

OmniMan: A Mobile Assistive Robot for Intralogistics Applications

Christof Röhrig, *Member, IAENG*, and Daniel Heß

Abstract—In human-robot collaboration, robots interact directly and safely with humans in a shared workspace without protective guards. This paper presents the design of *OmniMan*, which is a mobile manipulator for human-robot collaboration consisting of an omnidirectional mobile platform and a lightweight robotic arm. Mobile manipulators extend the workspace of manipulators (robotic arms) by mounting them on mobile platforms. The mobile platform is driven by Mecanum wheels, which provide 3 degrees of freedom in motion. The paper develops the kinematic model of the whole system including platform and manipulator. Furthermore the overall controller structure for *OmniMan* including the real-time synchronization of platform and manipulator is described.

Index Terms—Mobile Manipulation, Human Robot Collaboration, Mecanum Wheels

I. INTRODUCTION

IN human-robot collaboration humans interact with robots or manipulators in a shared workspace without protective guards. Manipulators have been used in a large range of applications mainly in the production industry, but their application is limited in scenarios which need a very large working space such as aerospace manufacturing, ship building, and wind turbine manufacturing. Mobile manipulation is a solution to overcome these limitations and counts to be a key technology not only for the production industry but also for professional service robotics such as intralogistics. Furthermore mobile manipulators find their applications in virtual or real laboratories for research and education [1], [2].

In the last years, research institutes have developed their own mobile manipulators based on commercial available mobile platforms and robotic arms. *Cody* from Georgia Institute of Technology consists of two arms from MEKA Robotics and a Mecanum wheeled Segway platform [3]. *Cody* was build mainly for research on service robotics in the health care domain. Another example in this area is *POLAR* (PersOnAL Assistant Robot) from Cornell University, which consists of a 7 DoF (degrees of freedom) Barrett arm mounted on a Segway Omni base [4]. *TOMM* from TU Munich is a wheeled humanoid robot, which consists of a Mecanum wheeled mobile Platform and two UR5 robotic arms [5]. Other popular examples in this domain are Willow Garage's PR2 and the Fraunhofer IPA's Care-O-bot 4.

Compared to the domains of professional and domestic service, mobile manipulators for industrial applications operate in more structured environments. However, they require a higher level of operational efficiency, e.g. in terms of

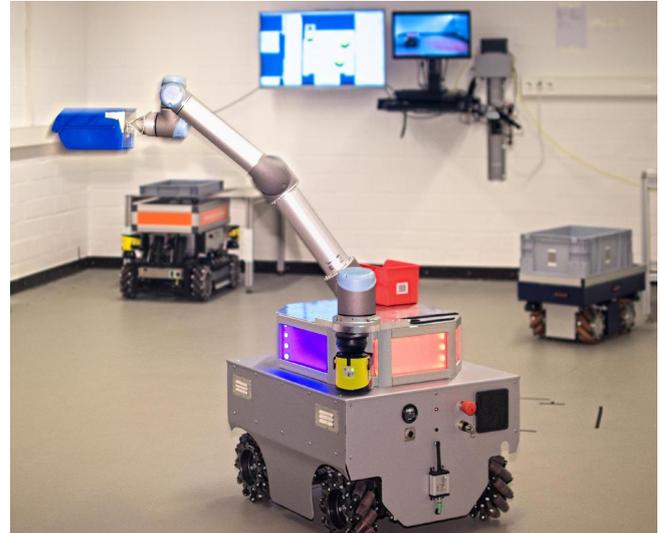


Figure 1. Mobile manipulator *OmniMan*: UR5 robotic arm on a Mecanum wheeled mobile platform

speed, accuracy and robustness, to be suitable for industrial applications [6]. Popular examples of mobile manipulators for industrial environments are KUKA's *omniRob* and *Moiros* [7] as well as *ANNIE* and *LiSA* from Fraunhofer IFF, which are all based on Mecanum wheeled platforms. *omiRob* is used mainly in intralogistics applications [8], [9], [10], where human-robot collaboration is often needed.

A Mecanum wheel consists of a central hub with free moving rollers, which are usually mounted at $\pm 45^\circ$ angles around the hubs' periphery. The outline of the rollers is such that the projection of the wheel appears to be circular. A mobile platform driven by Mecanum wheels provides 3 degrees of freedom in motion. Usually such platforms consists of four or more wheels. A typical configuration is the four-wheeled one of the *URANUS* omnidirectional mobile robot [11]. The drive structure of a Mecanum wheeled mobile platform with four or more wheels is over-actuated, which means that actuation conflicts may occur.

This paper extends the work presented in [12]. The kinematic equations of a mobile manipulator including 6-DoF manipulator and omnidirectional platform are developed. The paper develops a generalized kinematic model of an over-actuated Mecanum wheeled mobile platform, which includes the kinematic motion constraints of the system and that is valid for Omni wheels as well. Furthermore the paper describes the overall controller structure of our mobile manipulator *OmniMan* (Fig. 1). *OmniMan* consists of an Universal Robots UR5 robotic arm mounted on Mecanum wheeled mobile platform from MIAG Fahrzeugbau GmbH (Braunschweig, Germany). The paper describes different options for interfacing an UR arm controller to a central

This work was supported by the Ministry of Culture and Science of the German State of North Rhine-Westphalia (grant number 005-1703-0008).

University of Applied Sciences and Arts in Dortmund, Institute for the Digital Transformation of Application and Living Domains (IDiAL), Otto-Hahn-Str. 23, 44227 Dortmund, Germany Web: www.fh-dortmund.de/en/idial/, Email: roehrig@ieee.org

controller PC as well as to synchronize the movements of the arm with the platform in real-time.

