High-Precision Contour Tracking for Mobile Manipulators in Large-Scale Industrial Applications

Buu Hai Dang Trinh[a], Daniel Heß[b] and Christof Röhrig[c]
IDiAL – Institute for the Digital Transformation of Application and Living Domains, Dortmund University of Applied Sciences and Arts, Dortmund, Germany

Keywords: Mobile Manipulator, Artificial Landmarks, Contour Tracking, Trajectory Control, High-Precision Localization.

Abstract: Industrial manipulators are limited in their workspace due to mechanical constraints, which pose significant challenges in large-scale industrial applications. Expanding a robot's workspace often involves deploying additional stationary manipulators or integrating linear axes, both of which increase installation costs and system complexity without gaining much flexibility. A more effective and flexible solution is to integrate industrial manipulators onto mobile platforms. To support this research, the authors developed a mobile manipulator system consisting of a mobile platform driven by two Differential Drive Steering Units and an industrial robotic arm with six Degrees of Freedom (DoF). This configuration provides the system with nine DoF in its configuration space, substantially extending the workspace compared to conventional fixed-base manipulators. A trajectory control method is proposed to ensure smooth, low-vibration, and high-precision motion during operation. To enable accurate localization, a cost-effective method based on a 2D laser sensor and artificial landmarks is introduced. Furthermore, a high-precision contour tracking algorithm is developed to monitor the position of the end-effector relative to the workpiece. The proposed methods are validated through real-world experiments, demonstrating millimeter-level accuracy in both positioning and tracking.

1 INTRODUCTION

Manipulators are widely used in manufacturing due to their exceptional flexibility, precision, and efficiency. Owing to their ability to seamlessly integrate with a wide range of end-effectors, manipulators can be programmed to perform diverse tasks, including heavy, hazardous, and dangerous operations that pose potential risks to human workers, thereby enhancing workplace safety and reducing the likelihood of occupational accidents. Furthermore, they are capable of maintaining high levels of precision and repeatability, ensuring consistent product quality and reliability in production lines with stringent tolerance requirements. With the flexibility to be integrated into automated production systems, combined with sensors and advanced technologies, manipulators play a pivotal role in the advancement of smart manufacturing lines, thus promoting comprehensive industrial automation in modern industries.

Nevertheless, these manipulators are inherently limited in their workspace as a result of mechanical constraints such as joint configurations, structural dimensions, and arm length. In applications involving the processing of large-scale components — such as those encountered in shipbuilding, aerospace, and wind turbine manufacturing — the required workspace often exceeds the reach and flexibility of stationary manipulators.

Figure 1: MobileRobot: Omnidirectional mobile platform with a 6-DoF manipulator.

A common approach to overcoming workspace limitations is the strategic deployment of additional sta-

[a] https://orcid.org/0009-0008-3661-3569
[b] https://orcid.org/0000-0002-4627-8951
[c] https://orcid.org/0000-0002-3286-3703

Trinh, B. H. D., Heß, D. and Röhrig, C.
High-Precision Contour Tracking for Mobile Manipulators in Large-Scale Industrial Applications.
DOI: 10.5220/0013709500003982

In *Proceedings of the 22nd International Conference on Informatics in Control, Automation and Robotics (ICINCO 2025) - Volume 2*, pages 60-71
ISBN: 978-989-758-770-2; ISSN: 2184-2809

tionary manipulators throughout the work area. However, this approach considerably increases installation costs and system complexity, particularly with respect to coordination and control. Another widely used solution is the integration of linear axes — commonly referred to as the *seventh axis* — onto which a single manipulator is mounted and can traverse along a predefined track. For example, systems like the KUKA KL 1500 extend the working range of a robot without requiring multiple manipulators (KUKA Robotics, 2025). However, this configuration only allows movement along one axis and is not flexible when the workspace or task changes. Adjusting the system usually requires extra mechanical work and reprogramming. Mobile manipulators, which combine industrial manipulators with autonomous mobile platforms, are increasingly recognized as a promising solution to overcome the inherent workspace limitations (Ghodsian et al., 2023). Nevertheless, the development of such systems poses significant challenges, as it requires maintaining high positioning accuracy and operational efficiency while addressing the uncertainties introduced by both the mobile base and the manipulator (Sereinig et al., 2020).

Industrial mobile manipulators typically operate in highly structured environments where object positions are fixed and predetermined. This setup enables more predictable, stable, and efficient navigation and operation. Unlike those used in service or domestic applications, they must meet strict accuracy requirements — particularly at the end-effector — for tasks such as gripping, welding, painting, and assembly (Gawel et al., 2019). Since the end-effector's pose results from the combined kinematics of the mobile base and the manipulator, any inaccuracy in the platform's movement can compromise overall task performance.

As part of this research, a mobile manipulator system was designed combining a mobile platform driven by two Differential Drive Steering Units (DDSUs) and a 6-DoF industrial robotic arm (see Fig. 1). This configuration enables the platform to achieve omnidirectional movement while retaining the advantages of conventional wheels. Compared to Mecanum wheels, the DDSU-based platform significantly reduces vibrations, thereby enhancing the system's accuracy and stability. We performed a measurement of the displacement of the Tool Center Point (TCP) position caused by vibrations generated by the platform during the linear phase at 800 mm/s, as illustrated in Fig. 2. The results demonstrate that the DDSU-based platform generates significantly lower vibrations than the Mecanum-wheeled platform, leading to improved end-effector stability and positioning accuracy. Furthermore, we have presented a system control model to address the kinematic constraints of DDSUs. While these units enable omnidirectional movement, they introduce specific limitations that require careful control to maintain accurate and efficient system performance (Heß et al., 2023).

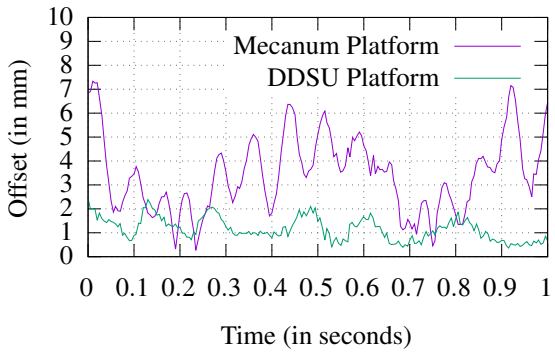


Figure 2: Displacement of base and TCP positions during linear motion.

The localization of mobile manipulators within the workspace is essential for performing tasks accurately. However, advanced localization systems based on 3D Light Detection and Ranging (LiDAR) sensor or motion capture systems are often prohibitively expensive and may not be feasible for many industrial environments. Radio-based localization technologies, such as Ultra-Wideband (UWB), offer high positioning accuracy and low latency, making them an attractive alternative. Nevertheless, UWB still encounters significant challenges in complex industrial environments due to signal interference and multipath propagation. The performance of UWB can degrade considerably in enclosed areas with metallic structures or dense machinery, where signal reflections and obstructions are prevalent (Delamare et al., 2020).

