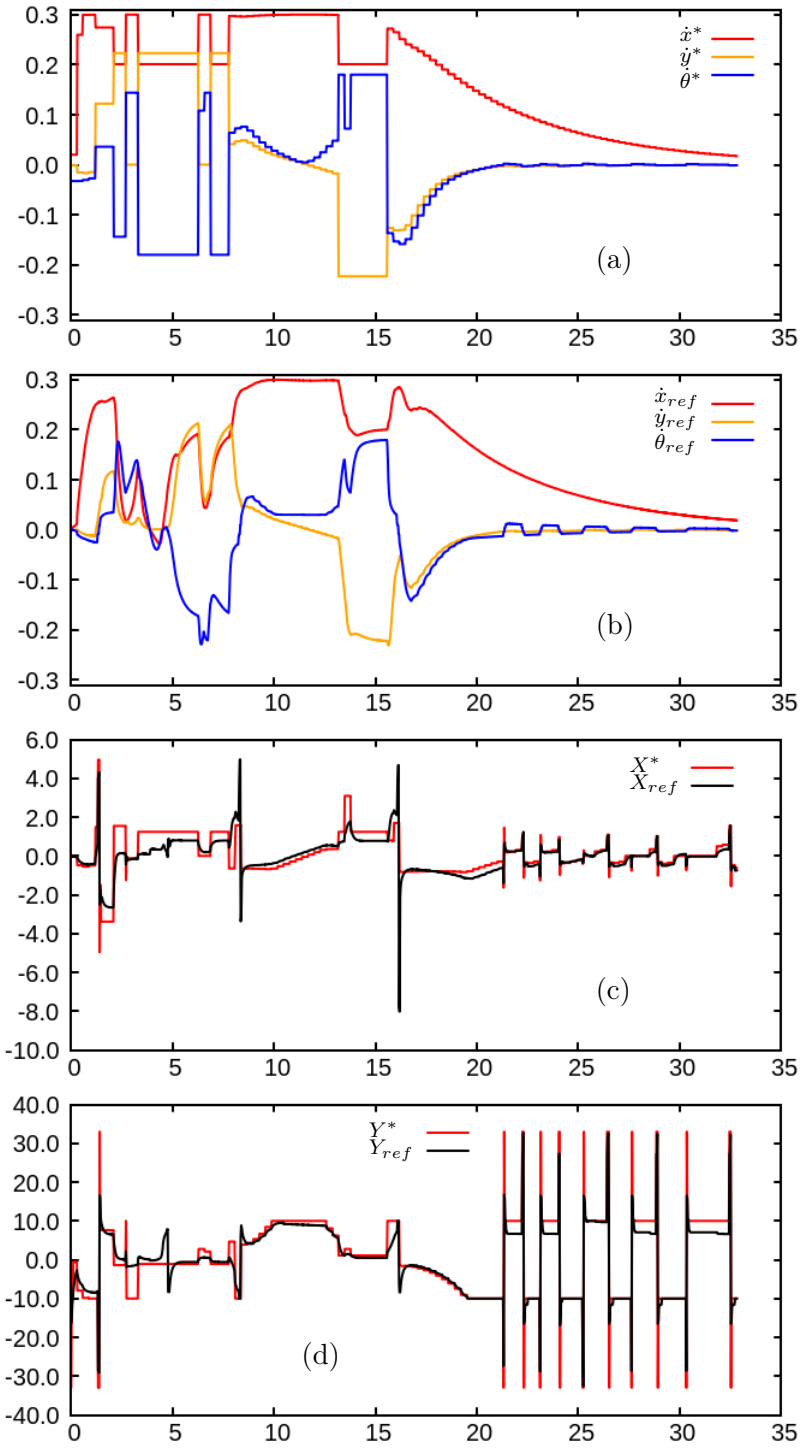


Figure 5.18: Simulation results of the practical application using the complementary route controller with (a), (b), the desired and reference (output of the controller) $3D$ robot velocity respectively (linear velocity in m/s and angular velocity in rad/s), (c), (d), the $x-axis$, and $y-axis$ components respectively of the desired and reference ICR points (in $meters$).