II. PROBLEM FORMULATION

We consider the problem of motion control of a mobile manipulator, which consists of a robotic arm and a mobile platform. The robotic arm provides 6 DoF in 3D Cartesian space, the omnidirectional mobile platform provides additional 3 DoF in 2D space (position and heading). Therefore, the omnidirectional mobile manipulator is an over-determined system, where the additional degrees of freedom in configuration space can be used to optimize the movement of the manipulator. In general, the forward kinematics of a mobile manipulator can be described by

$$\mathbf{x} = \mathbf{f}(\mathbf{q}), \text{ with } \mathbf{x} = [x, y, z, \alpha, \beta, \gamma]^T, \text{ and } \mathbf{q} = \begin{pmatrix} \mathbf{q}_p \\ \mathbf{q}_a \end{pmatrix}, \quad (1)$$

where \mathbf{x} is the pose of the mobile manipulator's tool frame with respect to the world frame ($\mathbf{x} \in \mathbb{R}^3 \times \text{SO}(3)$), \mathbf{q} are the generalized coordinates in the configuration space (C-space) with the 2D pose of the mobile platform $\mathbf{q}_p = [x_p, y_p, \theta_p]^T$, $\mathbf{x}_p \in \mathbb{R}^2 \times \mathbb{S}^1$ and the joint angles of the arm $\mathbf{q}_a = [\theta_1, \dots, \theta_6]^T$. The inverse kinematics of the system

$$\mathbf{q} = \mathbf{f}^{-1}(\mathbf{x}) \quad (2)$$

is over-determined and can be solved by numerical methods in order to optimize the motion in C-space.

The mobile platform is driven by 4 Mecanum wheels. A Mecanum wheeled mobile platform provides any desired motion in x - and y -direction and rotation θ around the z -axis, simultaneously. The mobile platform moves in 2D space, the pose of the platform in the world frame is defined as $\mathbf{x}_p = (x_p, y_p, \theta_p)^T$ (see Fig. 2). The robot frame is a

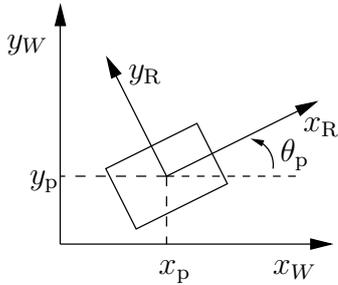


Figure 2. Definition of pose $\mathbf{x}_p = (x_p, y_p, \theta_p)^T$ and robot frame

coordinate frame, which is fixed at the mobile platform. Velocities in the robot frame (F_R) can be transformed into the world frame (F_W), as a function of the heading of the platform ($\theta = \theta_p$):

$$\dot{\mathbf{x}}_R = \mathbf{R}(\theta) \dot{\mathbf{x}}_W, \Rightarrow \dot{\mathbf{x}}_W = \mathbf{R}^{-1}(\theta) \dot{\mathbf{x}}_R \quad (3)$$

$$\text{with } \dot{\mathbf{x}}_W = \begin{pmatrix} \dot{x}_W \\ \dot{y}_W \\ \dot{\theta} \end{pmatrix}, \dot{\mathbf{x}}_R = \begin{pmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta} \end{pmatrix},$$

$$\mathbf{R}(\theta) = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The kinematic model of mobile platforms equipped with n Mecanum wheels is well known (see [11], [13]). The inverse

kinematics of the platform, which transform the velocity of the platform into wheel velocities, can be described by linear equations in the robot frame. It can be written under the general matrix form:

$$\dot{\boldsymbol{\varphi}} = \mathbf{J}_w \dot{\mathbf{x}}_R, \text{ with } \mathbf{J}_w \in \mathbb{R}^{n \times 3}, \quad (4)$$

where $\dot{\boldsymbol{\varphi}} = (\dot{\varphi}_1, \dot{\varphi}_2, \dots, \dot{\varphi}_n)^T$ are the angular velocities of the wheels and \mathbf{J}_w is a Jacobi matrix with constant parameters. For a platform equipped with $n > 3$ Mecanum wheels, the forward kinematic equations are over-determined. The forward kinematics can be obtained by using a least square approach and applying the Moore-Penrose pseudo-inverse to \mathbf{J} :

$$\dot{\mathbf{x}}_R = \mathbf{J}_w^+ \dot{\boldsymbol{\varphi}}, \text{ with } \mathbf{J}_w^+ = (\mathbf{J}_w^T \mathbf{J}_w)^{-1} \mathbf{J}_w^T \quad (5)$$

In contrast to the overdeterminacy in the mobile manipulator, the over-actuated drive structure of the platform can not be used to achieve additional degrees of freedom in C-space. The overdeterminacy of the platform yields to motion constraints, which must be fulfilled to avoid additional wheel slippage.

III. KINEMATIC MODEL OF THE MOBILE MANIPULATOR

A. Kinematic Model of the Universal UR5

The UR5 is a lightweight 6 DoF industrial manipulator manufactured by Universal Robots (UR). It has a weight of 18.4 kg, a reach of 85 cm and a maximal payload of 5 kg [14].

The forward kinematics of the UR5 arm can be easily described by the Denavit-Hartenberg (DH) method. The homogeneous transformation associated with one link can be described by

$$\begin{aligned} {}^{i-1}\mathbf{T}_i &= \begin{pmatrix} {}^{i-1}\mathbf{R}_i & {}^{i-1}\mathbf{p}_i \\ \mathbf{0} & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (6)$$

where ${}^{i-1}\mathbf{R}_i \in \text{SO}(3)$ represents the orientation, ${}^{i-1}\mathbf{p}_i \in \mathbb{R}^3$ the position, $\theta_i, d_i, a_i, \alpha_i$ are the DH parameters. The DH parameters of the UR5 arm can be found in Fig. 3. The