The main contributions of this paper are as follows:

1. A mobile manipulator system, integrating an industrial six-DoF robotic arm with a mobile platform driven by two DDSUs, has been developed to significantly expand the workspace for large-scale industrial applications.
2. A trajectory control method for the mobile manipulator has been proposed that ensures smooth, low-vibration coordinated motion while fully considering the kinematic and mechanical constraints of the DDSU-driven platform.
3. A cost-effective localization method based on a 2D laser sensor and artificial landmarks has been introduced.
4. A contour tracking algorithm has been devel-

oped to achieve high-precision control of the end-effector position relative to the workpiece during the entire task execution.

5. The proposed methods have been validated in real-world scenarios, demonstrating that the mobile manipulator system achieves millimeter-level positioning and tracking accuracy.

2 RELATED WORK

A variety of mobile manipulator systems have been developed to overcome the workspace limitations of stationary robots. Representative systems include KUKA's KMR iiwa and KMR QUANTEC, both of which integrate omnidirectional platforms with KUKA industrial arms. Another example is RB-KAIROS+, a mobile manipulator from Robotnik Automation that combines an omnidirectional base with the UR5e, UR10e, or UR16e collaborative robotic arms. These systems are designed for flexible industrial and intralogistics applications. Additionally, researchers at the Dortmund University of Applied Sciences and Arts developed OmniMan, a mobile manipulator system for human-robot collaboration in intralogistics (Röhrig and Heß, 2020).

All of these systems use Mecanum-wheeled platforms, which allow for omnidirectional movement in narrow spaces. Mecanum wheels consist of driven hubs combined with free-spinning rollers mounted at an angle to the wheel axis, allowing the platform to move in any direction. However, these rollers generate vibrations during motion, which can reduce the system's stability and accuracy and negatively affect production quality (Bae and Kang, 2016).

Zhewen et al. proposed combining Mecanum wheels with a dedicated suspension system to minimize vibrations and improve comfort (Zhewen et al., 2024). In addition to Mecanum wheel-based designs, omnidirectional platforms can be implemented using steered standard wheels. An overview of such platforms is provided (Jacobs, 2018). Wheel modules enabling omnidirectional motion can be designed either with active steering motors (Jacobs et al., 2012) or using differential drive mechanisms (Jacobs and Schaefer, 2020).

3 PROBLEM FORMULATION

In this paper, the problems of motion control, localization, and contour tracking for a large-scale industrial mobile manipulator system, designed specifically for automated welding of large objects are addressed. The system consists of a six DoF robotic arm mounted on an omnidirectional mobile platform driven by two DDSUs. The robotic arm enables full six DoF spatial manipulation, allowing precise control of position and orientation in three-dimensional space. In addition, the omnidirectional platform provides planar mobility, enabling two-dimensional translation and rotation. Consequently, the entire mobile manipulator system exhibits a total of nine DoF in its configuration space. Therefore, the mobile welding manipulator (MWM) is an overdetermined system where the additional DoF in configuration space can be used to optimize the motion of the MWM. The welding path is specified in the three-dimensional Cartesian space of the MWM's operational workspace. To maintain weld quality, the welding torch is required to track this path at predetermined linear and angular velocities. Since the welding path may extend beyond the reachable workspace of the robotic arm, it is necessary for the arm and the mobile platform to coordinate their movements synchronously. The forward kinematics of a mobile manipulator can be generally formulated as:

$$\boldsymbol{p} = \boldsymbol{f}(\boldsymbol{q}), \quad \text{with } \boldsymbol{p} = \begin{pmatrix} x \\ y \\ z \\ \alpha \\ \beta \\ \gamma \end{pmatrix}, \quad \text{and } \boldsymbol{q} = \begin{pmatrix} \boldsymbol{q}_\mathrm{p} \\ \boldsymbol{q}_\mathrm{a} \end{pmatrix}. \quad (1)$$

Where $\boldsymbol{p}$ denotes the pose of the mobile manipulator's tool frame with respect to the world frame, such that $\boldsymbol{p} \in \mathbb{R}^3 \times \mathrm{SO}(3)$. The vector $\mathbf{q}$ represents the generalized coordinates in the configuration space (C-space), consisting of the 2D pose of the mobile platform, defined as $\boldsymbol{q}_\mathrm{p} = (x_\mathrm{p}, y_\mathrm{p}, \theta_\mathrm{p})^\mathrm{T}$, where $\boldsymbol{q}_\mathrm{p} \in \mathbb{R}^2 \times \mathbb{S}^1$, and the joint angles of the manipulator arm, given by $\boldsymbol{q}_\mathrm{a} = (\theta_1, \ldots, \theta_6)^\mathrm{T}$. The inverse kinematics, crucial for trajectory planning and control, can thus be expressed as:

$$\boldsymbol{q} = \boldsymbol{f}^{-1}(\boldsymbol{p}) \quad (2)$$

The inverse kinematics problem of the system is overdetermined because it has more degrees of freedom than required to define the end-effector pose. This leads to multiple possible joint configurations for the same task, which can be resolved using numerical methods to optimize motion in the configuration space. In addition, the system's redundancy not only improves motion efficiency and safety but also extends the manipulator's reachable workspace beyond the physical limitations of the arm alone.

The omnidirectional mobile platform, driven by two DDSUs, can simultaneously translate along the x- and y-axes and rotate about the z-axis, thereby providing full mobility in the 2D plane. The world frame

F_W is a fixed reference frame used to describe the task space. The MWM operates within this workspace, and its motion is expressed relative to F_W. The robot frame F_R is a coordinate frame that is rigidly attached to the mobile platform, with its origin located at the geometric center of the platform (see Fig. 3).

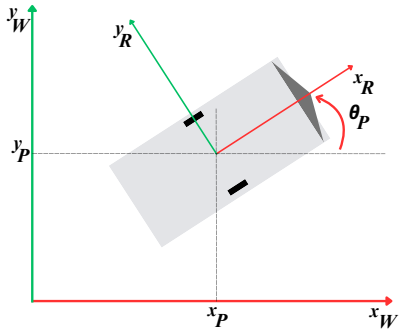


Figure 3: Pose definition of the mobile manipulator in the world $\boldsymbol{p}_\mathrm{p}$ and robot frame F_R.

