Proceedings of the XII International Symposium on Dynamic Problems of Mechanics (DINAME 2007),
P.S.Varoto and M.A.Trindade (editors), ABCM, Ilhabela, SP, Brazil, February 26 - March 2, 2007

A geometric approach to consideration of control variable limits for wheeled mobile robots with kinematic constraints

Holger Blume and Bodo Heimann

Hannover Center of Mechatronics, University of Hannover, Appelstr. 11, 30167 Hannover, Germany, {blume, heimann}@mzh.uni-hannover.de

Abstract: Subject of this paper is a method for consideration of control variable limits of single actuators in a redundant driven kinematic structure of a wheeled mobile robot (wmr). It is assumed that the desired robot state is given by the translational and rotational velocity of a coordinate system fixed to the robot frame. With the assumption of rigid body elements, motion is possible only when the kinematic constraints due to redundancy in actuation are met. According to the kinematic design the wmr reference point motion results in different set value characteristics for the steering actuators. This way limited actuation dynamics of single actuators influence the reachable state region for the complete structure. In case of a wmr moving in the horizontal plane the instantaneous center of rotation of all wheels must coincide, which limits the change of wmr state to a region determined by the dynamics of the steering actuators. By determination of the non feasible states of the single actuators a combination of all non feasible states by inversion results in the valid set value region. This method can be used in command generation for dynamically demanding feasible trajectories, as well as to limit set values generated by higher level control loops to feasible ones. In case the limits for valid commands are violated an alternative command can be generated, which in this example is closest to the original command. Furthermore this procedure provides a solution for singular operating points, when the instantaneous center of rotation approaches one of the steered wheels as the dynamic constraints of this wheel are considered in the final command. The paper describes the determination of the valid region for actuating commands in detail. Furthermore the generation of alternative feasible commands is presented. The efficient implementation to a µ-controller is described. Experimental results with an 8-DOF wheeled mobile robot prove the effectiveness of this approach.

Keywords: wmr, kinematic constraints, instantaneous center of rotation, control variable limit

INTRODUCTION

In wheeled mobile robots moving in the horizontal plane various locomotion kinematics have been developed each having assets and drawbacks (Campion et al. (1996)). In order to fulfil specific tasks with these robots the higher order control loop must be able to execute drive commands. Because the specialities of the kinematic structure of the robot is not necessarily used for fulfilment of the task, the inter operability of algorithms and the application of algorithms on different platforms demand a well defined interface between higher order control loops and the locomotion platform. In this area the unicycle kinematic as interface has been proven quite powerful, which is the reason it is used in a variety of robotic projects (Oriolo et al. (2002); Luca et al. (2000); Horn (1997)). The unicycle kinematic reduces the robot model to a single wheel in the center of the robot for which drive commands in form of velocities in x- and y-direction as well as the angular velocity around the vertical axis are given.

When leaving the specialities of the kinematics behind there is no common way to account for limitations of the locomotion system, which are due to the kinematic structure except strong limitations for v_x, v_y and ω, reducing the dynamic potential of the robot and the dynamics in robot control. When dealing with redundantly driven kinematic structures kinematic constraints are to be met. Assuming rigid bodies the robot can operate only with a consistent set of actuation variables. This means that each actuator must be controlled precisely in order to enable a coordinated movement of the robot. In dynamically demanding trajectories single actuators may approach saturation points. The constraints are violated and coordinated motion of the robot becomes impossible.

In the following a geometric based algorithm is presented, which offers the possibility to check the drive command for the next time step based on the desired drive command and the current state of the robot. It accounts for the control variable limits and suggests an alternative feasible drive command in case any constraints are violated.

The algorithm is based on the principle of the instantaneous center of rotation, which demands that the instantaneous center of rotation of all wheels must coincide in order achieve coordinated movement of the robot (Bhatt et al. (2004)). The algorithm inherently solves the problem of singular points when the instantaneous center of rotation approaches one of the steered wheels, as the dynamic constraints of this wheel are considered in the alternative command.

Although an analytical solution to the problem is chosen the algorithm is computational efficient. Experimental results show the effectiveness of the algorithm applied to an 8-DOF redundantly driven mobile platform (Abdellatif et al. (2002, 2003)). In section **KINEMATICS** some basic kinematic equations are shown to introduce nomenclature to the reader. Section **CONTROL VARIABLE LIMITS** introduces the actuator saturation limits and presents how to

check for violation of these limits. In the following section **COMMAND GENERATION UNDER CONSTRAINTS** it is described how to substitute a desired drive command by a feasible alternative one. Section **EXPERIMENTAL RESULTS** describes the setup of the wheeled mobile robot for showing the effectiveness of the algorithm and some experimental results. Finally the **CONCLUSIONS** section summarizes the results of this work.