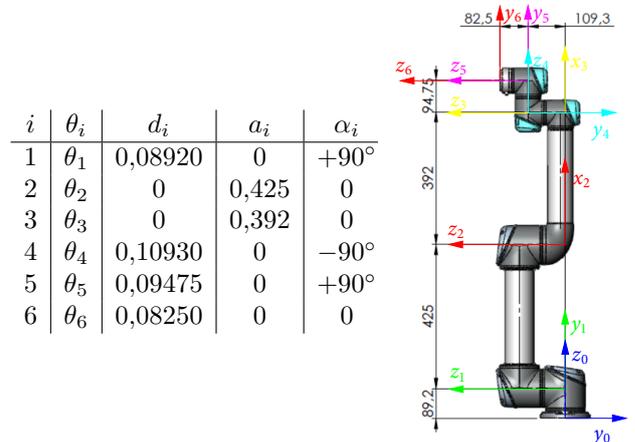


Figure 3. UR5 robotic arm with DH frames and DH parameters

forward kinematics of a 6 DoF robotic arm can be described by

$${}^0T_6 = \prod_{i=1}^6 {}^{i-1}T_i, \quad (7)$$

where ${}^{i-1}T_i$ are the DH matrices. The forward kinematics of the robotic arm can be described in compact form:

$$\mathbf{x} = \mathbf{f}_a(\mathbf{q}_a), \text{ with } \mathbf{x} = [x, y, z, \alpha, \beta, \gamma]^T, \\ \mathbf{q}_a = [\theta_1, \dots, \theta_6]^T, \quad (8)$$

where \mathbf{x} is the pose of the robotic arm in Cartesian space and \mathbf{q}_a are the joint variables of the arm. The inverse kinematics describe the joint angles as a function of the pose of the robotic arm

$$\mathbf{q}_a = \mathbf{f}_a^{-1}(\mathbf{x}), \quad (9)$$

and can not be solved in analytic form in general. For the Universal UR5 arm, an analytic solution for the inverse kinematics was developed by Kelsey Hawkins [15]. We solve the inverse kinematics of the mobile manipulator using a numerical solution, in order to use the additional degrees of freedom of the mobile manipulator for optimization of the movement in C-space. The numerical solution uses the Jacobian

$$\mathbf{J}_a(\mathbf{q}) = \frac{\partial \mathbf{f}_a(\mathbf{q}_a)}{\partial \mathbf{q}_a} = \begin{pmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} & \dots & \frac{\partial f_1}{\partial \theta_6} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_6}{\partial \theta_1} & \frac{\partial f_6}{\partial \theta_2} & \dots & \frac{\partial f_6}{\partial \theta_6} \end{pmatrix} \quad (10)$$

The i -th column of \mathbf{J}_a can be obtained numerically

$$\mathbf{J}_{a,i} = \begin{pmatrix} \mathbf{J}_v \\ \mathbf{J}_\omega \end{pmatrix} = \begin{pmatrix} \mathbf{z}_{i-1} \times (\mathbf{p}_6 - \mathbf{p}_{i-1}) \\ \mathbf{z}_{i-1} \end{pmatrix}, \quad (11)$$

where \mathbf{p}_6 is the position of the TCP. The vectors \mathbf{p}_{i-1} and \mathbf{z}_{i-1} can be found in the DH matrix

$${}^0T_{i-1} = \prod_{j=1}^{i-1} {}^{j-1}T_j = \begin{pmatrix} \mathbf{x}_{i-1} & \mathbf{y}_{i-1} & \mathbf{z}_{i-1} & \mathbf{p}_{i-1} \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (12)$$

B. Motion Model with Constraints for Mobile Platforms with Mecanum Wheels

In this section, the kinematics of an omnidirectional mobile platform with n Mecanum wheels is developed. A Mecanum wheel consists of a central hub with free moving rollers. A mobile platform driven by 3 or more Mecanum wheels provides 3 DoF in motion. For wheel i , we define the wheel frame F_i and the roller frame $F_{r,i}$, which are in a fixed position in the robot frame F_R (see Fig. 4). For simplification, the kinematic model is described in 2D, which means that all frames lie within the xy -plane. The position of the wheel frame F_i with respect to the robot frame is described by the 3 constant parameters: α_i , l_i and δ_i , where δ_i defines the rotation angle between F_i and the F_R and is usually equal to zero. γ_i defines the angle of the roller with respect to the wheel frame. $\varphi_i(t)$ drives the wheel and defines the rotation angle of the wheel around its horizontal axis of rotation. The wheel is driven in the direction of its x_i axis. The wheel has one contact point to the plane and is able to rotate around this point (rotation around its z_i axis).

For simplification, it is assumed that there is only one roller that has contact to the floor and that the contact point stays always in the center of the roller (and the wheel). The roller frame $F_{r,i}$ as well as the wheel frame F_i has its origin in this point of contact. The $x_{r,i}$ axis lies in the shaft of the roller. The wheel is able to move free in direction of the $y_{r,i}$ axis.

The movements of the platform yield to the velocities \dot{x}_R , \dot{y}_R , $l \cdot \dot{\theta}$ in the contact point of the wheel, which can be transformed to the roller frame $F_{r,i}$:

$$\dot{x}_{r,i}^R = \dot{x}_R \cos(\gamma_i + \delta_i) + \dot{y}_R \sin(\gamma_i + \delta_i) + l_i \dot{\theta} \cos(\alpha_i + \pi/2 - \delta_i - \gamma_i) \quad (13)$$

$$\dot{y}_{r,i}^R = -\dot{x}_R \sin(\gamma_i + \delta_i) + \dot{y}_R \cos(\gamma_i + \delta_i) + l_i \dot{\theta} \sin(\alpha_i + \pi/2 - \delta_i - \gamma_i) \quad (14)$$

where γ_i is the rotation angle between the wheel frame and the roller frame (usually $\pm 45^\circ$) and δ_i is the angle between the wheel frame and the robot frame (usually 0°). The rotation of the wheel drives the velocity $\dot{x}_i = r \cdot \dot{\varphi}_i$ in the point of contact, which can be transformed in x and y components of the roller frame $K_{r,i}$:

$$\dot{x}_{r,i}^\varphi = r \dot{\varphi}_i \cos(\gamma_i), \quad \dot{y}_{r,i}^\varphi = -r \dot{\varphi}_i \sin(\gamma_i) \quad (15)$$

Since the roller can not move in direction of its shaft, $\dot{x}_{r,i}^R = \dot{x}_{r,i}^\varphi$ must be true, which leads to

$$\dot{x}_R \cos(\gamma_i + \delta_i) + \dot{y}_R \sin(\gamma_i + \delta_i) + l_i \dot{\theta} \cos(\alpha_i + \pi/2 - \delta_i - \gamma_i) = r \dot{\varphi}_i \cos(\gamma_i), \quad (16)$$

and finally to the inverse kinematic equation of wheel i

$$\dot{\varphi}_i = \frac{1}{r \cdot \cos(\gamma_i)} (\cos(\delta_i + \gamma_i), \sin(\delta_i + \gamma_i), l_i \sin(\delta_i + \gamma_i - \alpha_i)) \dot{\mathbf{x}}_R. \quad (17)$$

For a platform equipped with n Mecanum wheels, (17) can be used to obtain the inverse kinematics of the platform as

$$\dot{\varphi} = \mathbf{J}_w \dot{\mathbf{x}}_R, \text{ with } \mathbf{J}_w \in \mathbb{R}^{n \times 3} \quad (18)$$

If γ_i is set equal to zero, the Mecanum wheel becomes an Omni wheel and therefore (17) can be used to model the inverse kinematics of an Omni wheel driven platform as well.