The pose of the mobile platform relative to the world frame is defined as $\boldsymbol{p}_\mathrm{p} = (x_\mathrm{p}, y_\mathrm{p}, \theta_\mathrm{p})^\mathrm{T}$, where x_p and y_p represent the position in the 2D plane, and θ_p is the heading angle around the orthogonal z-axis. This coordinate setup allows for a clear separation between global task planning in the world frame and local control actions in the robot frame. The platform's pose $\boldsymbol{p}_\mathrm{p}$ serves as a critical reference for defining its velocity, localization, and control inputs within the world frame. However, for control implementation, these quantities are generally transformed and expressed in the robot frame F_R, where they are more convenient and consistent to manage in motion control schemes. The transformation of velocities from the robot frame to the world frame is performed using the platform's heading angle ($\theta = \theta_\mathrm{p}$):

$$\begin{aligned} \dot{\boldsymbol{p}}_\mathrm{R} = \boldsymbol{R}(z,\theta)\,\dot{\boldsymbol{p}}_\mathrm{W}, \quad \Rightarrow \quad \dot{\boldsymbol{p}}_\mathrm{W} = \boldsymbol{R}^{-1}(z,\theta)\,\dot{\boldsymbol{p}}_\mathrm{R}, \\ \text{with } \dot{\boldsymbol{p}}_\mathrm{W} = \begin{pmatrix} \dot{x}_\mathrm{W} \\ \dot{y}_\mathrm{W} \\ \dot{\theta} \end{pmatrix}, \; \dot{\boldsymbol{p}}_\mathrm{R} = \begin{pmatrix} \dot{x}_\mathrm{R} \\ \dot{y}_\mathrm{R} \\ \dot{\theta} \end{pmatrix}, \\ \text{and } \boldsymbol{R}(z,\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \tag{3}$$

To determine the necessary wheel velocities and steering angles that enable the platform to achieve a desired motion in the robot frame, the inverse kinematics of the platform is formulated as a set of nonlinear equations:

$$\begin{pmatrix} \dot{\boldsymbol{\varphi}} \\ \dot{\boldsymbol{\beta}} \end{pmatrix} = \boldsymbol{f}_\mathrm{p}(\dot{\boldsymbol{p}}_\mathrm{R}), \tag{4}$$

where

- $\dot{\boldsymbol{p}}_\mathrm{R} = (\dot{x}_\mathrm{R}, \dot{y}_\mathrm{R}, \dot{\theta})^\mathrm{T}$ is the velocity of the platform over the ground described in the robot frame.
- $\dot{\boldsymbol{\varphi}} = (\dot{\varphi}_1, \dot{\varphi}_2, \ldots, \dot{\varphi}_n)^\mathrm{T}$ is the vector of wheel angular velocities.
- $\dot{\boldsymbol{\beta}} = (\dot{\beta}_1, \dot{\beta}_2, \ldots, \dot{\beta}_m)^\mathrm{T}$ is the vector of wheel steering angles.

The forward kinematics can be obtained by using a nonlinear least-squares approach

$$\dot{\boldsymbol{p}}_\mathrm{R} = \boldsymbol{f}_\mathrm{p}^{-1} \begin{pmatrix} \dot{\boldsymbol{\varphi}} \\ \dot{\boldsymbol{\beta}} \end{pmatrix}, \tag{5}$$

In contrast to the overdeterminacy in the mobile manipulator, the over-actuated drive structure of the platform cannot be used to achieve additional degrees of freedom in C-space. The overdeterminacy of the platform is subject to motion constraints, which must be satisfied to avoid additional wheel slip.

The global localization of the MWM is performed using a 2D LiDAR sensor combined with strategically placed artificial landmarks around the workspace of the MWM. A localization algorithm processes the LiDAR and landmark data to estimate the pose of the platform relative to the world coordinate frame $\boldsymbol{p}_\mathrm{p}$. The localization algorithm processes the LiDAR measurements and the known locations of these landmarks to estimate the platform's pose relative to the world coordinate frame F_W.

Although the localization system achieves an accuracy of a few centimeters, this level of precision remains insufficient to meet the stringent requirements of industrial applications. Furthermore, the global localization system is affected by environmental interference, landmark occlusions, and the limited scanning range of the LiDAR, leading to reduced stability during operation. For the welding task, high-precision contour tracking of the target workpiece is required to ensure accurate torch positioning along the welding path. Since the global localization system alone cannot guarantee the required accuracy at the end-effector, a 2D profile scanner is mounted at the end-effector to support local contour tracking. This sensor does not directly provide contour information, but instead returns profile data as a set of distance measurements captured along the scanning line. Therefore, additional data processing is necessary to reconstruct and accurately estimate the object's contour.

By combining global localization with contour tracking, the system is capable of performing high-precision manipulation tasks with enhanced accuracy and operational stability.

4 PROPOSED LOCALIZATION METHOD

4.1 Overview and Requirements

Accurate localization of the mobile platform is a fundamental prerequisite for performing high-precision manipulation tasks in large-scale industrial applications. In structured and static environments, Adaptive Monte Carlo Localization (AMCL) offers an efficient and cost-effective solution for mobile platform localization. However, its accuracy remains limited, with position errors often reaching several tens of centimeters. Although the integration of QR codes on the floor improves localization accuracy, it still fails to meet the stringent requirements of industrial applications. This limitation becomes even more critical in large and dynamic environments, where noise, moving obstacles, and changes in the environment further degrade accuracy (Wang et al., 2021).

For contour tracking applications such as welding, the system requires centimeter-level localization accuracy of the mobile base to ensure stable and precise end-effector control. Improved base accuracy minimizes the need for manipulator compensation, thereby enhancing the stability and quality of the weld path.

4.2 Localization Architecture

In this project, the NAV245 2D LiDAR sensor (SICK AG, Germany) is employed for environmental perception and distance measurement. The sensor covers a 270° scanning field with an angular resolution of 0.001° for reflectors and 0.25° for raw contour data. Operating at 25 Hz, the sensor completes a full scan in 40 ms and supports a maximum detection range of 50 m, depending on the target's properties. According to the manufacturer, for reflector-based data, the typical systematic measurement error is ±10 mm, while the statistical error is 8 mm (SICK AG, 2022). These features enable precise and low-latency positioning of the robot in large indoor environments using only a single sensor.

Artificial landmarks are constructed using RA3-class microprismatic retroreflective foil, which enables incident laser beams to be returned directly to the sensor regardless of the angle of incidence. This full-cube microprism technology achieves remission values of up to 3,000%, which is considerably higher than the maximum 100% remission of natural surfaces. As a result, measurements on reflectors provide higher precision and signal stability, making them ideal for landmark-based localization in indoor environments.

The odometry system is derived from the DDSU developed by Götting KG. Each DDSU integrates a pseudo-incremental encoder providing 65,535 pulses per revolution for wheel rotation measurement of the foure active wheels of both DDSUs, and a steering angle encoder with 4,096 pulses per revolution for accurate steering position feedback. These high-resolution sensors provide precise relative motion estimation of the mobile platform. However, while the high-resolution encoders ensure low-noise and high-frequency odometry data, the odometry data tends to accumulate errors over time due to wheel slip and integration drift. Fig. 4 illustrates the overall pipeline of the proposed high-precision and robust localization system.

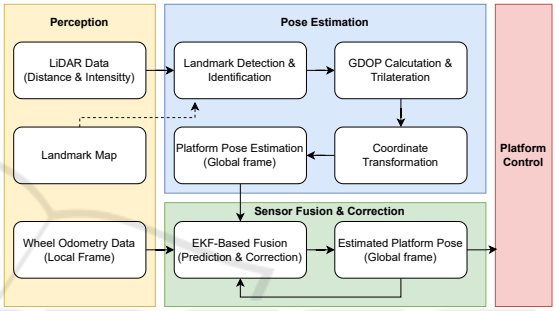


Figure 4: Overview of the proposed localization architecture.