KINEMATICS

The interface between higher order control loops and the robotic platform is derived from the unicycle kinematics. However, the drive system of the mobile robot consists of a number of driven and steered wheels. The exemplary kinematic structure, with four driven and steered wheels is depicted in Fig. 1. The algorithms presented here are also valid for non symmetrical kinematic structures. As the preferred direction of travel is heading towards the y-axis, angles are calculated mathematically positive starting at the y-axis. According to the rigid body assumption, movement is only possible if kinematic constraints are met. For a mobile platform with nonzero dimensions the instantaneous center of rotation constitutes the constraints formulation. Given the instantaneous center of rotation $\vec{m}$ and the rotational velocity ω

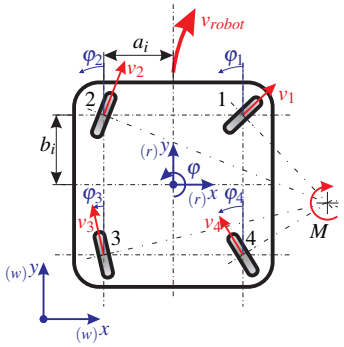


Figure 1 – Robot kinematics.

the velocity $\vec{v}$ of point $\vec{p}$ in a $\mathbb{R}^2$ environment is derived from

$$\begin{aligned} \vec{v} &= \begin{pmatrix} 0 & -\omega \\ \omega & 0 \end{pmatrix} \cdot \begin{pmatrix} r_x \\ r_y \end{pmatrix} \quad \text{or} \\ &= \quad \mathbf{\Omega} \quad \cdot \quad \vec{r} \quad , \end{aligned} \tag{1}$$

with $\vec{r}$ being the vector from $\vec{m}$ to $\vec{p}$. Given a desired robot state consisting of $\vec{v}$ and ω, the instantaneous center of rotation is calculated by

$$\vec{r} = \vec{p} - \vec{m} = \quad \mathbf{\Omega}^{-1} \cdot \vec{v} \quad \wedge \quad \omega \neq 0 \quad \text{or} \tag{2}$$

$$\vec{m} = \ldots\ldots = \vec{p} - \mathbf{\Omega}^{-1} \cdot \vec{v} \quad \wedge \quad \omega \neq 0 \quad , \tag{3}$$

with

$$\mathbf{\Omega}^{-1} = \begin{pmatrix} 0 & \frac{1}{\omega} \\ \frac{-1}{\omega} & 0 \end{pmatrix} \quad \wedge \quad \omega \neq 0 \quad . \tag{4}$$

To calculate the wheel angles φ_i and velocities $\vec{v}_{wh,i}$ from the desired robot state the location of the wheel $\vec{w}_i$ within the robot frame must be known:

$$\vec{v}_{wh,i} = \mathbf{\Omega} \cdot (\vec{w}_i - \vec{m}) \tag{5}$$

$$\tan \varphi_i = \frac{-v_{x,i}}{v_{y,i}} \tag{6}$$

$$\omega_{wh,i} = \frac{|\vec{v}_{wh,i}|}{r_{wh,i}} \tag{7}$$

As the wheel is driven in both directions the calculation of the velocity is not distinct,

$$\vec{v}_{wh,i}(\varphi_i, \omega_{wh,i}) = \vec{v}_{wh,i}(\varphi_i \pm \pi, -\omega_{wh,i}) \quad , \tag{8}$$

which is advantageous because in certain driving maneuvers reducing the steering angle necessary by $\frac{\pi}{2}$. To avoid ambiguities the direction of the wheel velocity is used instead of the real steering angle. It is essential that the wheel steering angle φ_i has low control error, as otherwise coordinated movement of the robotic platform is impossible.

CONTROL VARIABLE LIMITS

As stated before the coordination of the wheel steering angle is essential for a consistent state of the robot. The dynamics of the steering angle is limited which leads to a problem in dynamically demanding trajectories: If one of the wheels fails in adjusting the steering angle, the kinematic constraints are violated and coordinated movement of the robot becomes impossible. Thus in the following it is shown, how to calculate the actuation limits for each single wheel and what impact these limits have on the maneuverability of the whole structure.

The calculation of the actuation limits is based on a second order model with limited rotational acceleration and limited rotational speed. First a prediction of the steering angle for the next time step is calculated. The time derivation is approximated by the first difference quotient:

$$\varphi_{pred,i} = \varphi_i(k-1) + \Delta\varphi_{soll,i}(k-1,k) \tag{9}$$

$$\dot{\varphi}_{pred,i} = (\varphi_{pred,i} - \varphi_i(k-1)) \cdot \frac{1}{T} \tag{10}$$

The limits around the current wheel state are calculated based on maximum available acceleration in both direction while accounting for the limits in rotational speed:

$$\Delta\varphi_{min,max,i} = (\dot{\varphi}_{pred,i} \pm \ddot{\varphi}^*_{i\,max} \cdot T) \cdot T \qquad \text{with} \tag{11}$$

$$\ddot{\varphi}^*_{max,i} = \begin{cases} \ddot{\varphi}_{max,i}, & \text{for } (|\dot{\varphi}_{pred,i}| + \ddot{\varphi}_{max,i} \cdot T) < \dot{\varphi}_{max,i} \\ (\dot{\varphi}_{max,i} - |\dot{\varphi}_{pred,i}|) \cdot \frac{1}{T} & \text{otherwise} \end{cases} \tag{12}$$