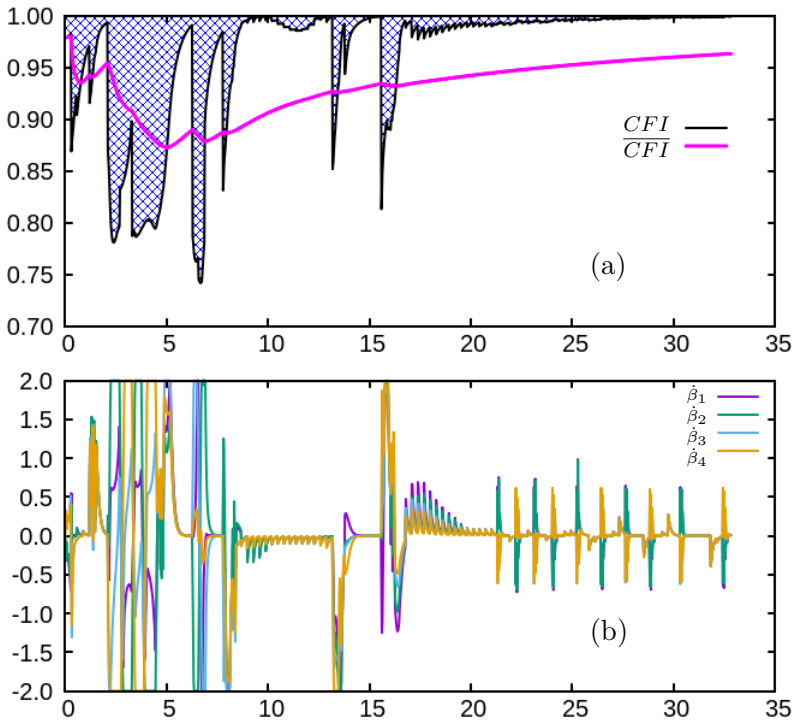


Figure 5.19: Simulation results of the practical application using the complementary route controller with (a) the command fulfillment index (*dimensionless*) and the evolution of its average value $\overline{CFI}$, (b) the corresponding steering velocity (in *rad/s*).

converge to their corresponding desired values. This is because, in contrast with FOMR structured robots, the robot velocity components are coupled.

The X and Y components of the $\boldsymbol{ICR}^*$, and $\boldsymbol{ICR}_{ref}$ signals for the direct route controller are shown in Fig. 5.20c, and 5.20d respectively. By comparing these figures with their complementary route counterparts, the poor responsiveness of the first is evident, especially at pure heading control during the time period $t = [25, 48[s$. The controller output $\boldsymbol{ICR}_{ref}$ takes longer time to converge to the $\boldsymbol{ICR}^*$, which in turn is reflected on the longer simulation time required to reach its target relative pose ($48s$).

The command fulfillment index is shown in Fig. 5.19a, and 5.21a respectively in the usual order, along with the evolution of its moving average value $\overline{CFI}$ computed using:

$$\overline{CFI} = \sum_{0}^{N_s} \frac{CFI}{N_s},$$

where $N_s = t/t_s$ is the number of sampling instances (points) at time t. The CFI metric quantitatively shows the enhanced responsiveness discussed earlier for the complementary route controller. Recall that the area above the CFI curve (in hashed