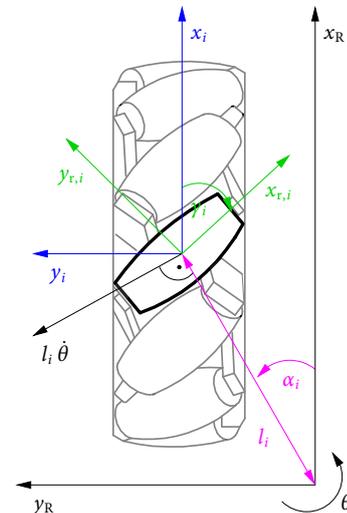


Figure 4. Mecanum wheel with frames

For a platform with more than 3 Mecanum wheels, the kinematics are over-determined, which means that the wheel speed of $m = n - 3$ of the n wheels are a linear combination of the other ones. Only 3 rows of J_w are linear independent, m rows are a linear combination of the first 3 rows. To obtain the kinematic motion constraints, J_w can be split up into 2 sub-matrices:

$$\mathbf{J}_w = \begin{pmatrix} \mathbf{J}_3 \\ \mathbf{J}_m \end{pmatrix}, \text{ with } \mathbf{J}_3 \in \mathbb{R}^{3 \times 3}, \mathbf{J}_m \in \mathbb{R}^{m \times 3}, m = n - 3 \quad (19)$$

The velocities in the robot frame can be defined by 3 wheels:

$$\begin{pmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta} \end{pmatrix} = \mathbf{J}_3^{-1} \begin{pmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \end{pmatrix}$$

this leads to m kinematic motion constraints:

$$\mathbf{J}_m \mathbf{J}_3^{-1} \begin{pmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \end{pmatrix} = \begin{pmatrix} \dot{\varphi}_4 \\ \vdots \\ \dot{\varphi}_n \end{pmatrix},$$

which can be expressed in general matrix form

$$\mathbf{T} \dot{\varphi} = \mathbf{0}, \text{ with } \mathbf{T} = (\mathbf{J}_m \mathbf{J}_3^{-1}, -\mathbf{I}_m), \quad (20)$$

where \mathbf{I} denotes the identity matrix. We define a vector of angular error velocities $\dot{\epsilon} = (\dot{\epsilon}_1 \dots \dot{\epsilon}_m)^T$ to detect violations of the kinematic constraints:

$$\dot{\epsilon} = \mathbf{T} \dot{\varphi}, \quad (21)$$

\mathbf{T} and $\dot{\epsilon}$ can be used to augment the kinematics of the platform

$$\dot{x}_g = \mathbf{J}_g^{-1} \dot{\varphi}, \text{ with } \dot{x}_g = \begin{pmatrix} \dot{x}_R \\ \dot{\epsilon} \end{pmatrix}, \mathbf{J}_g^{-1} = \begin{pmatrix} \mathbf{J}_w^+ \\ \mathbf{T} \end{pmatrix} \quad (22)$$

where \mathbf{J}_g is an invertible square matrix, which describes the augmented inverse kinematics of the system:

$$\dot{\varphi} = \mathbf{J}_g \dot{x}_g \quad (23)$$

The sub-vector $\dot{\epsilon}$ in \dot{x}_g can be used to control the kinematic constraints.

A typical configuration of a Mecanum wheeled platform consists of 4 wheels (see Fig. 5). The positions of the wheels

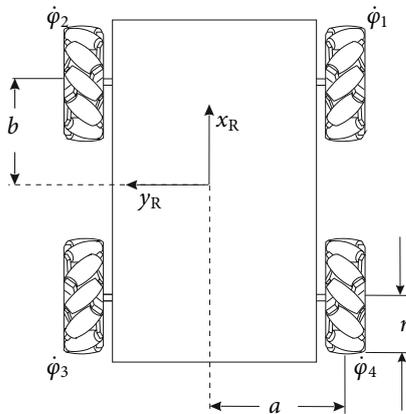


Figure 5. Omnidirectional platform with Mecanum wheels (top view)

with respect to the robot frame can be defined as $\delta_i = 0$ and $l_i = \sqrt{a^2 + b^2}$. This leads to the configuration parameters shown in Table I. These parameters in conjunction with

Table I
CONFIGURATION OF THE PLATFORM IN FIG. 5

i	α_i	γ_i
1	$-\arctan\left(\frac{a}{b}\right)$	$+\frac{\pi}{4}$
2	$+\arctan\left(\frac{a}{b}\right)$	$-\frac{\pi}{4}$
3	$\frac{\pi}{2} + \arctan\left(\frac{a}{b}\right)$	$+\frac{\pi}{4}$
4	$-\frac{\pi}{2} - \arctan\left(\frac{a}{b}\right)$	$-\frac{\pi}{4}$

(17) yield to the inverse kinematics

$$\begin{pmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \\ \dot{\varphi}_4 \end{pmatrix} = \mathbf{J}_w \begin{pmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta} \end{pmatrix}, \text{ with } \mathbf{J}_w = \frac{1}{r} \begin{pmatrix} 1 & 1 & (a+b) \\ 1 & -1 & -(a+b) \\ 1 & 1 & -(a+b) \\ 1 & -1 & (a+b) \end{pmatrix} \quad (24)$$