The actual steering angle limit for the next time step is calculated as a sum of the predicted state $\varphi_{pred,i}$ and the valid area around it $\Delta\varphi_{min,max,i}$. The area of valid steering angles for one wheel is depicted in Fig. 2(a) and calculated according to

$$\varphi_{min,max,i} = \varphi_{pred,i} + \Delta\varphi_{min,max,i} \quad . \tag{13}$$

The beam connecting from the center of the wheel towards the instantaneous center of rotation is perpendicular to the

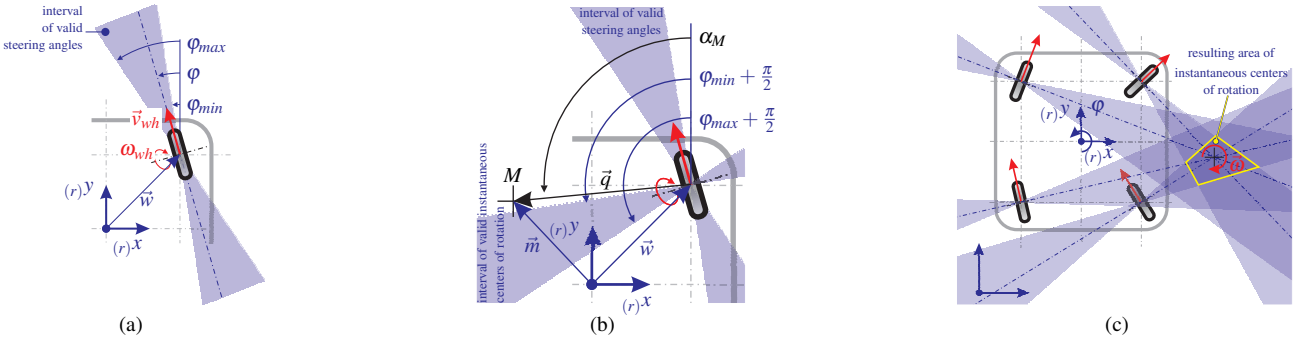


Figure 2 – Control variable limits: a) Valid steering angles for the next time step. b) Valid locations for the instantaneous center of rotation in the next time step. c) Combination of the actuation limits for each wheel.

steering angle. This way, a valid area for the instantaneous center of rotation can be calculated from the actuation limits of the wheel as depicted in Fig. 2(b). The angle of the beam towards the instantaneous center of rotation is calculated from Eq. (14)

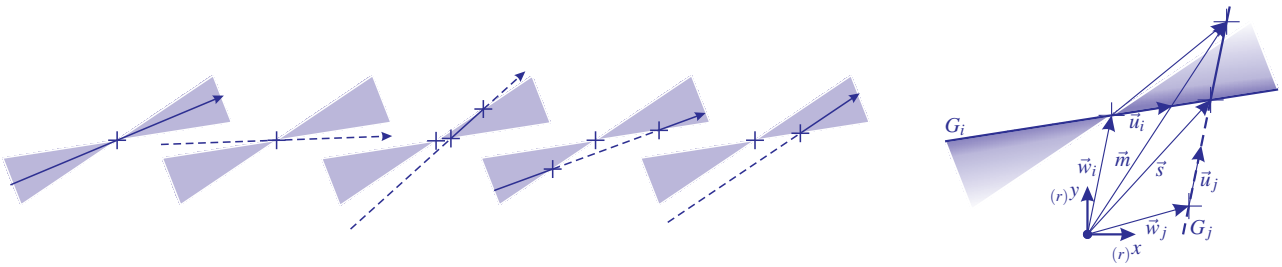
$$\alpha_{M,i} = -\arctan\left(\frac{m_x - w_{x,i}}{m_y - w_{y,i}}\right) \quad . \tag{14}$$

With this a condition for the area of valid steering angles for the next time step is given according to

$$\left.\begin{aligned} \varphi_{min,i} \leq \alpha_{M,i} - \frac{\pi}{2} \leq \varphi_{max,i} \\ v_{wh,min,i} \leq \quad v_i \quad \leq v_{wh,max,i} \end{aligned}\right\} \quad \text{for} \quad i \in 1,2,3,4 \quad . \tag{15}$$

Adding up the condition for each wheel leads to the resulting area of valid positions for the instantaneous center of rotation as depicted in Fig. 2(c). The condition can be tested independently for each wheel, by comparison of $\alpha_{M,i}$, which results from the desired robot state, with $\varphi_{min,max,i}$, resulting from the current robot state, according to Eq. (15). In order to consider the above mentioned limits, when creating a drive command, not only a given desired state must be checked for violation, but the area of position for valid commands must be known. In the following the calculation of borders of the limiting area is derived. For each wheel there is an upper and lower limit for the valid area described by a straight

line dividing valid and non valid areas. When taking into account a second wheel, crossings will occur. Given the valid segment of one wheel which is represented by two limiting straight lines, the limiting line from a second wheel will cross according to five different cases (see Fig. 3(a)). The valid and non valid segments of a straight line are calculated as follows. Intersection points between the number of lines must be calculated. Given a straight line G_i and a straight line



(a) (b)

Figure 3 – Section of limiting lines. a) Modes of section. b) Section Detail.