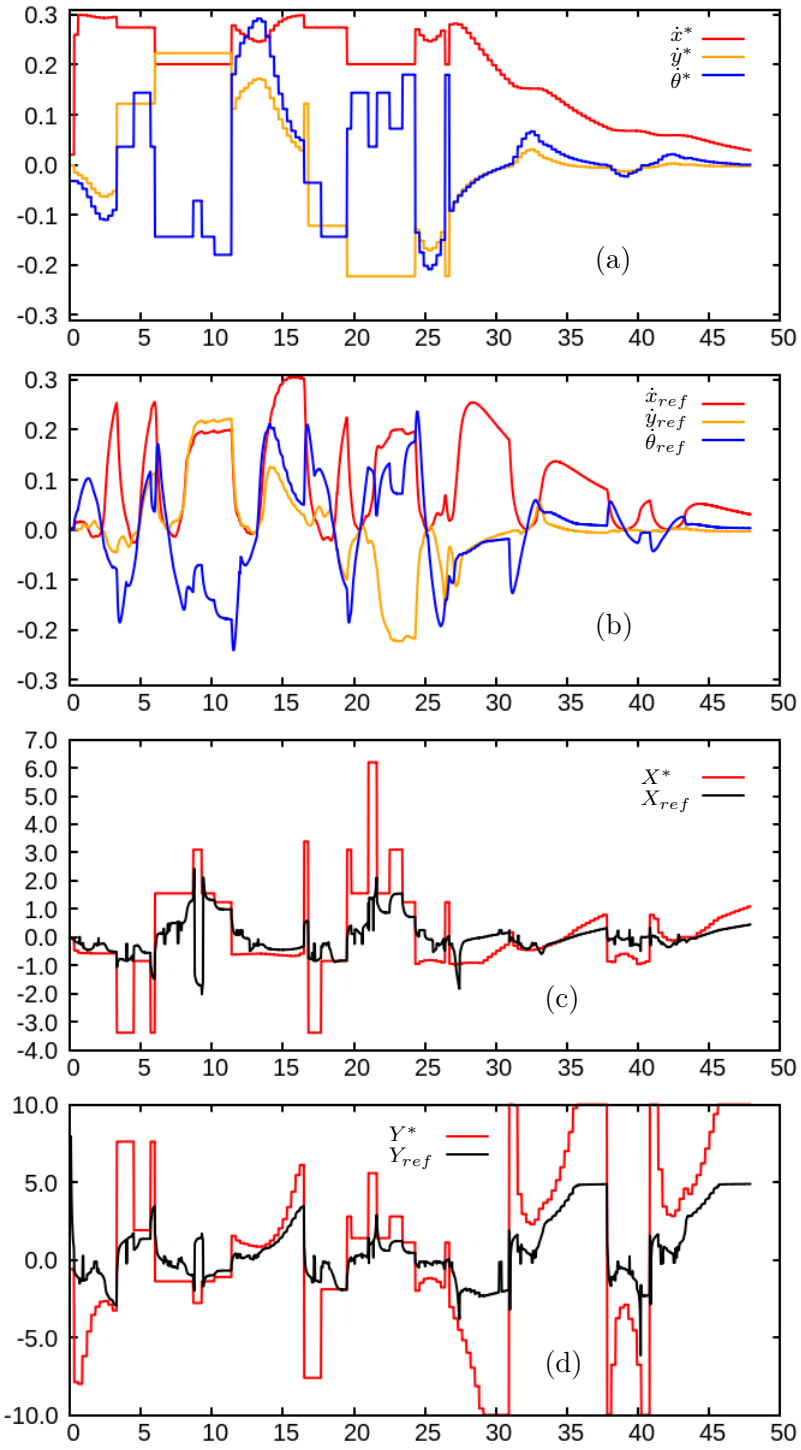


Figure 5.20: Simulation results of the practical application using the direct route (conventional) controller with (a), (b), the desired and reference (output of the controller) $3D$ robot velocity respectively (linear velocity in m/s and angular velocity in rad/s), (c), (d), the $x - axis$, and $y - axis$ components respectively of the desired and reference ICR points (in $meters$).

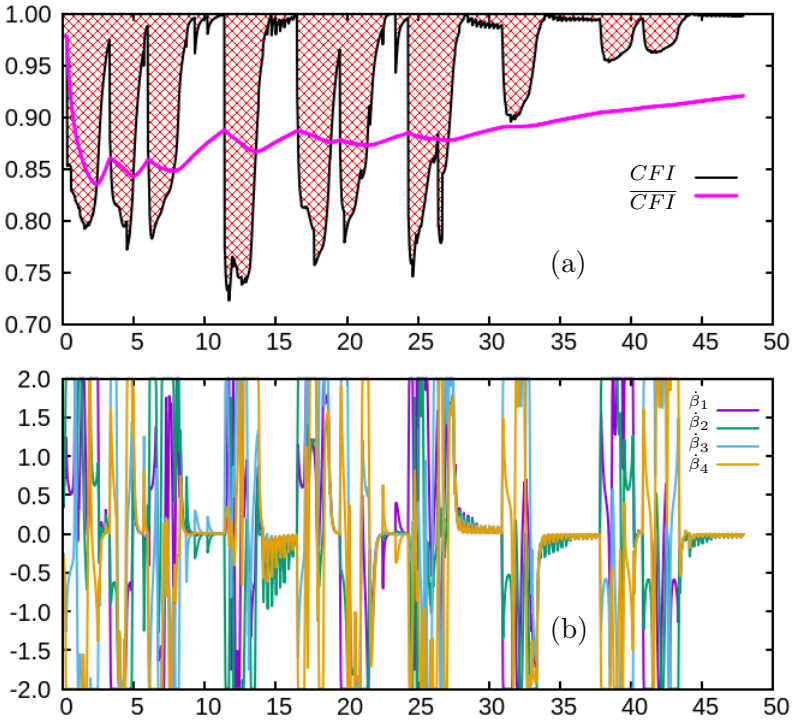


Figure 5.21: Simulation results of the practical application using the direct route (conventional) controller with (a) the command fulfillment index (*dimensionless*) and the evolution of its average value $\overline{CFI}$, (b) the corresponding steering velocity (in *rad/s*).

blue and red colors in the respective figures) is directly proportional to robot velocity error.

In Fig. 5.19b, and 5.21b, the respective steering velocity for both controllers is shown. It is evident that the proposed algorithm is less demanding in terms of steering joint motion, apart from the issue encountered during the time period $t = [2, 5[s$ where the direct route controller was activated. It is worth noting that this incident can be avoided by further tuning parameters like the penalty factor T_{bias}, and/or the dimensions of the elliptic footprint axes. However we would like to show, here, that such behavior can be expected when the desired ICR point is near the robot's geometric center for a thorough performance evaluation.

Finally, this application scenario has been performed experimentally using the Neobotix-MPO700 industrial mobile robot equipped with laser scanners. The test using the direct route controller suffered severe vibrations due to the excessive steering required (refer to Fig. 5.21b) and is considered as a failure. On the other hand, the complementary route controller test was performed successfully with satisfactory

Figure 5.22: Snap shots of the real experiment employing the complementary route ICR controller.

behavior, as shown in the video supplied [1], snapshots of which are provided in Fig. 5.22.

5.5 Conclusion

In this paper, a complementary route based ICR controller is introduced. Its performance against the conventional ICR controller is compared quantitatively using the developed evaluation metric: *"command fulfillment index"*. The two controllers are also compared in two case studies. The first features simple discontinuous velocity signals applied at low frequency to allow for a detailed investigation of both controllers behavior. The second is a practical application in which command discontinuity occurs at much higher frequency. The complementary route ICR controller shows superior results in all the tests performed.

[1] Also visible at: https://youtu.be/qsFAlTPs1yw.