The *Perception* module is responsible for continuously collecting distance and intensity information from a 2D LiDAR scanner as well as odometry data from wheel encoders. The *Pose Estimation* module utilizes LiDAR data in combination with a previously generated landmark map containing known global positions of all landmarks in the workspace. This enables the system to accurately detect and identify the landmarks present in the current scan. Once the landmarks are identified, the module calculates the Geometric Dilution of Precision (GDOP) to evaluate how the spatial configuration of the landmarks relative to the sensor affects the accuracy of the position estimation. The GDOP metric guides the selection of the most suitable landmark-set for trilateration, ensuring improved estimation reliability. The trilateration method is used to estimate the global position of the scanner. Finally, by applying a coordinate transformation, the global pose of the mobile platform is determined.

Trilateration can achieve high positioning accuracy when landmarks are correctly identified and the distance measurements are sufficiently reliable. However, in practice, these measurements are often affected by noise and environmental conditions. Due to its sensitivity, even small errors can result in significant deviations in position estimation. To improve localiza-

tion accuracy and robustness, an Extended Kalman Filter (EKF) is employed to fuse the landmark-based position estimates with the robot's internal odometry. Through this process, the system effectively compensates for odometry drift and minimizes the impact of environmental noise on LiDAR-based measurements, ensuring consistent pose estimation over time. Finally, the global pose generated by the *Sensor Fusion & Correction* module is transmitted to the system controller for platform control.

4.3 Localization Algorithm

To ensure the availability of a sufficient number of reference landmarks for position estimation, a total of 12 artificial landmarks were mounted on the laboratory walls (see Fig. 5). The positions of these landmarks were determined by combining measurements from a Vicon motion tracking system and the NAV245 LiDAR scanner, which significantly reduced errors compared to manual measurement methods. The landmarks were strategically arranged in distinct pairs, ensuring that the distance within each pair was unique.

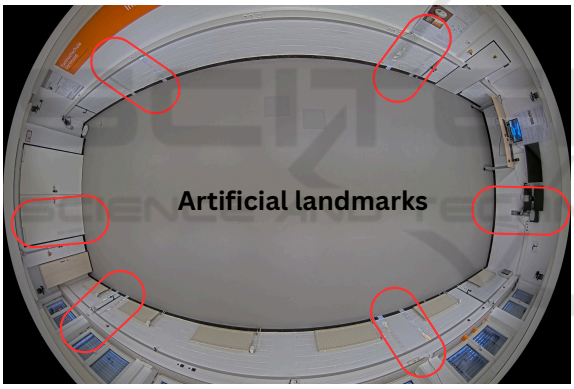
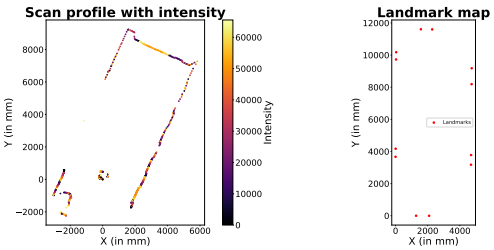


Figure 5: Landmark setup for the experimental environment.

In each scan cycle, the LiDAR sensor generates a scan profile representing the relative positions of all points detected within its field of view (see Fig. 6a). The artificial landmarks are characterized by significantly higher reflection intensity compared to natural surfaces. By applying a predefined intensity threshold, the system is able to distinguish the artificial landmarks from other objects. Once detected, these landmarks are matched against the stored landmark map (see Fig. 6b) to determine their identities. As a result of this process, each successfully identified landmark is associated with both its relative position with respect to the LiDAR sensor and its global position within the workspace.

The global position of the LiDAR scanner is estimated using the Trilateration method. This method



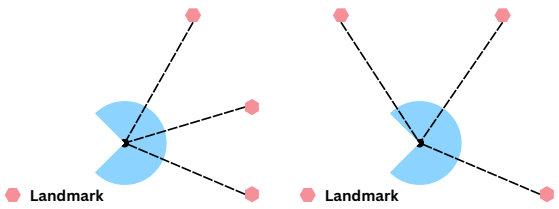
(a) LiDAR scan profile (b) Reference landmark map

Figure 6: Required data for the *Pose Estimation* Module.

utilizes the measured distances between the scanner and three previously identified artificial landmarks, located at $M = (x_m, y_m)$, $N = (x_n, y_n)$, and $K = (x_k, y_k)$. The distances to these landmarks, denoted as r_m, r_n, and r_k, are measured by the LiDAR scanner. The scanner's position (x, y) is then determined by solving the following system of nonlinear equations:

$$\begin{cases} (x-x_m)^2 + (y-y_m)^2 = r_m^2 \\ (x-x_n)^2 + (y-y_n)^2 = r_n^2 \\ (x-x_k)^2 + (y-y_k)^2 = r_k^2 \end{cases} \tag{6}$$

The accuracy of position estimation using the Trilateration method depends on the geometric arrangement of the landmarks relative to the LiDAR sensor. In cases where the landmarks have an unfavorable geometric configuration (see Fig. 7a)—such as when the reference landmarks are closely spaced or aligned along a straight line—even small errors in distance measurements can result in significant position estimation errors (Li et al., 2020). Therefore, to improve estimation accuracy and minimize the influence of measurement noise, only landmark combinations with an optimal geometric configuration (see Fig. 7b)—characterized by a low GDOP—are selected for the Trilateration process.



(a) High GDOP (b) Low GDOP

Figure 7: Relationship between the arrangement of the landmarks and the GDOP value.

The GDOP is calculated using the observation matrix A, which contains the direction vectors from the LiDAR sensor to the landmarks, according to the fol-

lowing equation:

$$GDOP = \sqrt{trace\left((A^{\mathrm{T}}A)^{-1}\right)},$$
$$\text{with } A = \begin{pmatrix} \frac{x-x_1}{r_1} & \frac{y-y_1}{r_1} \\ \dots & \dots \\ \frac{x-x_n}{r_n} & \frac{y-y_n}{r_n} \end{pmatrix}, \quad (7)$$
$$\text{and } r_i = \sqrt{(x-x_i)^2+(y-y_i)^2}$$

The global orientation of the LiDAR is calculated using the *atan2* function based on the relative positions of the detected landmarks and the LiDAR's estimated position. By applying geometric coordinate transformation to the global pose of the LiDAR, the global pose of the mobile platform can be accurately estimated. This pose is then used as an input for the *Sensor Fusion & Correction* module, where it will be fused with odometry data using the EKF to enhance the accuracy and robustness of the position estimation. For the implementation of the EKF, the state vector is defined as:

$$\mathbf{x}_t = (x_t^W, y_t^W, \theta_t^W)^{\mathrm{T}} = f(\mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{w}_t), \quad (8)$$

where $\mathbf{u}_t$ is the control input vector at time t, representing the measured relative motion of the mobile platform in its local coordinate frame, obtained from odometry data. The process noise $\mathbf{w}_t$ models uncertainties in the motion model and is assumed to follow a zero-mean Gaussian distribution to account for errors such as wheel slip and odometry drift. In the prediction step, the state is propagated based on the control input, and the covariance is updated to reflect the uncertainty:

$$\hat{\mathbf{x}}_{t|t-1} = f\left(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0\right) \quad (9)$$
$$\mathbf{P}_{t|t-1} = \mathbf{A}_t \mathbf{P}_{t-1|t-1} \mathbf{A}_t^{\mathrm{T}} + \mathbf{W}_t \mathbf{Q}_t \mathbf{W}_t^{\mathrm{T}} \quad (10)$$

where $\mathbf{P}_{t|t-1}$ is the predicted state covariance matrix, representing the uncertainty of the state after the prediction step. $\mathbf{A}_t$ and $\mathbf{W}_t$ are the Jacobians of the motion model with respect to the state and the process noise, respectively. $\mathbf{Q}_t$ is the process noise covariance matrix, representing the uncertainties introduced by the system dynamics and odometry errors.