The kinematic constraint can be obtained using (20)

$$0 = \mathbf{T} \dot{\varphi}, \text{ with } \mathbf{T} = (1, 1, -1, -1), \quad (25)$$

which leads to the augmented forward kinematics

$$\dot{x}_g = \mathbf{J}_g^{-1} \dot{\varphi}, \text{ with } \dot{x}_g = \begin{pmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta} \\ \dot{\epsilon}_1 \end{pmatrix}, \dot{\varphi} = \begin{pmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \\ \dot{\varphi}_4 \end{pmatrix}, \quad (26)$$

$$\mathbf{J}_g^{-1} = \frac{r}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ \frac{1}{a+b} & \frac{-1}{a+b} & \frac{-1}{a+b} & \frac{1}{a+b} \\ \frac{4}{r} & \frac{4}{r} & \frac{-4}{r} & \frac{-4}{r} \end{pmatrix}$$

and the augmented inverse kinematics

$$\dot{\varphi} = \mathbf{J}_g \dot{x}_g, \text{ with } \mathbf{J}_g = \frac{1}{r} \begin{pmatrix} 1 & 1 & (a+b) & 1 \\ 1 & -1 & -(a+b) & 1 \\ 1 & 1 & -(a+b) & -1 \\ 1 & -1 & (a+b) & -1 \end{pmatrix} \quad (27)$$

where r is the radius of the wheels, a and b are given by the dimension of the platform (see Fig. 5). Eqn. (26) is used in the motion controller of the platform to execute odometry and to sense the coupling error. Eqn. (27) is used in the motion controller to control the speeds in robot frame and to control the kinematic constraints.

C. Experimental Results

In order to evaluate the effectiveness of the coupling controller several experiments are performed with some of our mobile platforms, which are shown in Fig 1. The results of one of these experiments are shown in Fig. 7. The platform moves a path with straight lines forwards, sideways and backwards. After that, it moves a circular path with two rotations back to the starting point. The red curve in the picture shows the coupling error ϵ_1 . The five movements of the platform yield to a coupling error different from zero, where the individual movements can be easily distinguished. The coupling error may be caused by parameter differences between the velocity control loops of individual wheels for instance variations in friction, motor constants or amplifier gains. The coupling error leads to an additional wheel slippage, which decreases the accuracy of odometry.

The blue curve in Fig 7 shows the coupling error ϵ_1 with active coupling control ($k_c = 2$). Compared to the red curve, the coupling error is reduced nearly to zero.

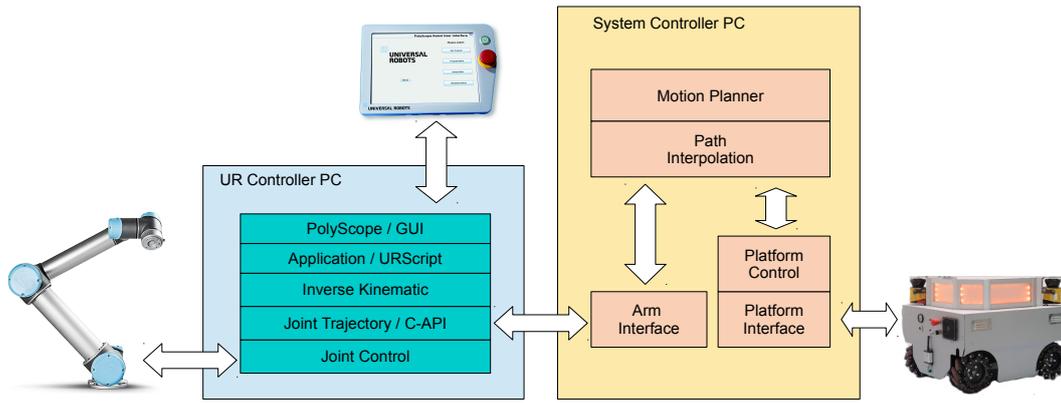


Figure 6. Control architecture of Mecanum wheeled mobile manipulator

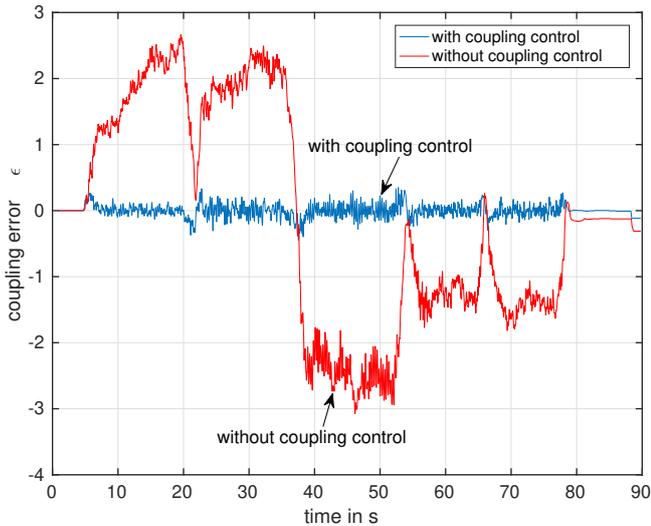


Figure 7. Coupling error over time

D. Dynamic Model for Mobile Platforms with Mecanum Wheels

This section will briefly describe a dynamic model of Mecanum wheeled platforms including the friction force of the Mecanum wheels. The dynamics of a Mecanum wheeled platform can be described in the robot frame:

$$\mathbf{F}_R = \mathbf{H} \ddot{\mathbf{x}}_R = \mathbf{J}^{-1} \mathbf{F}_{\text{wheel}} \quad (28)$$

with

$$\mathbf{F}_R = \begin{pmatrix} F_x \\ F_y \\ M_\theta \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{pmatrix}, \quad \mathbf{F}_{\text{wheel}} = \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{pmatrix},$$

where \mathbf{F}_R are the translational forces and the moment of force (torque) in robot frame, which accelerate the platform, m is the mass of the platform for translation, I is the moment of inertia for platform rotation around the z axis and $\mathbf{F}_{\text{wheel}}$ are the tangential contact forces of the Mecanum wheels, which accelerate the platform. The acceleration in robot frame can be obtained from accelerations in world frame by differentiating (3), which leads to

$$\ddot{\mathbf{x}}_R = \dot{\mathbf{R}}(\theta) \dot{\mathbf{x}}_W + \mathbf{R}(\theta) \ddot{\mathbf{x}}_W \quad (29)$$