Chapter 6

Conclusion and Future Perspectives

Contents

6.1 Summary of the Thesis 79

6.2 Conclusions and Future Work 81

6.1 Summary of the Thesis

In Chapter 3 we introduced a kinematic model that is robust to all kinds of singularities. The model itself is representational singularity free for 3 or more steerable wheeled robots, thanks to the use of the $3D$ Cartesian space velocity formulation. We then added a numeric treatment for such model, making it robust against kinematic singularities. In effect, it provides a damping behavior opposite to that of the singular, where the singular steer joint velocity slows down to zero at the singular configuration instead of increasing indefinitely. A formula for tuning such damping is provided so that the steering maximum acceleration is well respected. This approach can be used to control the SWMR in $3D$ Cartesian velocity space, assuming continuous velocity commands. A major merit of this development is that it provides a solution to the all singularities at the kinematic model level which is expected to greatly lower down the computational cost, making it most suitable for high sampling rate controller implementation.

In contrast to the approach adopted in Chapter 3 utilizing only a $3D$ Cartesian space velocity based kinematic model, Chapter 4 presents two decoupled controllers for the steering and driving systems. The latter utilizing the model presented in Chapter 3 for controlling the driving speed, while the former being an ICR based steering controller, the objective of which is to handle discontinuity in the robot velocity commands. Such discontinuity is highly expected in applications where both

the human and the robot share the same workspace for safety reasons. We believe that ICR based controllers are best suited for steering coordination since the steer joint coordinates are computed from a unique ICR point (output of the controller), rather than from the robot $3D$ velocity vector which would result in accumulating steering position errors. The steering controller developed, employs a QP optimization algorithm taking into account the maximum steering velocity/acceleration as linear constraints and outputs the best feasible *"next sample time"* ICR point on the move from a current to a desired one. From which point, the *"next sample time"* steering joint positions are computed, differentiated to obtain the steering velocity commands to be sent to the actuator drives. The $3D$ Cartesian velocity space kinematic controller then utilizes the output steering velocity command to compute a compatible wheel velocity commands (since the wheel type employed is off centered steerable). The framework has been successfully validated in simulation as well as in real world experiments featuring discontinuous velocity commands as expected by the target application.

Chapter 5 tackles a common problem in ICR based controllers, specifically in heading control applications, where a constant robot orientation is desired (as in visual servoing tasks). In such case, the output of an ICR controller will keep changing the sign of the angular velocity in a continuous corrective action to maintain the desired heading angle. This requires moving the ICR point from one extreme of the ICR space to the other, leading to large ICR movement which is far from being time efficient. On the other hand, such ICR motion results usually in the ICR point passing by the robot geometric center, at which ICR location, the feasible robot velocity is a rotation in place, which results in blocking the robot motion. This in practice results in failure to accomplish the task in hand. Chapter 5 introduced a new approach to ICR control, where moving from a current ICR point to a desired one is done in a complementary fashion rather than the direct one, by moving across artificial ICR space borders. We used QP optimization to find the shortest complementary ICR route, employing square shaped ICR space. Then the obtained complementary route is compared with the direct one in terms of expected time requirements, the shortest of which is then chosen and followed. This method results in much faster convergence time and is shown successful in both simulations and in real experiments.

6.2 Conclusions and Future Work

The complementary ICR controller shows promising results in terms of convergence time (from an initial ICR point to a desired one), and in overall steering motion. We believe that such complementary route can be further enhanced to avoid issues perturbing the robot motion, and reported in Chapter 5. Also the way we handled kinematic singularity in the ICR space, presented in Chapter 4, can be either enhanced or replaced to avoid the chattering effect (reported as well) in steering acceleration/velocity, and to be less restrictive on the accessible ICR space. Another interesting idea is to explore the effect of utilizing a circular shaped ICR borders instead of the square shaped one, which might make implementation more easier/straight forward.

Handling steer joint position limit is not handled in our work due to lack of motivation since the SWMR hardware available at our lab (Neobotix-MPO700) has no such limitation, where the steering joints can run continuously with no trouble. Despite this fact, we believe our framework can be extended to take into account joint position limits seamlessly thanks to the complementary route development. Using such controller, the ICR point can move from one quadrant of the ICR space to the complementary one (for example, from the 1^{st} to the 3^{rd} or from the 2^{nd} to the 4^{th} or vice-versa) with minimal steering motion (less than or equal 90^o in rotation angle). It can be shown that a SWMR with joint steering position limit ranging from 0^o to 360^o has full accessibility to the ICR space with no need for any further development employing the complementary route algorithm presented in Chapter 5.

The steerable wheeled mobile robots require precise coordination of the steering axes as well as driving rates for those employing off centered wheels. This requires high performance motion drives for the steering/driving actuators, where the error in following velocity (or position) commands has to be very small, this in fact lead us to a conclusion that a dynamic model is required for such systems, not only the kinematic one. As a future work, the dynamic model of SWMR shall be developed and the controller shall be implemented on the robot "on-board" computer to achieve higher sampling rate (the current sample rate used in experiments is 40 Hz).

In terms of mobile robot structure, we should hold a comparison between the performance of SWMR and that of conventional wheeled FOMR as those employing Castor wheels or differential drive with an offset turret. The latter being a very promising candidate for providing omni-directional capability making full use of the low cost, high load capacity and smooth (vibration free) operation of conventional wheels. Differential drive system with offset turret has also an extra advantage in

terms of hardware cost, since it requires no redundancy in actuation in contrast to SWMR and fully motorized Castor wheeled mobile robots. It has 3 degrees of mobility instantaneously accessible powered by 3 actuators; two for the differential drive wheels and one for the turret rotation.