The correction step is triggered immediately after the global pose of the mobile platform is calculated from the LiDAR measurements. It is performed using the following set of equations:

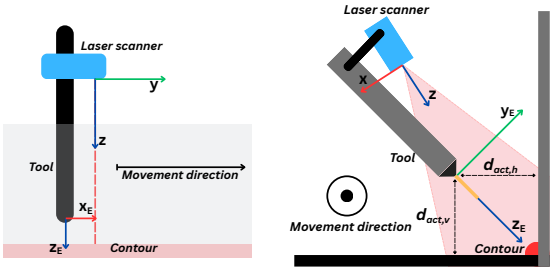
$$\tilde{\mathbf{y}}_t = \mathbf{z}_t - h\left(\hat{\mathbf{x}}_{t|t-1}, 0\right) \quad (11)$$
$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + k_t \mathbf{K}_t \tilde{\mathbf{y}}_t \quad (12)$$
$$\mathbf{K}_t = \mathbf{P}_{t|t-1}\mathbf{H}_t^{\mathrm{T}}\left(\mathbf{H}_t \mathbf{P}_{t|t-1}\mathbf{H}_t^{\mathrm{T}} + \mathbf{V}_t \mathbf{R}_t \mathbf{V}_t^{\mathrm{T}}\right)^{-1} \quad (13)$$
$$\mathbf{P}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t)\mathbf{P}_{t|t-1} \quad (14)$$

Here, the measurement function $h\left(\hat{\mathbf{x}}_{t|t-1}, 0\right)$ maps the predicted state into the measurement space. The residual $\tilde{\mathbf{y}}_t$ is calculated as the difference between the actual measurement $\mathbf{z}_t$ and the predicted measurement. The Kalman gain $\mathbf{K}_t$ is computed to balance the uncertainties between the predicted state and the measurement, considering the measurement noise covariance $\mathbf{R}_t$. The updated state estimate $\hat{\mathbf{x}}_{t|t}$ represents the corrected global pose of the mobile platform at time t, combining the prediction from odometry with the correction from the LiDAR measurements.

5 TRAJECTORY CONTROL AND CONTOUR TRACKING

5.1 Contour Tracking Algorithm

Based on the pre-defined plan, the motion controller of the mobile manipulator guides the TCP along the processing trajectory. However, factors such as wheel slippage, uneven terrain, joint backlash, and structural flex can lead to deviations between the planned and actual trajectory of the TCP. To ensure precise task execution, an algorithm is required to accurately detect these deviations. A 2D laser profile scanner is mounted on the same end-effector as the welding torch to capture the contour profile near the tool. It is oriented such that the scanning area is angled forward in the direction of motion, allowing it to detect the profile of the area just ahead of the tool path (see Fig. 8a). In T-joint welding applications, the weld seam is located at the intersection of the vertical and horizontal plates. To accurately identify the seam, the laser scanner must have a field of view that includes parts of both surfaces (see Fig. 8b).



(a) Frontal view (b) Lateral view

Figure 8: Tool and Sensor Arrangement for Contour Tracking.

During each scan cycle, the laser profile scanner captures a two-dimensional depth profile of the target surface. Each profile consists of a sequence of discrete data points, with each point containing three key

values: the x-coordinate, representing the horizontal position along the scan line; the z-coordinate, indicating the measured height relative to the scanner; and the intensity. The overall workflow of the system for detecting the machining point and adjusting the trajectory, based on laser profile data, is illustrated in Fig. 9.

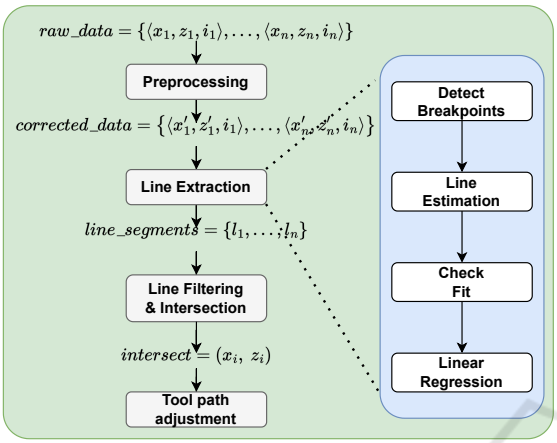


Figure 9: Workflow for welding position detection and trajectory adjustment using laser profile data.

Before any computation is performed, the raw laser profile data must be preprocessed. Irrelevant or unreliable points—typically characterized by low intensity values—are removed. This helps reduce noise and ensures that only meaningful surface features are used in the next steps. To reduce measurement noise in the distance measurements, a simple linear smoothing filter is applied. This filter gives the current value d_i the highest weight, while still considering the adjacent values to reduce local noise. The stability of breakpoint detection and line fitting is improved through this filtering approach. Such filtering enhances the quality of breakpoint detection and the fitting of line segments that follow.

$$d'_i = 0.2d_{i-1} + 0.6d_i + 0.2d_{i+1}, \quad \text{with } d_i = \sqrt{z_i^2 + x_i^2} \tag{15}$$

The lines are extracted using the Split-and-Merge algorithm. This algorithm provides a recursive approach for segmenting point data into linear components. In the first step, a line l is drawn between the first and last points of the currently considered set of points. Next, all points between the current start and end points are evaluated, and the orthogonal distance from each point d_l^i to l is computed.

$$d_l^i = \vec{v} \cdot \begin{pmatrix} x_i - x_s \\ z_i - z_s \end{pmatrix}, \quad \text{with } \vec{v} = \frac{1}{\sqrt{(z_s - z_e)^2 + (x_s - x_e)^2}} \begin{pmatrix} z_e - z_s \\ x_s - x_e \end{pmatrix} \tag{16}$$

If the maximum distance exceeds a predefined threshold, the corresponding point is classified as an outlier with respect to the current line segment l. The point set is then split at this location, and the algorithm is recursively applied to the resulting subsets. The recursion terminates once all points in a segment lie within the threshold distance from their respective fitted line. As shown in Figure 10, the black lines are fitted to the red points.