and considers the effect of the *Coriolis* force. Each wheel i is driven by an electrical motor, which produces the moment

$$M_i = r F_i + M_{\text{fric},i} + I_i \ddot{\varphi}_i \quad (30)$$

where r is the wheel radius, M_{fric} is the total of the moment produced by friction, I_i is the moment of inertia of the wheel and $\ddot{\varphi}_i$ is the angular acceleration of the wheel: $\ddot{\varphi} = \mathbf{J} \ddot{\mathbf{x}}_R$. The moment of motor i is produced by the armature current of the motor: $M_i = c_\phi I_{a,i}$, where c_ϕ is the motor constant. The dynamics of the motor current can be described as

$$L_a \frac{dI_{a,i}}{dt} + R_a I_{a,i} = U_{a,i} - U_{\text{emf},i}, \quad (31)$$

with $U_{\text{emf},i} = c_\phi n_{\text{gear}} \dot{\varphi}_i$

where L_a is the armature inductance, R_a is the armature resistance, U_a is the armature voltage, U_{emf} is the voltage produced by the back electro mechanical force, n_{gear} is the ratio of the gear that connects motor with wheel.

The moment produced by friction M_{fric} is a sum of the internal friction produced by the motor and the gear, the rolling friction on the floor and the friction in the bearing of the rollers and therefore M_{fric} depends largely on the direction of movement. We model the moment produced by friction as a function of the platform movement

$$M_{\text{fric},i} = f_i(\dot{\mathbf{x}}_R), \quad (32)$$

which can be experimentally identified [16].

E. Forward Kinematics of the Mobile Manipulator

The development of the forward kinematics is straight forward. The position of the platform with respect to the world frame can be described in 3D by the homogeneous transformation matrix

$${}^w\mathbf{T}_p(\mathbf{q}_p) = \begin{pmatrix} \cos \theta_p & -\sin \theta_p & 0 & x_p \\ \sin \theta_p & \cos \theta_p & 0 & y_p \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (33)$$

where θ_p is the heading of the platform with respect to the world frame and (x_p, y_p) is the position of the platform in the world frame.

The robotic arm is mounted on the platform in the fixed position (d_x, d_y, d_z) with respect to the robot frame, which

yields to the constant homogeneous transformation matrix

$${}^pT_0 = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (34)$$

The forward kinematics of the robot arm are described by the DH-matrix

$${}^0T_6(\mathbf{q}_a) = \prod_{i=1}^6 {}^{i-1}T_i, \quad (35)$$

this yields directly to the forward kinematics $\mathbf{x} = \mathbf{f}(\mathbf{q})$ of the whole mobile manipulator, which can be obtained from

$${}^wT_6(\mathbf{q}_p, \mathbf{q}_a) = {}^wT_p(\mathbf{q}_p) \cdot {}^pT_0 \cdot {}^0T_6(\mathbf{q}_a). \quad (36)$$

F. Inverse Kinematics of the Mobile Manipulator

We solve the inverse kinematics of the mobile manipulator

$$\mathbf{q} = \mathbf{f}^{-1}(\mathbf{x})$$

using a numerical approach. This approach needs the Jacobian matrix of the forward kinematics of the mobile manipulator:

$$\mathbf{J}(\mathbf{q}) = \frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}} = \begin{pmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \dots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_6}{\partial q_1} & \frac{\partial f_6}{\partial q_2} & \dots & \frac{\partial f_6}{\partial q_n} \end{pmatrix}$$

This Jacobian consists of the Jacobian of the mobile platform and the manipulator

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q}) \cdot \dot{\mathbf{q}}, \text{ with } \mathbf{J}(\mathbf{q}) = (\mathbf{J}_p(\mathbf{q}), \mathbf{J}_a(\mathbf{q})). \quad (37)$$

The $\mathbf{J}_a(\mathbf{q})$ can be obtained from (10), (11), (12). The 3D kinematics of the platform can be obtained from (33) and (34)

$$\mathbf{x} = \mathbf{f}_p(\mathbf{q}_p) = \begin{pmatrix} x_p + d_x \cos \theta_p - d_y \sin \theta_p \\ y_p + d_x \sin \theta_p + d_y \cos \theta_p \\ d_z \\ 0 \\ 0 \\ \theta_p \end{pmatrix},$$

$$\text{with } \mathbf{x} = [x, y, z, \alpha, \beta, \gamma]^T, \mathbf{q}_p = [x_p, y_p, \theta_p]^T, \quad (38)$$

where (d_x, d_y, d_z) denotes the position of the arm on the platform. Thereby, the Jacobian is easily obtained:

$$\mathbf{J}_p(\mathbf{q}_p) = \frac{\partial \mathbf{f}_p(\mathbf{q}_p)}{\partial \mathbf{q}_p} = \begin{pmatrix} 1 & 0 & -d_x \sin \theta_p - d_y \cos \theta_p \\ 0 & 1 & d_x \cos \theta_p - d_y \sin \theta_p \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (39)$$

In order to solve the inverse kinematics, a rough estimation \mathbf{q}_k , near the final solution $\mathbf{q} = \mathbf{f}^{-1}(\mathbf{x})$ is needed. If the mobile manipulator moves a continues path, this can be the preceding point on the path. Starting with \mathbf{q}_k , joint coordinates nearer to the final solution can be obtained by

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{J}^{-1}(\mathbf{q}_k) \cdot (\mathbf{x} - \mathbf{x}_k), \text{ with } \mathbf{x}_k = \mathbf{f}(\mathbf{q}_k)$$

The iteration is aborted, when the error falls below a predefined limit:

$$|\mathbf{x} - \mathbf{x}_k| < \epsilon$$

IV. CONTROLLER DESIGN

As aforementioned, our mobile manipulator consists of a UR5 robotic arm and a Mecanum wheeled mobile platform. It is controlled by a control system, which is distributed on 2 control PCs. Fig. 6 shows the control architecture of the whole system. Details of the different controllers are described in the next sections.

A. Control of the UR5 Manipulator

The UR5 can be controlled at three different levels: The Graphical User Interface (GUI) Level, the Script Level and the C Application Programming Interface (C-API) Level [17]. *PolyScope* is the graphical user interface (GUI) for operating the robotic arm and for creating and executing robot programs. *URScript* is the robot programming language used to control the robot at the Script Level. This programs can be saved directly on the robot controller or commands can be sent via TCP/IP socket to the robot.