Appendix A

Singularity Treatment and Acceleration Limits

In this appendix it is proven that the damping element δ_2 computed using (3.19) guarantees respecting the maximum acceleration limit of the steering joint, when used in (3.13) serving as numeric treatment for kinematic singularity.

A.1 Case I: $\delta_{2c} \leq 0$

From (3.17), this case corresponds to:

$$\sqrt{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}}} \leq (\tilde{x}_i^2 + \tilde{y}_i^2),$$

and we can propose:

$$\tilde{x}_i^2 + \tilde{y}_i^2 = \sqrt{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}}} + c, \tag{A.1}$$

where $c \in \mathbb{R}^+$ is a positive semi-definite constant. Substituting by (A.1) in (3.15), where $\delta_2 = \delta_1$ from (3.19) we obtain:

$$\ddot{\beta}_i = \frac{k_i(\dot{\xi}, \ddot{\xi})}{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}} + 2c_1 * \sqrt{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}}} + c_1^2}, \tag{A.2}$$

where $c_1 = c + \delta_1$ is also positive semi-definite. Rearranging (A.2) we obtain:

$$\frac{\ddot{\beta}_i}{\ddot{\beta}_{i(max)}} = \frac{k_i(\dot{\xi}, \ddot{\xi})}{|k_i(\dot{\xi}, \ddot{\xi})| + 2c_1 * \sqrt{|k_i(\dot{\xi}, \ddot{\xi})| \ddot{\beta}_{i(max)}} + c_1^2 \ddot{\beta}_{i(max)}}. \tag{A.3}$$

From (A.3), it is evident that the right hand side is a unit fraction, and consequently the steer joint acceleration will comply with the maximum limit.

A.2 Case II: $\delta_{2c} > 0$

In this case, we assume that the value of δ_2 is constant and as such the derivative in (3.15) still holds. This is a reasonable assumption after few movements of the robot passing by kinematic singularity, where it quickly converges to the maximum value of δ_{2c}. In such case, (3.17) corresponds to:

$$\sqrt{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}}} > (\tilde{x}_i^2 + \tilde{y}_i^2),$$

again we can propose:

$$\tilde{x}_i^2 + \tilde{y}_i^2 = \sqrt{\frac{|k_i(\dot{\xi}, \ddot{\xi})|}{\ddot{\beta}_{i(max)}}} - c. \tag{A.4}$$

Substituting by (A.4) in (3.15), we obtain the same formulae in (A.2) and (A.3) with $c_1 = \max(\delta_{2c}) - c$. However the definition of the $\max(*)$ function guarantees that $\max(\delta_{2c}) > c$, so $c_1 > 0$ and consequently $\ddot{\beta}_i > \ddot{\beta}_{i(max)}$. Hence we conclude that using (3.19) along with (3.13), the kinematic singular behavior is well damped while respecting the steer joint acceleration limits.

Bibliography

[AS93] J. Ackermann and W. Sienel. Robust yaw damping of cars with front and rear wheel steering. *IEEE Transactions on Control Systems Technology*, 1(1):15–20, Mar 1993.

[AS03] Tokihiko Akita and Katsuhiko Satoh. Development of 4ws control algorithms for an suv. *JSAE Review*, 24(4):441 – 448, 2003.

[BC96] A. Betourne and G. Campion. Kinematic modelling of a class of omnidirectional mobile robots. In *IEEE Int. Conf. on Robotics and Automation*, pages 3631–3636, 1996.

[BJ04] Thomas Bak and Hans Jakobsen. Agricultural robotic platform with four wheel steering for weed detection. *Biosystems Engineering*, 87(2):125 – 136, 2004.

[BO08] Julie Behan and Derek T. O'Keeffe. The development of an autonomous service robot. implementation: "lucas"—the library assistant robot. *Intelligent Service Robotics*, 1(1):73–89, Jan 2008.

[BOW+07] C. Borst, C. Ott, T. Wimbock, B. Brunner, F. Zacharias, B. Bauml, U. Hillenbrand, S. Haddadin, A. Albu-Schaffer, and G. Hirzinger. A humanoid upper body system for two-handed manipulation. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2766–2767, April 2007.

[BS03] Kyung-Seok Byun and Jae-Bok Song. Design and construction of continuous alternate wheels for an omnidirectional mobile robot. *Journal of Robotic Systems*, 20(9):569–579, 2003.

[CBC+10] 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*, pages 4305–4310, 2010.

[CBN96] G. Campion, G. Bastin, and B. Novel. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Trans. on Robotics and Automation*, 12(1):47–62, 1996.

[CCT+13] Wei-Hsi Chen, Ching-Pei Chen, Jia-Shiuan Tsai, Jackie Yang, and Pei-Chun Lin. Design and implementation of a ball-driven omnidirectional spherical robot. *Mechanism and Machine Theory*, 68:35 – 48, 2013.

[CHB+10] C. P. Connette, S. Hofmeister, A. Bubeck, M. Haegele, and A. Verl. Model-predictive undercarriage control for a pseudo-omnidirectional, wheeled mobile robot. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, pages 1–6, June 2010.