G_j, which as marginal case crosses the instantaneous center of rotation $\vec{m}$ (compare Fig. 3(b)),

$$G_i: \ \vec{p} = \vec{w}_i + k_i \cdot \vec{u}_i \quad \text{with} \quad k_i \in \mathbb{R} \tag{16}$$

$$G_j: \ \vec{m} = \vec{w}_j + k_j \cdot \vec{u}_j \quad \text{with} \quad k_j \in \mathbb{R} \tag{17}$$

the intersection point $\vec{s}$ is determined by $k_{j,S}$ for G_j and $k_{i,S}$ for G_i in

$$\vec{m} = \vec{w}_j + (k_{j,S} + k_j{}^*) \cdot \vec{u}_j \tag{18}$$

with elements

$$m_x = w_{j,x} + (k_{j,S} + k_j{}^*) \cdot u_{j,x} \quad \text{and} \tag{19}$$

$$m_y = w_{j,y} + (k_{j,S} + k_j{}^*) \cdot u_{j,y} \quad , \tag{20}$$

changing the reference point of the line equation to the intersection point. To check wether an arbitrary instantaneous center of rotation $\vec{m}$ meets the constraint $\varphi_{min,i} + \frac{\pi}{2} \leq \alpha_{M,i}$ which is defined in Eq. (15) one can use Eq. (14) with substitution of $\vec{m}$ by Eq. (19 & 20)

$$\tan\left(\varphi_{min,i} + \frac{\pi}{2}\right) \leq -\left(\frac{w_{j,x} + (k_{j,S} + k_j{}^*) \cdot u_{j,x} - w_{i,x}}{w_{j,y} + (k_{j,S} + k_j{}^*) \cdot u_{j,y} - w_{i,y}}\right) \quad . \tag{21}$$

Non valid section are determined by variables k_j^* which do not meet the condition Eq. (21). With assumption (a):

$$w_{j,y} + (k_{j,S} + k_j{}^*) \cdot u_{j,y} - w_{i,y} > 0 \tag{22}$$

Eq. (21) transforms to

$$(k_{j,S} + k_j{}^*) \cdot \left[u_{j,x} + u_{j,y} \cdot \tan\left(\varphi_{min,i} + \frac{\pi}{2}\right)\right] \leq (w_{j,x} - w_{i,x}) + (w_{j,y} - w_{i,y}) \cdot \tan\left(\varphi_{min,i} + \frac{\pi}{2}\right) \quad . \tag{23}$$

With assumption (b):

$$u_{j,x} + u_{j,y} \cdot \tan\left(\varphi_{min,i} + \frac{\pi}{2}\right) > 0 \tag{24}$$

Eq. (23) transforms to

$$k_j{}^* \leq \frac{(w_{j,x} - w_{i,x}) + (w_{j,y} - w_{i,y}) \cdot \tan\left(\varphi_{min,i} + \frac{\pi}{2}\right)}{u_{j,x} + u_{j,y} \cdot \tan\left(\varphi_{min,i} + \frac{\pi}{2}\right)} - k_{j,S} \quad . \tag{25}$$

If one of the assumptions is violated, in each case the relational operator obverts. Both assumptions violated, the result stays the same. The testing result is determined by the sign of k_j^*:

- $k_j{}^* \leq 0$:
  Valid points are located in negative direction from intersection, non valid points are located in positive direction respectively.
- $k_j{}^* \geq 0$:
  Valid points are located in positive direction from intersection, non valid points are located in negative direction respectively.

This way for each straight line the direction of valid and non valid instantaneous centers of rotation is identified. As this assignment switches in the wheels steering center the location of intersection of the crossing straight line must be considered. Furthermore, the relationship between the angles of crossing and crossed straight line determines, wether the crossing lines switches in positive or negative direction into the non valid section. Summing up: the solution to the problem is deduced from the position of the intersection and the angle difference of the crossing lines.

The subsumption of these conditions lead to a decision matrix according to Fig. 4, which for lower (4(a)) and upper (4(b)) limit depends on the sign of the angle-difference $\Delta\alpha_{i,j}$ and of the location of intersection on the straight line $k_{i,S}$. This sequence is rather efficient, as it depends on two comparisons and the determination of one point of intersection.