User programs that uses the C-API are executed on the UR controller and interact directly at the joint level with a cycle time of 8 ms. At this level, the UR controller can be supplied by either joint velocities or a combination of joint positions, joint velocities and joint accelerations. In our mobile manipulator, we use the C-API and control the UR5 at joint level. At this level, the synchronization of the platform movements with the arm movements is feasible in real-time. This enables trajectory planning for the whole system in Cartesian space and controlling the movements of platform and arm synchronously at joint level, while avoiding obstacles in real time, without leaving the trajectory.

The structure of the controllers with its interfaces is shown in Fig. 6. The robotic arm is controlled by the UR controller PC at joint level. The joint positions of the arm are generated by the path interpolation module in the system controller PC and streamed every 8 ms over a TCP/IP socket and Ethernet to the UR controller PC. The UR controller PC controls the joints and transfers the control signals over an interface module to the motors.

B. Control of the Mecanum Wheeled Mobile Platform

The controller of the Mecanum wheeled mobile platform is integrated in the system controller PC (platform control in Fig. 6). The internal structure of this controller is shown in Fig. 8. The path interpolation generates a stream of poses $\mathbf{x}_p = (x_p, y_p, \theta_p)^T$ and its derivatives $\dot{\mathbf{x}}_p, \ddot{\mathbf{x}}_p$ as input for the controller. The structure of the controller is derived from a classical cascade structure. It consists of a position controller, which controls the pose of the platform in world frame, a wheel controller, which controls the velocity of the wheels and a coupling controller, which controls the kinematic motion constraints.

The position controller obtains the actual pose of the platform by odometry and generates the velocities of the platform in the world frame plus feed forwarding the velocities from the path interpolation. Details of the odometry for Mecanum wheeled platforms can be found in [18]. These velocities are transformed into the robot frame and then into reference velocities for the wheel controller. These reference velocities are calculated using the inverse kinematics of the platform (17) and therefore meet the kinematic constraints

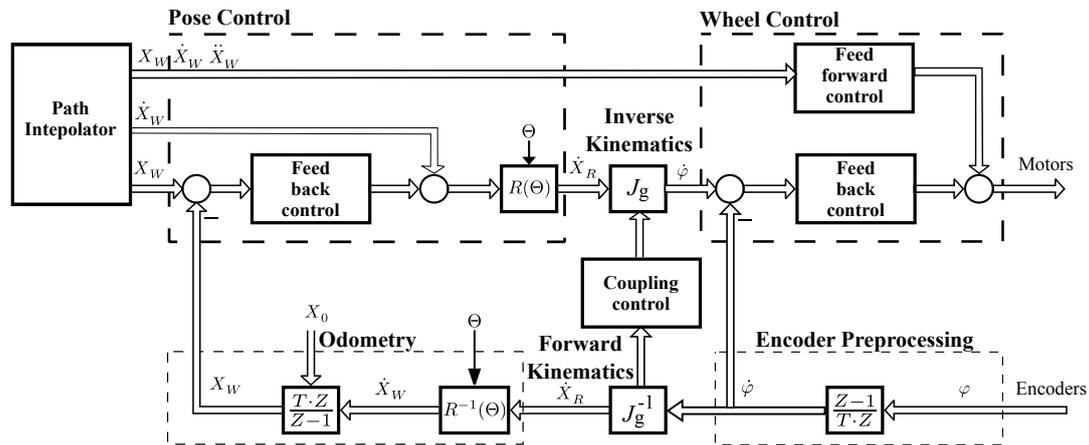


Figure 8. Controller structure of Mecanum wheeled platform

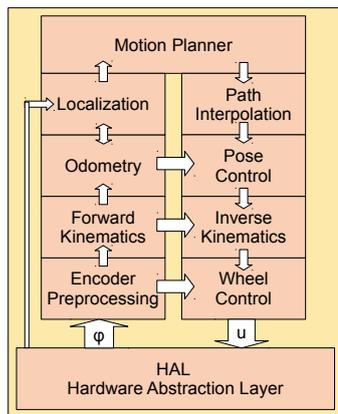


Figure 9. Software architecture of the navigation stack for controlling the mobile platform

at any time. Owing to parameter differences in the control loops of the wheels or unbalanced loads, the actual wheel velocities may violate the kinematic motion constraints. These violations lead to additional wheel slippage.

Aim of the coupling controller is to change the reference velocities of the wheel controllers in such a way that the actual velocities meet the motion constraints [19]. The wheel controller controls the velocity of the wheels by feed back control using the wheel encoders plus optional calculated torque feed forward control.

Fig. 9 shows the software architecture of the navigation stack. The bottom of the navigation stack builds a hardware abstraction layer (HAL), which hides technical details of the communications with the robot [20]. The navigation stack contains 3 control loops: The pose control loop, the coupling controller at the level of wheel kinematics and the underlying wheel control loop. The localization layer obtains sensor data from HAL and estimates the pose of the platform using this data and odometry [21], [18]. The top of the stack builds the motion planner for the platform.

V. CONCLUSIONS

In this paper, we have developed a compact and easy applicable kinematic model of an omnidirectional mobile manipulator. The mobile manipulator consists of a manipulator with 6 DoF and an over-actuated Mecanum wheeled mobile platform, which provides additional 3 DoF. The kinematic

model includes the kinematic motion constraints of the mobile platform as well as the over-determined motion model of the whole mobile manipulator.

The kinematic model of the omnidirectional mobile manipulator is described by 9 DoF in the configuration space, which provide 6 DoF in Cartesian space. Therefore, the inverse kinematics are solved with a numerical approach using a Jacobi matrix. The proposed kinematic model is valid for all serial manipulators with 6 DoF, mounted on an omnidirectional mobile platform driven by $n \geq 4$ Mecanum wheels.