[CHV12] C. Connette, M. Hägele, 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*, pages 4959–4965, 2012.

[CLM10] L. Clavien, M. Lauria, and F. Michaud. Instantaneous centre of rotation estimation of an omnidirectional mobile robot. In *2010 IEEE International Conference on Robotics and Automation*, pages 5435–5440, May 2010.

[CLTB09] Christophe Cariou, Roland Lenain, Benoit Thuilot, and Michel Berducat. Automatic guidance of a four-wheel-steering mobile robot for accurate field operations. *Journal of Field Robotics*, 26(6-7):504–518, 2009.

[CPHV08] C. P. Connette, A. Pott, M. Hagele, and A. Verl. Control of an pseudo-omnidirectional, non-holonomic, mobile robot based on an icm representation in spherical coordinates. In *2008 47th IEEE Conference on Decision and Control*, pages 4976–4983, Dec 2008.

[CPHV09] C. P. Connette, C. Parlitz, M. Hagele, and A. Verl. Singularity avoidance for over-actuated, pseudo-omnidirectional, wheeled mobile robots. In *IEEE Int. Conf. on Robotics and Automation*, pages 4124–4130, 2009.

[CPHV10] C. P. Connette, A. Pott, M. Hägele, and A. Verl. Addressing input saturation and kinematic constraints of overactuated undercarriages by predictive potential fields. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4775–4781, Oct 2010.

[CY17] J. T. Costa and M. Yim. Designing for uniform mobility using holonomicity. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2448–2453, May 2017.

[DBB+02] Olaf Diegel, Aparna Badve, Glen Bright, Johan Potgieter, and Sylvester Tlale. Improved mecanum wheel design for omni-directional robots. In *Proc. 2002 Australasian Conference on Robotics and Automation, Auckland*, pages 117–121, 2002.

[DWASH11] A. Dietrich, T. Wimböck, A. Albu-Schäffer, and G. Hirzinger. Singularity avoidance for nonholonomic, omnidirectional wheeled mobile platforms with variable footprint. In *IEEE Int. Conf. on Robotics and Automation*, pages 6136–6142, 2011.

[ESWBB07] A. El-Shenawy, A. Wellenreuther, A. S. Baumgart, and E. Badreddin. Comparing different holonomic mobile robots. In *2007 IEEE International Conference on Systems, Man and Cybernetics*, pages 1584–1589, Oct 2007.

[FBG+09] M. Fuchs, C. Borst, P. Robuffo Giordano, A. Baumann, E. Kraemer, J. Langwald, R. Gruber, N. Seitz, G. Plank, K. Kunze, R. Burger, F. Schmidt, T. Wimboeck, and G. Hirzinger. Rollin' justin - design considerations and realization of a mobile platform for a humanoid upper body. In *2009 IEEE International Conference on Robotics and Automation*, May 2009.

[FCF+10] F. Ferland, L. Clavien, J. Frémy, D. Létourneau, F. Michaud, and M. Lauria. Teleoperation of azimut-3, an omnidirectional non-holonomic platform with steerable wheels. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2515–2516, Oct 2010.

[FRC96] L. Ferriere, B. Raucent, and G. Campion. Design of omnimobile robot wheels. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 4, pages 3664–3670 vol.4, Apr 1996.

[Gas] Luca Di Gaspero. Quadprog++. Accessed: 23-04-2017.

[GFASH09] 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*, pages 4080–4087, 2009.

[Gfr08] A. Gfrerrer. Geometry and kinematics of the mecanum wheel. *Computer Aided Geometric Design*, 25(9):784 – 791, 2008. Classical Techniques for Applied Geometry.

[GH14] C. Gruber and M. Hofbaur. Practically stabilizing motion control of mobile robots with steering wheels. In *2014 IEEE Conference on Control Applications (CCA)*, pages 1312–1317, Oct 2014.

[GHS04] Birgit Graf, Matthias Hans, and Rolf D. Schraft. Care-o-bot ii—development of a next generation robotic home assistant. *Autonomous Robots*, 16(2):193–205, Mar 2004.

[HBMK09] M. Hvilshoj, S. Bogh, O. Madsen, and M. Kristiansen. The mobile robot "little helper": Concepts, ideas and working principles. In *2009 IEEE Conference on Emerging Technologies Factory Automation*, pages 1–4, Sept 2009.

[HK00] Robert Holmberg and Oussama Khatib. Development and control of a holonomic mobile robot for mobile manipulation tasks. *The International Journal of Robotics Research*, 19(11):1066–1074, 2000.

[HKL10] K. L. Han, H. Kim, and J. S. Lee. The sources of position errors of omni-directional mobile robot with mecanum wheel. In *2010 IEEE International Conference on Systems, Man and Cybernetics*, pages 581–586, Oct 2010.

[HNB+11] B. Horan, Z. Najdovski, T. Black, S. Nahavandi, and P. Crothers. Oztug mobile robot for manufacturing transportation. In *2011 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3554–3560, Oct 2011.