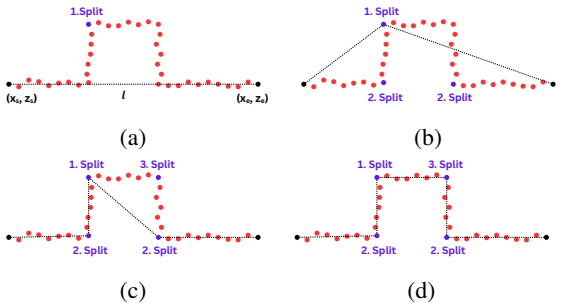


Figure 10: Stepwise representation of the operation of the recursive line extraction method.

Breakpoint analysis allows the segmentation of laser profile data into geometrically consistent sections, such as areas where the surface changes direction or curvature. Performing this analysis before applying the recursive splitting process helps reduce unnecessary iterations and improves the overall efficiency of line extraction. In this paper, the Bearing Angle (BA) method (Harati and Siegwart, 2007) is used to detect geometric discontinuities. This approach computes a bearing angle value at each scan point based on consecutive distance measurements and the scanner's angular resolution α. This bearing angle is defined as:

$$BA_i = \frac{d'_i - d'_{i-1}\cos(\alpha)}{\sqrt{d_i'^2 + d_{i-1}'^2 - 2d'_i d'_{i-1}\cos(\alpha)}} \tag{17}$$

A breakpoint is detected at point i if the absolute difference between two consecutive BA values exceeds a predefined threshold:

$$|BA_i - BA_{i-1}| \geq \theta_{\text{threshold}} \tag{18}$$

The result of the split phase is a set of data subsets, where each subset is assumed to be well approximated by an individual straight line. However, to avoid overfitting—that is, generating too many segments due to excessive sensitivity to small noise—additional validity checks are required. These include, for example, verifying whether each segment contains a sufficient number of points.

After the data has been segmented through the split phase, each subset is processed independently to estimate the best-fitting straight line. This is done by

applying linear regression to the points within each segment (Arras and Tomatis, 1999). The resulting line is then reformulated in Hesse normal form, which is particularly useful for subsequent geometric operations. The Hesse parameters are computed using the following equations:

$$\alpha = \text{atan2}\left(-\frac{1}{2}\cdot\frac{\sum_{i=1}^{N}(\bar{z}-z_i)(\bar{x}-x_i)}{\sum_{i=1}^{N}(\bar{z}-z_i)^2(\bar{x}-x_i)^2}\right) \quad (19)$$

$$p = \bar{x}\cdot\cos(\alpha)+\bar{z}\cdot\sin(\alpha) \quad (20)$$

Here, $\bar{x}$ and $\bar{z}$ denote the mean values of the coordinates of the respective point set. The computed line minimizes the orthogonal distance to the input points, providing a robust and consistent representation for downstream processing, such as computing the intersection between two adjacent line segments. Once the intersection of two neighboring fitted lines is determined, it is used as a control point to update the trajectory of the welding tool.

5.2 Trajectory Tracking Control Scheme

The proposed control scheme for trajectory tracking of the MWM is illustrated in Fig. 11. Based on the Cartesian weld path, the motion planner computes the complete trajectory $\boldsymbol{q}(t) = (\boldsymbol{q}_\text{p}(t), \boldsymbol{q}_\text{a}(t))^\text{T}$ in the configuration space. The motion interpolator computes the current state and its derivatives for each time step in t to ensure smooth and continuous motion.

The platform's motion controller is based on our previously proposed control scheme for Mecanum-driven omnidirectional platforms (Röhrig et al., 2017). The underlying kinematic model of the MWM is also adopted from our earlier work on redundant mobile manipulators, as presented in (Heß et al., 2023). The platform control consists of pose control, platform kinematics, odometry, localization, and DDSU control. The platform kinematics are derived from the platform's motion model. The DDSU controller controls the steering angle β_i and the velocity $\boldsymbol{v}_i$ for both DDSUs based on differential drive kinematics using the angular velocities $\dot{\boldsymbol{\varphi}}$ of the four driven wheels. Odometry is computed using the wheel encoder values $\boldsymbol{\varphi}$ and the measured passive steering angles $\boldsymbol{\beta}$. Localization is performed by fusing the odometry with data from a SICK NAV245 LiDAR.

The tracking of the weld path is done by a laser sensor that measures the deviation from the weld path. Since the robotic arm is much more precise and dynamic than the robotic platform, the tracking control uses the arm to control the deviation from the weld path. The deviation is measured in the end-effector coordinate frame, hence the end-effector Jacobian

$${}^\text{E}J_\text{a}(\boldsymbol{q}_\text{a}) = \frac{\partial \boldsymbol{f}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} {}^\text{E}R_0 & \boldsymbol{0}_{3\times3} \\ \boldsymbol{p}_{3\times3} & {}^\text{E}R_\text{p} \end{pmatrix} {}^\text{p}J_\text{a}(\boldsymbol{q}_\text{a})\dot{\boldsymbol{q}}_\text{a} \quad (21)$$

is used to control the deviation. The tracking control scheme is based on the well known *Resolved-Rate Motion Control* (see (Corke, 2023)). When the arm correction $\Delta\boldsymbol{q}$ becomes large, the motion planner calculates a new trajectory to correct the platform's pose and reduce $\Delta\boldsymbol{q}$.

6 EXPERIMENTAL EVALUATION

To validate the proposed algorithm, two experiments were conducted: one assessing the accuracy of the mobile platform, and the other evaluating contour tracking in a welding task.

6.1 Experimental Setup

The MWM used for the experiments described here comprises a platform equipped with two DDSUs and a KUKA KR 6 R900-2 robotic arm, controlled via a KRC5 controller. An embedded PC running a real-time Linux operating system that connects the mobile platform's DDSUs via Controller Area Network (CAN) bus and the manipulator's KRC5 controller via real-time Ethernet. Serving as a centralized real-time control system, the system controller PC synchronizes the setpoints of all components in real time to ensure coordinated operation. Every 200 ms, the SICK NAV245 LiDAR provides a full 270° scan, which is used by the platform controller as feedback for the pose control loop (see Fig. 11). The scanCONTROL 3012-100 laser profile sensor, mounted on the same end-effector as the welding torch, is used in Trajectory Tracking to enable closed-loop control of the robotic arm. In the experimental setup, the T-joint is simulated using an L-shaped profile instead of two separate workpieces (see Fig. 12). For ground truth comparison, the MWM's global pose is obtained via a Vicon motion capture system, which offers high-precision 6-DoF tracking.