REFERENCES

- [1] C. Röhrig and A. Jochheim, "The virtual lab for controlling real experiments via Internet," in *Proceedings of the 11th IEEE International Symposium on Computer-Aided Control System Design*, Hawaii, USA, Aug. 1999, pp. 279–284.
- [2] R. Bischoff, U. Huggenberger, and E. Prassler, "KUKA youbot - a mobile manipulator for research and education," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 1–4.
- [3] C.-H. King, T. L. Chen, A. Jain, and C. C. Kemp, "Towards an assistive robot that autonomously performs bed baths for patient hygiene," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, IEEE, IEEE, 2010, pp. 319–324.
- [4] Y. Jiang, M. Lim, C. Zheng, and A. Saxena, "Learning to place new objects in a scene," *The International Journal of Robotics Research*, vol. 31, no. 9, pp. 1021–1043, 2012.
- [5] E. Dean-Leon, B. Pierce, F. Bergner, P. Mittendorfer, K. Ramirez-Amaro, W. Burger, and G. Cheng, "TOMM: Tactile omnidirectional mobile manipulator," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2441–2447.
- [6] M. Hvilshøj, S. Bøgh, O. Skov Nielsen, and O. Madsen, "Autonomous industrial mobile manipulation (AIMM): past, present and future," *Industrial Robot: An International Journal*, vol. 39, no. 2, pp. 120–135, 2012.
- [7] C. Sprunk, B. Lau, P. Pfaff, and W. Burgard, "An accurate and efficient navigation system for omnidirectional robots in industrial environments," *Autonomous Robots*, vol. 41, no. 2, pp. 473–493, 2017.
- [8] D. Wahrmann, A.-C. Hildebrandt, C. Schuetz, R. Wittmann, and D. Rixen, "An autonomous and flexible robotic framework for logistics applications," *Journal of Intelligent & Robotic Systems*, pp. 1–13, 2017.
- [9] C. Wurl, T. Fritz, Y. Hermann, and D. Hollnabacher, "Production logistics with mobile robots," in *ISR 2018; 50th International Symposium on Robotics*, VDE. VDE, 2018, pp. 213–218.
- [10] D. Pavlichenko, G. M. García, S. Koo, and S. Behnke, "KittingBot: A mobile manipulation robot for collaborative kitting in automotive logistics," *arXiv preprint arXiv:1809.05380*, 2018.
- [11] P. F. Muir and C. P. Neuman, "Kinematic modeling for feedback control of an omnidirectional wheeled mobile robot," in *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4, IEEE. IEEE, 1987, pp. 1772–1778.

- [12] C. Röhrig and D. Heß, "OmniMan: An omnidirectional mobile manipulator for human-robot collaboration," in *Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2019*, Hong Kong, 13-15 March 2019, pp. 55–60.
- [13] G. Campion, G. Bastin, and B. Dandrea-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.
- [14] *UR5 Technical Specifications*, Universal Robots A/S, Odense, Denmark, 2015, http://www.universal-robots.com/media/50588/ur5_en.pdf.
- [15] K. P. Hawkins, "Analytic inverse kinematics for the universal robots UR-5/UR-10 arms," Georgia Institute of Technology, Tech. Rep., 2013, https://smartech.gatech.edu/bitstream/handle/1853/50782/ur_kin_tech_report_1.pdf.
- [16] F. Künemund, D. Heß, and C. Röhrig, "Energy efficient kinodynamic motion planning for holonomic AGVs in industrial applications using state lattices," in *Proceedings of the 47th International Symposium on Robotics (ISR 2016)*, Munich, Germany, Jun. 2016, pp. 459–466.
- [17] *The URScript Programming Language, Version 3.1*, Universal Robots A/S, Odense, Denmark, Jan. 2015.
- [18] C. Röhrig, Heß, and F. Künemund, "RFID-based localization of mobile robots using the received signal strength indicator of detected tags," *Engineering Letters*, vol. 24, no. 3, pp. 338–346, Aug. 2016.
- [19] C. Röhrig, D. Heß, and F. Künemund, "Motion controller design for a mecanum wheeled mobile manipulator," in *Proceedings of the IEEE Conference on Control Technology and Applications (CCTA 2017)*, Kohala Coast, Hawai'i, USA, Aug. 2017, pp. 444–449.
- [20] D. Heß, F. Künemund, and C. Röhrig, "Linux based control framework for mecanum based omnidirectional automated guided vehicles," in *Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2013, WCECS 2013*, San Francisco, USA, 23-25 October, 2013, pp. 395–400.
- [21] C. Röhrig, C. Kirsch, J. Lategahn, M. Müller, and L. Telle, "Localization of Autonomous Mobile Robots in a Cellular Transport System," *Engineering Letters*, vol. 20, no. 2, pp. 148–158, Jun. 2012.



Christof Röhrig received his Diploma degree from the University of Bochum, Germany, in 1993, and his Doctor degree from the University of Hagen, Germany, in 2003, both in electrical engineering. Between 1993 and 1997 he was head of Automated Systems Engineering at Reinoldus Transport- und Robotertechnik GmbH Dortmund, Germany. From 1997 until 2003 he was with the Control Systems Engineering Group at University of Hagen. Since 2003, he is Professor of Computer Science at the University of Applied Sciences and Arts in Dortmund, Germany, where he heads the Intelligent Mobile Systems Lab (IMSL). The IMSL develops intelligent algorithms for mobile systems, also third party funded projects on robotics, assistance systems for demographic change and real time locating systems are continually taken care of. Christof Röhrig is a founder member of the research focus 'Mobile Business - Mobile Systems (MBMS)' and 'BioMedizinTechnik (BMT)' of the University of Applied Science and Arts Dortmund. He is also a founder member and on the board of the 'Institute for the Digital Transformation of Application and Living Domains' (IDiAL). Christof Röhrig is (co-)author of more than 100 national and international peer-reviewed publications. In his field, he continuously reviews papers and is member of program committees. His current research interests include mobile robots and Real Time Locating Systems.



Daniel Heß received his Diploma degree in 2008, and the Masters degree in 2010, both in computer science from the University of Applied Sciences and Arts in Dortmund, Germany. Since 2008, he is working in different research projects at the Intelligent Mobile Systems Lab, University of Applied Sciences and Arts Dortmund, where he is currently working toward the Doctor degree. His current research interests include sensor data fusion and localization.