[HS03] J. Hermosillo and S. Sekhavat. Feedback control of a bi-steerable car using flatness application to trajectory tracking. In *Proceedings of the*

2003 American Control Conference, 2003., volume 4, pages 3567–3572 vol.4, June 2003.

[HTKR+15] D. Holz, A. Topalidou-Kyniazopoulou, F. Rovida, M. R. Pedersen, V. Krüger, and S. Behnke. 2015 ieee 20th conference on emerging technologies factory automation (etfa). pages 1–9, Sept 2015.

[Ilo75] B.E. Ilon. Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base, April 8 1975. US Patent 3,876,255.

[KCAP17] A. Khelloufi, A. Cherubini, N. Achour, and R. Passama. Tentacle-based moving obstacle avoidance for omnidirectional robots with visibility constraints. In *2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2017.

[LL00] Choon-Young Lee and Ju-Jang Lee. Walking-support robot system for walking rehabilitation: design and control. *Artificial Life and Robotics*, 4(4):206–211, Dec 2000.

[LL05] K.H. Low and Y.P. Leow. Kinematic modeling, mobility analysis and design of wheeled mobile robots. *Ad. Robotics*, 19(1):73–99, 2005.

[LMD+17] Z. Li, P. Moran, Q. Dong, R. J. Shaw, and K. Hauser. Development of a tele-nursing mobile manipulator for remote care-giving in quarantine areas. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3581–3586, May 2017.

[LNL+06a] M. Lauria, I. Nadeau, P. Lepage, Y. Morin, P. Giguere, F. Gagnon, D. Letourneau, and F. Michaud. Design and control of a four steered wheeled mobile robot. In *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, pages 4020–4025, Nov 2006.

[LNL+06b] M. Lauria, I. Nadeau, P. Lepage, Y. Morin, P. Giguere, F. Gagnon, D. Letourneau, and F. Michaud. Kinematical analysis of a four steered wheeled mobile robot. In *2006 IEEE International Symposium on Industrial Electronics*, volume 4, pages 3090–3095, July 2006.

[LQXX09] T. L. Lam, H. Qian, Y. Xu, and G. Xu. Omni-directional steer-by-wire interface for four wheel independent steering vehicle. In *2009 IEEE*

International Conference on Robotics and Automation, pages 1383–1388, May 2009.

[LTCM06] Roland Lenain, Benoit Thuilot, Christophe Cariou, and Philippe Martinet. High accuracy path tracking for vehicles in presence of sliding: Application to farm vehicle automatic guidance for agricultural tasks. *Autonomous Robots*, 21(1):79–97, Aug 2006.

[MC02] François Michaud and Serge Caron. Roball, the rolling robot. *Autonomous Robots*, 12(2):211–222, Mar 2002.

[MF00] K. L. Moore and N. S. Flann. A six-wheeled omnidirectional autonomous mobile robot. *IEEE Control Systems*, 20(6):53–66, Dec 2000.

[MLA+03] F. Michaud, D. Letourneau, M. Arsenault, Y. Bergeron, R. Cadrin, F. Gagnon, M. A. Legault, M. Millette, J. F. Pare, M. C. Tremblay, P. Lapage, Y. Morin, J. Bisson, and S. Caron. Azimut, a leg-track-wheel robot. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 3, pages 2553–2558 vol.3, Oct 2003.

[MLA+05] François Michaud, Dominic Létourneau, Martin Arsenault, Yann Bergeron, Richard Cadrin, Frédéric Gagnon, Marc-Antoine Legault, Mathieu Millette, Jean-François Paré, Marie-Christine Tremblay, Pierre Lepage, Yan Morin, Jonathan Bisson, and Serge Caron. Multi-modal locomotion robotic platform using leg-track-wheel articulations. *Autonomous Robots*, 18(2):137–156, Mar 2005.

[MN87a] P. Muir and C. Neuman. Kinematic modeling for feedback control of an omnidirectional wheeled mobile robot. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, volume 4, pages 1772–1778, Mar 1987.

[MN87b] Patrick F. Muir and Charles P. Neuman. Kinematic modeling of wheeled mobile robots. *Journal of Robotic Systems*, 4(2):281–340, 1987.

[MT90] N. Matsumoto and M. Tomizuka. Vehicle lateral velocity and yaw rate control with two independent control inputs. In *1990 American Control Conference*, pages 1868–1875, May 1990.

[NH86] Y. Nakamura and H. Hanafusa. Inverse kinematic solutions with singularity robustness for robot manipulator control. *ASME Journal of Dyn. Sys., Measur., and Control*, 108(3):163–171, 1986.

[NTLM16] Ange Nizard, Benoit Thuilot, Roland Lenain, and Youcef Mezouar. Nonlinear path tracking controller for bi-steerable vehicles in cluttered environments**this work was supported by the european commission (feder) and the labex im0bs3 through fui-aap14 vipafleet. *IFAC-PapersOnLine*, 49(15):19 – 24, 2016. 9th IFAC Symposium on Intelligent Autonomous Vehicles IAV 2016.

[NTST00] K. Nagatani, S. Tachibana, M. Sofne, and Y. Tanaka. Improvement of odometry for omnidirectional vehicle using optical flow information. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 468–473, 2000.

[OAGM13] R. Oftadeh, M. M. Aref, R. Ghabcheloo, and J. Mattila. Bounded-velocity motion control of four wheel steered mobile robots. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 255–260, July 2013.