6.2 Experimental Results

In order to evaluate the effectiveness of the proposed localization method, we applied a series of algorithms to the dataset obtained from the NAV245 laser scanner. The localization accuracy improved progressively with each enhancement in the algorithmic pipeline. Trilateration served only as the baseline, producing the largest

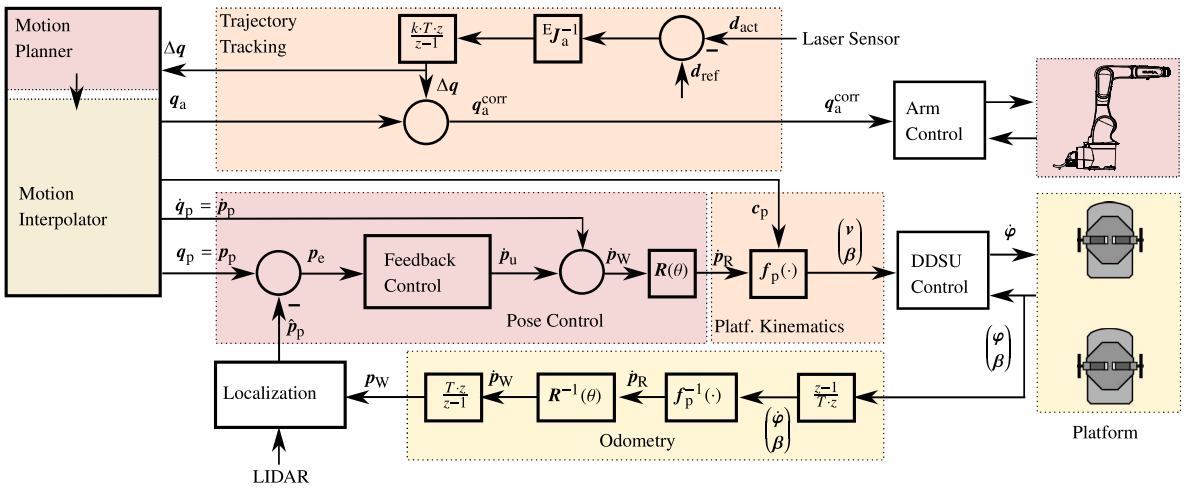
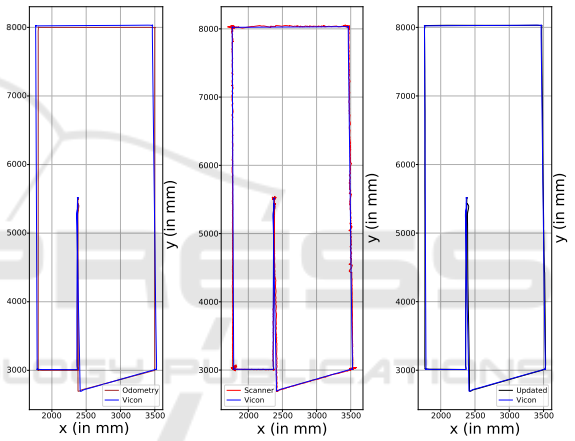


Figure 11: Block diagram for tracking a welding path with MWM.

Figure 12: Simulation of T-Joint weld using L-Profile.



(a) Using odometry data. (b) Using Trilateration + GDOP. (c) Trilateration + GDOP + EKF.

Figure 13: Comparison of estimated trajectories using different methods.

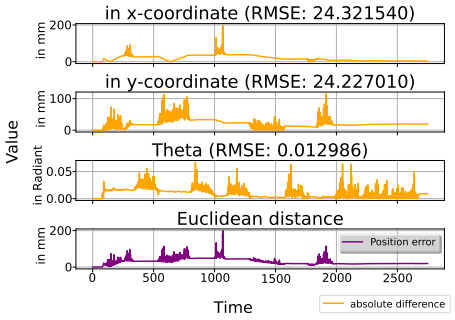
variance in position estimates. Incorporating the optimal geometric configuration of the detected landmarks — based on GDOP analysis — significantly reduced the standard deviation, minimized measurement noise, and produced position estimates that were more stable and more tightly clustered around the true location.

Next, the MWM was driven along a predefined trajectory. During this motion, both odometry data and scan measurements were recorded. Figure 13 show the estimated trajectory using (a) raw odometry data, (b) trilateration with GDOP, and (c) sensor fusion combining odometry with trilateration + GDOP using an Extended Kalman Filter. Each method was compared against ground-truth data obtained from a Vicon motion capture system.

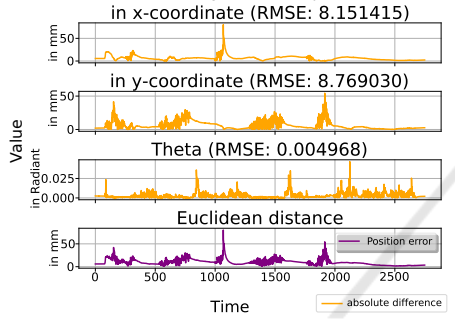
Due to wheel slippage and integration drift, the trajectory estimated from odometry diverges significantly from the ground truth. By contrast, trilateration enhanced with GDOP filtering yields a trajectory that more closely aligns with the actual path. However, residual noise remains—particularly in turning segments or in areas with limited landmark coverage—likely due to suboptimal geometric configurations for trilateration. The EKF-based data fusion method effectively mitigates these errors, significantly enhancing the accuracy and stability of mobile platform localization.

To demonstrate the effectiveness of the EKF-based localization approach, Figure 14 presents the absolute differences between the estimated and ground-truth values over time. The odometry-only method resulted in Root Mean Square Errors (RMSE) of 24.32 mm (x), 24.23 mm (y), and 0.01299 rad (θ). In contrast, the EKF-based fusion significantly reduced these errors, achieving RMSEs of 8.15 mm (x), 8.77 mm (y), and 0.00497 rad (θ). These results highlight the robustness and precision of the proposed fusion method

in dynamic operating scenarios. However, for precise end-effector control during welding, local contour tracking is required.



(a) Using Odometry



(b) Using EKF

Figure 14: Absolute error between estimated and ground-truth values.

Contour tracking aims to accurately detect the machining point, which, in the context of simulated T-joint welding, refers to the intersection line between two metal plates. Maintaining precise and continuous alignment between the welding torch and this intersection line is a critical factor that determines the quality and stability of the weld. Figure 15 illustrates a typical depth and intensity profile acquired from the 2D profile laser.

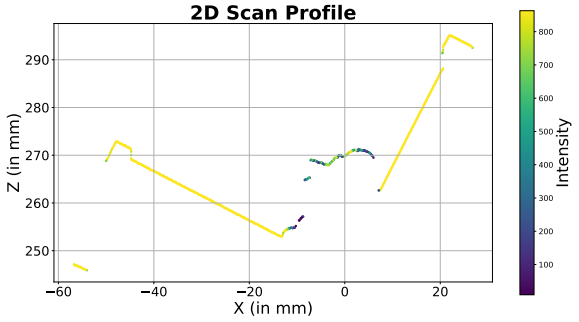
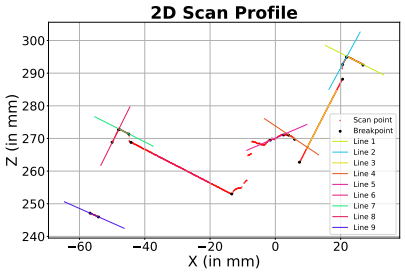
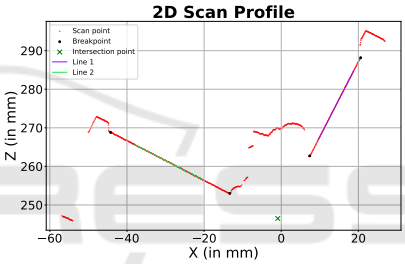


Figure 15: Depth and Intensity Profile from 2D Laser Scan.