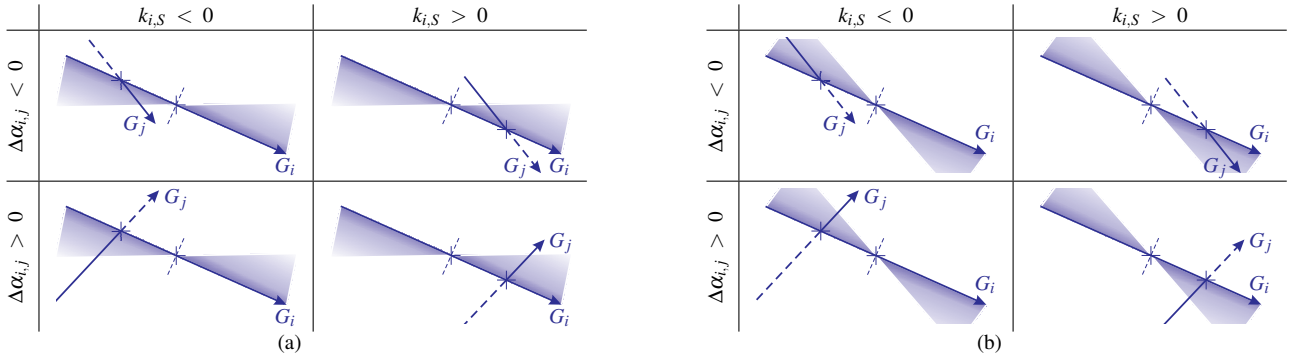


Figure 4 – Determination of valid and non valid segments by simple comparisons. a) Configurations for $\alpha_{i,min} \leq \alpha_i$. b) Configurations for $\alpha_i \leq \alpha_{i,max}$.

This information leads to knowledge about position and size of non valid sections. The combination of sections violating at least one constraint after inversion delivers valid segments: a section which is rendered void cannot become valid in the following. The sections not determined as void must meet all constraints. The combination of the valid segments of all lines represent the area of valid locations for the instantaneous center of rotation for the next time step. The combination of segments is depicted in Fig. 5.

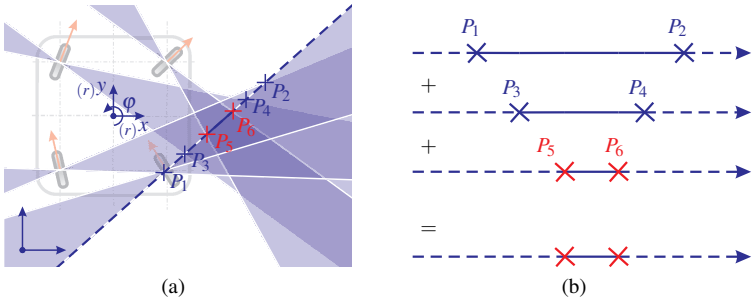


Figure 5 – Construction of valid segments. a) Composition of non valid sections. b) Resulting valid section between P_5 and P_6.

COMMAND GENERATION UNDER CONSTRAINTS

The algorithm described in the previous chapters is capable of identifying non feasible drive commands which violate dynamic constraints due to set value limitations. As a region of valid drive commands is calculated with the region for the instantaneous center of rotation, a non valid drive command can be substituted by a valid one. The distance between both instantaneous centers or rotation is calculated by

$$\vec{d} = \vec{m}^* - \vec{m} \quad . \tag{26}$$

Here the optimum substitute is chosen as a command whose position of the instantaneous center of rotation is closest to the valid region. The criterion for optimality is expressed as

$$\begin{aligned} d &= \|\vec{m}^* - \vec{m}\| \quad \text{or} \\ d &= \sqrt{(m_x{}^* - m_x)^2 + (m_y{}^* - m_y)^2} \end{aligned} \tag{27}$$

for the error between the original command with $\vec{m}^*$ and the substitute $\vec{m}$. The minimum distance can only be determined on the edge of the valid region, which induces that $\vec{m} \in G_i$, and by

$$G_i: \quad \vec{m} = \vec{w}_i + k_i \cdot \vec{u}_i \quad \text{mit} \quad k_i \in \mathbb{R}$$

with

$$\begin{aligned} m_x &= w_{i,x} + k_i \cdot u_{i,x} \quad \text{and} \\ m_y &= w_{i,y} + k_i \cdot u_{i,y} \end{aligned}$$

changes the optimality criterion to

$$d = \sqrt{(m_x^* - w_{i,x} + k_i \cdot u_{i,x})^2 + (m_y^* - w_{i,y} + k_i \cdot u_{i,y})^2} \quad . \tag{28}$$

The optimum substitute can be determined analytically by calculating the distance of the instantaneous center of rotation from a line (see Fig. 6). For those line segments, that cannot be linked to the instantaneous center of rotation by an orthogonal vector, because the resulting point is not within the limits of the edge of the valid section, the minimum is either the left or right segment limit of the edge of the valid section as shown in Fig. 6(b).