[OGM13] 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*, pages 4845–4851, 2013.

[OGM14] 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)*, pages 2925–2931, 2014.

[OGM15] R. Oftadeh, R. Ghabcheloo, and J. Mattila. A time-optimal bounded velocity path-following controller for generic wheeled mobile robots. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 676–683, May 2015.

[PK94] F. G. Pin and S. M. Killough. A new family of omnidirectional and holonomic wheeled platforms for mobile robots. *IEEE Transactions on Robotics and Automation*, 10(4):480–489, Aug 1994.

[PTC+15] D. Portugal, P. Trindade, E. Christodoulou, G. Samaras, and J. Dias. On the development of a service robot for social interaction with the elderly. In *IET International Conference on Technologies for Active and Assisted Living (TechAAL)*, pages 1–6, Nov 2015.

[RCF+09] U. Reiser, C. Connette, J. Fischer, J. Kubacki, A. Bubeck, F. Weisshardt, T. Jacobs, C. Parlitz, M. Hägele, and A. Verl. Care-o-bot 3 - creating a product vision for service robot applications by integrating design and technology. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1992–1998, Oct 2009.

[RO18] Pierluigi Rea and Erika Ottaviano. Design and development of an inspection robotic system for indoor applications. *Robotics and Computer-Integrated Manufacturing*, 49:143 – 151, 2018.

[RSK10] A. Ramirez-Serrano and R. Kuzyk. Modified mecanum wheels for traversing rough terrains. In *2010 Sixth International Conference on Autonomic and Autonomous Systems*, pages 97–103, March 2010.

[RU93] D. B. Reister and M. A. Unseren. Position and constraint force control of a vehicle with two or more steerable drive wheels. *IEEE Transactions on Robotics and Automation*, 9(6):723–731, Dec 1993.

[SCFP17] M. Sorour, A. Cherubini, P. Fraisse, and R. Passama. Motion discontinuity-robust controller for steerable mobile robots. *IEEE Robotics and Automation Letters*, 2(2):452–459, April 2017.

[SCPF16] M. Sorour, A. Cherubini, R. Passama, and P. Fraisse. Kinematic modeling and singularity treatment of steerable wheeled mobile robots with joint acceleration limits. In *IEEE Int. Conf. on Robotics and Automation*, pages 2110–2115, 2016.

[SCSB06] P. F. Santana, C. Candido, V. Santos, and J. Barata. A motion controller for compliant four-wheel-steering robots. In *2006 IEEE International Conference on Robotics and Biomimetics*, pages 532–537, Dec 2006.

[SL09] K. H. Seo and J. J. Lee. The development of two mobile gait rehabilitation systems. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(2):156–166, April 2009.

[SMG15] C. Stoger, A. Muller, and H. Gattringer. Kinematic analysis and singularity robust path control of a non-holonomic mobile platform with several steerable driving wheels. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4140–4145, 2015.

[SN11] M. F. Selekwa and J. R. Nistler. Path tracking control of four wheel independently steered ground robotic vehicles. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 6355–6360, Dec 2011.

[SPS12] 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*, pages 5360–5366, 2012.

[TNFM12] Filip Tomik, Shahin Nudehi, Louis L. Flynn, and Ranjan Mukherjee. Design, fabrication and control of spherobot: A spherical mobile robot. *Journal of Intelligent & Robotic Systems*, 67(2):117–131, Jul 2012.

[TNM96] B. Thuilot, B. Novel, and A. Micaelli. Modeling and feedback control of mobile robots equipped with several steering wheels. *IEEE Trans. on Robotics and Automation*, 12(3):375–390, 1996.

[US16] T. Ushijima and J. Satake. Development of a breathing detection robot for a monitoring system. In *2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS)*, pages 790–795, Aug 2016.

[WA95] M. West and H. Asada. Design and control of ball wheel omnidirectional vehicles. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, volume 2, pages 1931–1938 vol.2, May 1995.

[Wad06] M. Wada. Virtual link model for redundantly actuated holonomic omnidirectional mobile robots. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 3201–3207, May 2006.

[Wam86] C. W. Wampler. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. *IEEE Trans. on Sys., Man, and Cybernetics*, 16(1):93–101, 1986.

[WBE+90] C. R. Weisbin, B. L. Burks, J. R. Einstein, R. R. Feezell, W. W. Manges, and D. H. Thompson. Hermies-iii: A step toward autonomous mobility, manipulation and perception. *Robotica*, 8(1):7–12, 1990.

[WCGR02] R. L. Williams, B. E. Carter, P. Gallina, and G. Rosati. Dynamic model with slip for wheeled omnidirectional robots. *IEEE Trans. on Robotics and Automation*, 18(3):285–293, Jun 2002.

[YBB+09] Junyou Yang, Dianchun Bai, Shan Bai, Yuqing Li, and Shuoyu Wang. Design of mechanical structure and tracking control system for lower limbs rehabilitative training robot. In *2009 International Conference on Mechatronics and Automation*, pages 1824–1829, Aug 2009.

[Yos85] Tsuneo Yoshikawa. Manipulability of robotic mechanisms. *The International Journal of Robotics Research*, 4(2):3–9, 1985.