The raw scan profile is processed to extract line segments using the proposed contour tracking algorithms, as shown in Figure 16a. In the final step, the intersection of relevant line segments is computed to determine the precise machining point, as shown in Figure 16b. This intersection point is then used to regulate the relative position between the laser sensor and the workpiece. Since both the torch and the sensor are mounted on the same end-effector, this adjustment also determines the relative position between the torch and the workpiece.



(a) Breakpoint detection and line segmentation



(b) Intersection point estimation

Figure 16: Estimation of the precise machining point through line extraction and intersection of 2D scan profile data.

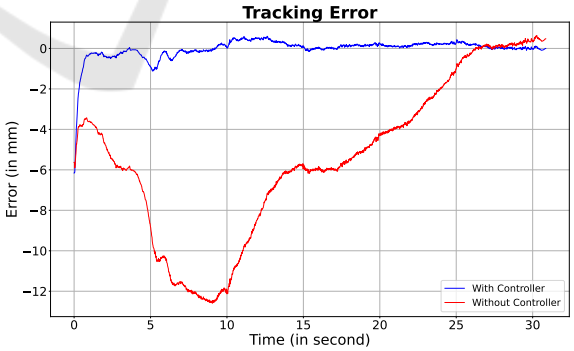


Figure 17: Tracking error over time.

Figure 17 illustrates the deviation of the torch from the welding path during motion. While the MWM moves along the workpiece, the mobile platform maintains global localization using LiDAR, and the robotic arm compensates for any deviations from the predefined weld path. The blue curve represents the tracking error with active contour tracking using laser sensor

feedback, whereas the red curve shows the error without such feedback control. The tracking system effectively corrects deviations of the platform, which reach up to 12 mm in this experiment. The blue curve shows that the deviation does not exceed 2 mm when the tracking controller is used.

7 CONCLUSION

In this paper, we presented a MWM designed for high-precision contour tracking in large-scale industrial environments. The proposed system combines a 6-DoF robotic arm suitable for welding tasks with an omnidirectional mobile platform driven by DDSUs, allowing for smooth and stable motion with reduced vibration. The DDSUs contain kinematic constraints that must be considered in motion planning and control. Each unit has a limited steering angle and can be driven in two different configurations by changing the direction of the wheel speeds. The entire weld trajectory is planned in Cartesian space, with explicit consideration of the motion constraints introduced by the DDSU-driven platform. To achieve accurate localization, an Extended Kalman Filter fuses LiDAR data with wheel odometry, resulting in a centimeter-level pose estimate. However, this level of accuracy is insufficient for high-precision welding. Therefore, a 2D laser profile scanner mounted on the end-effector is used to detect the machining point and measure deviations from the weld path. Since the robotic arm is significantly more accurate and dynamic than the mobile platform, the tracking controller uses the arm to adjust the torch trajectory in order to compensate for deviations from the weld path.

Experimental results confirm that the proposed trajectory tracking control scheme achieves millimeter-level accuracy at the end-effector relative to the workpiece, satisfying the precision requirements of industrial welding applications.

REFERENCES

Arras, K. and Tomatis, N. (1999). Improving robustness and precision in mobile robot localization by using laser range finding and monocular vision. pages 177 – 185.

Bae, J.-J. and Kang, N. (2016). Design optimization of a mecanum wheel to reduce vertical vibrations by the consideration of equivalent stiffness. *Shock and Vibration*, 2016:1–8.

Corke, P. (2023). *Manipulator Velocity*, pages 307–331.

Delamare, M., Boutteau, R., Savatier, X., and Iriart, N. (2020). Static and dynamic evaluation of an uwb localization system for industrial applications. *Sci*, 2:23.

Gawel, A., Blum, H., Pankert, J., Krämer, K., Bartolomei, L., Ercan Jenny, S., Farshidian, F., Chli, M., Gramazio, F., Siegwart, R., Hutter, M., and Sandy, T. (2019). A fully-integrated sensing and control system for high-accuracy mobile robotic building construction.

Ghodsian, N., Benfriha, K., Olabi, A., Gopinath, V., and Arnou, A. (2023). Mobile manipulators in industry 4.0: A review of developments for industrial applications. *Sensors*, 23.

Harati, A. and Siegwart, R. (2007). A new approach to segmentation of 2d range scans into linear regions. pages 2083–2088.

Heß, D., Trinh, D., Röhrig, C., and Parys, M. (2023). Mobilerobot: Control of a redundant kinematic using drive-steering modules for mobile manipulation.

Jacobs, T. (2018). Omnidirectional robot undercarriages with standard wheels - a survey. pages 1–6.

Jacobs, T., Connette, C., and Hägele, M. (2012). Design of wheel modules for non-holonomic, omnidirectional mobile robots in context of the emerging control problems. pages 1–4.

Jacobs, T. and Schaefer, E. (2020). Design of compact omnidirectional wheel modules with internal differential kinematics. In *ISR 2020; 52th International Symposium on Robotics*, pages 1–7.

KUKA Robotics (2025). *Roboter Lineareinheiten*. KUKA Robotics. Available online: https://www.kuka.com/de-de/produkte-leistungen/robotersysteme/roboterperipherie/lineareinheiten (accessed on 5 June 2025).

Li, B., Zhao, K., and Shen, X. (2020). Dilution of precision in positioning systems using both angle of arrival and time of arrival measurements. *IEEE Access*, 8:192506–192516.

Röhrig, C. and Heß, D. (2020). *Mobile Manipulation for Human-Robot Collaboration in Intralogistics*, pages 1–20.

Röhrig, C., Heß, D., and Künemund, F. (2017). Motion controller design for a mecanum wheeled mobile manipulator.

Sereinig, M., Werth, W., and Faller, L.-M. (2020). A review of the challenges in mobile manipulation: systems design and robocup challenges. *e & i Elektrotechnik und Informationstechnik*, 137:1–12.

SICK AG (2022). *NAV245-10100 Product Datasheet*. SICK AG, Waldkirch, Germany. Available online: https://www.sick.com/1074308 (accessed on 15 May 2025).

Wang, J., Li, C., Li, B., Pang, C., and Fang, Z. (2021). High-precision and robust localization system for mobile robots in complex and large-scale indoor scenes. *International Journal of Advanced Robotic Systems*, 18(5):17298814211047690.

Zhewen, Z., Hongliu, Y., Chengjia, W., Pu, H., and Jiangui, W. (2024). A comprehensive study on mecanum wheel-based mobility and suspension solutions for intelligent nursing wheelchairs. *Scientific Reports*, 14.