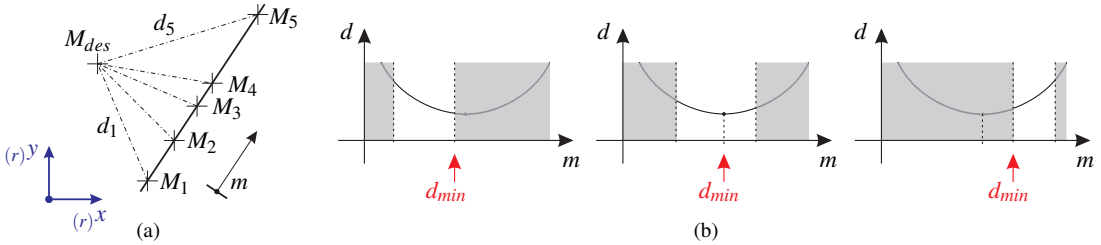


Figure 6 – Optimality criterion.

It is assumed, that not only the steering wheel dynamics, but also the dynamics of the robots translational velocity is limited. To generate a substitute drive command, not only the instantaneous center of rotation but also the substitute angular velocity is determined. Here the norm of the desired translational robot velocity is used to calculated the optimized angular velocity and the resulting optimized translational velocities for the robot:

$$|\vec{v}_{robot}| = \sqrt{\dot{x}_{des}^2 + \dot{y}_{des}^2} \tag{29}$$

$$\omega_{opt} = \frac{|\vec{v}_{robot}|}{\sqrt{m_{opt,x}^2 + m_{opt,y}^2}} \tag{30}$$

$$\vec{v}_{robot,opt} = \vec{\omega}_{opt} \times -\vec{m}_{opt} \tag{31}$$

Special Cases

The algorithm described so far works only while moving with nonzero angular velocity, but fails to cover the case when moving straight. The instantaneous center of rotation, which is mandatory for the algorithm to work, moves towards infinity, when moving with zero angular velocity. However, in practise, the instantaneous center of rotation is assumed on a large but finite distance to the robot, leading to a negligible steering angle. This measure has only minor numerical effect, allowing to work with the algorithm as described.

A second effect is as follows: The influence of the algorithm may lead to unwanted behavior if the instantaneous center of rotation switches from one side of the robot towards the opposite side, which occurs, when following an s-shaped path. For a short time purely translational movement occurs, which arouses areas of feasible instantaneous centers of rotation on both sides of the robot. Here the sign of the angular velocity switches. When the angular velocity approaches zero, the instantaneous center of rotation departs from the robot. It switches sides in infinity and approaches the robot from infinity on the opposite side.

If the feasible instantaneous center of rotation cannot follow the desired one a case may occur, in which the instantaneous center of rotation of the desired drive command M_{des} has already switched the sign of the angular velocity, while there is no feasible area on that side of the robot as depicted in Fig. 7. The path of the instantaneous center of rotation, which is meant to move through infinity, will be changed according to the algorithm such, that the shortest distance between the feasible and desired one is chosen, leading it into the wrong direction through the center of the robot $M_{opt,1}$. To avoid this behavior, the plane is divided by the center axis of the robot into two parts. If the instantaneous center of rotation crosses this axis from a distance with certain threshold, the above mentioned case is assumed, and the instantaneous center of rotation is lead through infinity $M_{opt,2}$. If the actual instantaneous center of rotation is very close to the axis when switching sides, direct crossing of the axis is permitted.

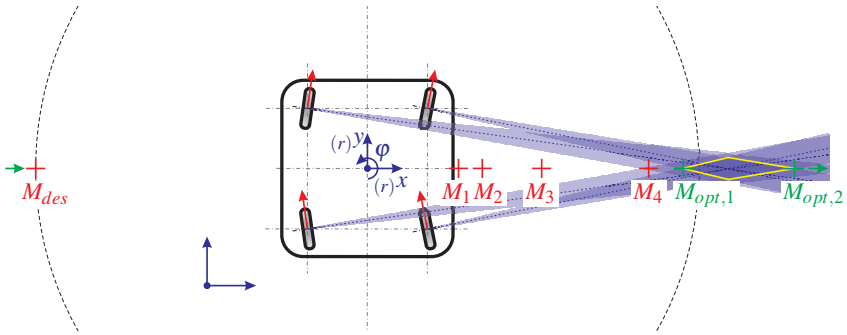
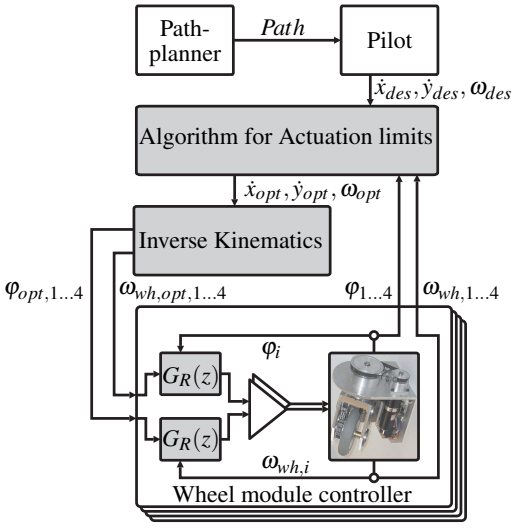


Figure 7 – Unwanted behavior while traveling on s-shaped path.

EXPERIMENTAL RESULTS

The algorithm described above is implemented on an experimental platform shown in Fig. 8(a). The platform is equipped with four identical wheel modules, which are actively driven an steered. The vertical steering axis of the wheel module runs through the center of the wheel. The low level control algorithms are implemented on an a *mpc-555* μ-controller running *RTOS-UH* real time operating system. The control structure is depicted in Fig. 8(b), the software modules implemented on the μ-controller are highlighted in grey color. The overall setup contains additional processing

(a)



(b)

Figure 8 – Experimental setup. a) Wheeled mobile robot. b) Control structure, including the algorithm introduced in this paper.

units for higher order control loops. A path planning module calculates a path represented by straight line and polar spline segments. The Path is forwarded to the pilot, which is in charge of path control (the localization module it relies on for position information is not shown here). The interface between the pilot and the mobile platform is done according to the unicycle kinematics, containing translational velocities $\dot{x}_{des}$, $\dot{y}_{des}$ and the angular velocity ω_{des} around the vertical axis, each specified within a platform fixed coordinate system. The desired velocities are fed into the algorithm, which is presented in this paper along with the measured steering angles $\varphi_{1...4}$ and wheel velocities $\omega_{wh,1...4}$ of each wheel. The output of the algorithm is an optimized set of velocities $\dot{x}_{opt}$, $\dot{y}_{opt}$ and ω_{opt}, which is fed into the inverse kinematics module to acquire set values for steering angles $\varphi_{opt,1...4}$ and wheel velocities $\omega_{wh,opt,1...4}$. These set values are controlled within single axis control loops for each wheel module (wheel module controller). All state variables can be logged to acquire measurement data.

The algorithm has been tested on the mobile platform presented above with an exemplary path shown in Fig. 9(c). The functionality of the algorithm is proven, as depicted in Fig. 9. In Fig. 9(a) steering angles $\varphi_{1...4}$, steering velocities $\dot{\varphi}_{1...4}$ and steering accelerations $\ddot{\varphi}_{1...4}$ are depicted without limiting dynamics. The same trajectory is recorded in Fig. 9(b) with limits applied to $|\dot{\varphi}_{1...4,max}| = \frac{\pi}{2} \cdot \frac{1}{s}$ and $|\ddot{\varphi}_{1...4}| = \frac{10\pi}{2} \cdot \frac{1}{s^2}$. The figures show, that the wheels states are kept within the desired limits. As the desired set values are altered there is an influence on the robot path. However, the desired velocities are determined as non feasible, which justifies manipulations. Path deviations are assumed to be corrected in higher order control loops, which can be shown to work in practise. When plotting the path, deviations are very small as shown in

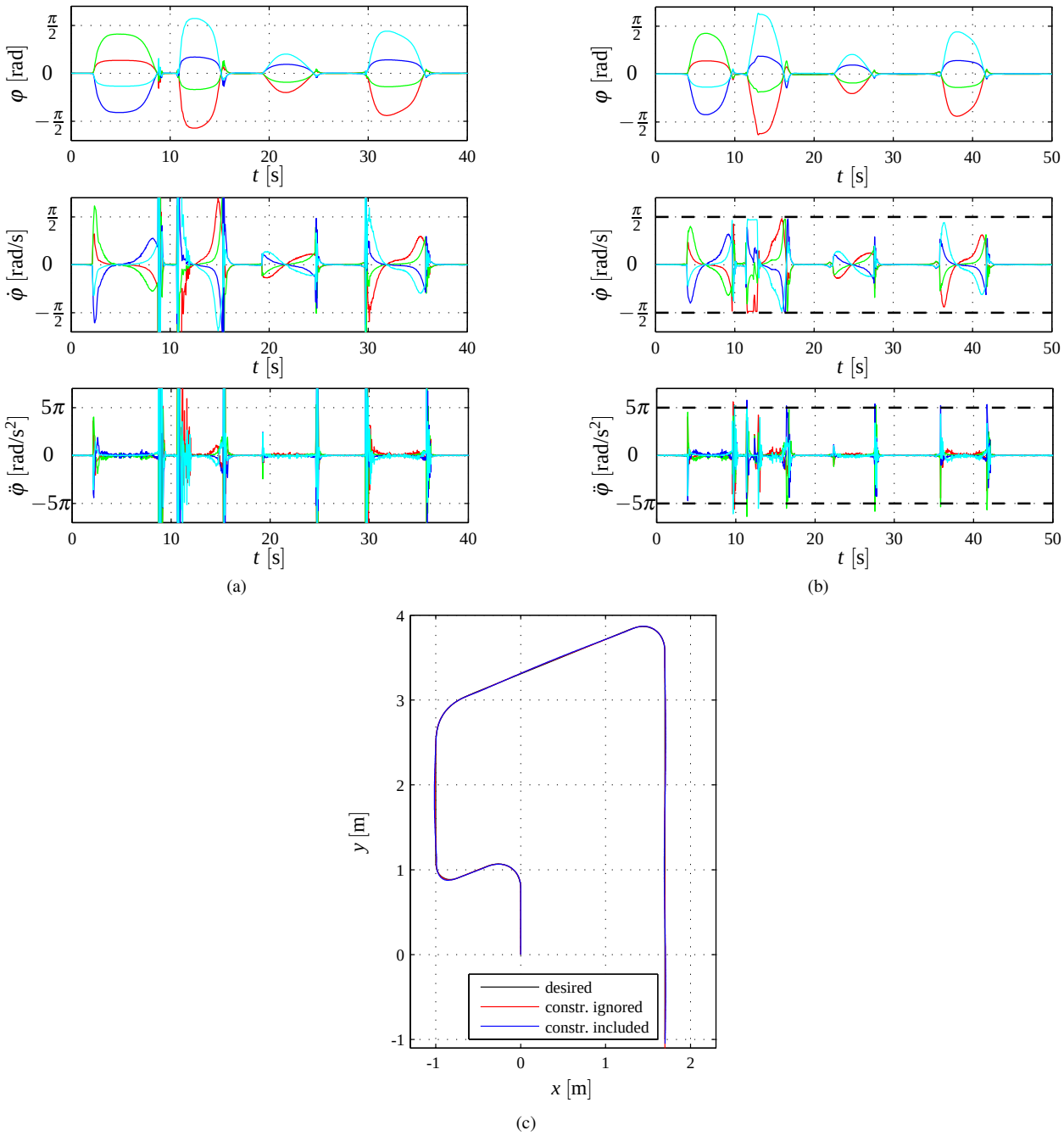


Figure 9 – Influence of algorithm. a) Without limits. b) With limits. c) With limits.

Fig. 9(c). The additional deviations of the algorithm presented can be neglected, which renders the algorithm applicable in a general sense.

CONCLUSIONS

In this paper a computationally efficient method for consideration of control variable limits in redundantly actuated wheeled mobile robots is presented. The algorithm takes into account the limited dynamics of the steering angle, minimizing violations of the kinematic constraints which may lead to non coordinated movement of the robot. The method is based on the calculation of a feasible area to place the instantaneous center of rotation in the next time step such, that every single wheel is capable of reaching the desired state. In case the desired drive command is not executable, an alternative drive command is calculated.

The method was tested on an exemplary wheeled mobile robot. The average execution time using a *mpc-555* μ-controller was at 3.3 ms, which is far beyond the control rate of 20 Hz. Experimental results show, that the limits adjusted are met and that path deviation induced by the algorithm can be corrected by higher order control loops. Usage of the

algorithm not only for optimization of a given command, but also for in path planning can be used. This especially refers to section **CONTROL VARIABLE LIMITS** where the feasible area can be used as a side condition in the path planning algorithm.

REFERENCES

Abdellatif, H., Michaelsen, A. and Heimann, B. , 2003, Position estimation and control of an 8-dof omnidirectional mobile robot, Proc. of Raad02, 12th Internatinal Workshop on Robotics in Alpe-Adria-Danube Region.

Abdellatif, H., Weidemann, D., Michaelsen, A., Grotjahn, M. and Heimann, B. , 2002, Control of the omnidirectional mobile robot marge, Proc. of the 14th CISM-IFToMM Symp. on the Theory and Practice of Robots and Manipulators (RoManSy), pp. 101–108.

Bhatt, R., Tang, C. P. and Krovi, V. , 2004, Geometric motion planning and formation optimization for a fleet of nonholonomic wheeled mobile robots, Proceedings of the 2004 IEEE International Conference on Robotics & Automation, pp. 3276–3281.

Borenstein, J. and Feng, L. 1994, UMBmark A Method for Measuring, Comparing, and Correcting Dead-reckoning Errors in Mobile Robots, number UM-MEAM-94-22, Technical Report.

Campion, G., Bastin, G. and AndrCa-Novel, B. D. , 1996, Structural properties and classification of kinematic and dynamic models of wheeled mobile robots, IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION 12(1): 47–62.

Castellanos, J. A. and Tardós, J. D. 1999, Mobile Robot Localization and Map Building, Kluwer Academic Publishers.

Horn, J. P. 1997, Bahnführung eines mobilen Roboters mittels absoluter Lagebestimmung durch Fusion von Entfernungsbild- und Koppelnavigationsdaten, Dissertation, Lehrstuhl für Steuerungs- und Regelungstechnik, Technische Universität München, VDI-Verlag, Düsseldorf.

Luca, A. D., Oriolo, G. and Vendittelli, M. , 2000, Stabilization of the unicycle via dynamic feedback linearization, Proceedings of the 6th IFAC Symp. on Robot Control, Vienna, Austria, pp. 397–402.

Oriolo, G., Luca, A. D. and Vendittelli, M. , 2002, Wmr control via dynamic feedback linearization: Design, implementation, and experimental validation, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 10, NO. 6 10(6): 835–852.

RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper.