

Robot Calibration

Odometry and extrinsic laser calibration with the robot's sensory system

Student: Ricardo B. Sousa  ^{1,2}

Supervisors: António P. Moreira  ^{1,2}, Héber Miguel Sobreira  ²

¹ Faculty of Engineering of the University of Porto

(email: up201503004@edu.fe.up.pt, amoreira@fe.up.pt)

² INESC TEC – Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência

(email: ricardo.b.sousa@inesctec.pt, heber.m.sobreira@inesctec.pt)

Abstract. An autonomous robot requires localization systems to determine its location in the environment. Localization methods such as wheel odometry (dead reckoning) or map matching with laser scanners (absolute localization) are usually implemented on industrial mobile robots. However, wheel odometry requires the kinematic parameters of the robot, and map matching with laser scanners needs the laser's relative pose to the robot's local frame. This work formulates a calibration method to estimate the odometry and the extrinsic 2D laser parameters simultaneously. The proposed calibration algorithm assumes that the odometry and laser scan matching estimations should converge to the same value optimizing the parameters required by each localization method. Preliminary results show that indeed the cost function formulated in this work decreases with the parameters' global optimization, even though further work is still required to evaluate the calibration method's accuracy.

1 Introduction

Localization is a fundamental system required for a robot to navigate autonomously in a dynamic and unpredictable environment. Only when the robot is able to determine its location in the environment which is inserted that it is possible to follow the desired trajectory and evaluate if the robot reached its goal [1]. Two localization methods commonly applied together are absolute and dead reckoning. Absolute localization usually depends on map matching, active or passive landmarks, or satellite signals. Dead reckoning is usually based on wheel, visual and/or laser odometry [2].

Wheel odometry uses the robot's kinematic model and wheels displacement in order to estimate the robot's relative position to a previous time instant. It is inexpensive, high sampling rate, simple, and easy to apply in real-time [2]. The kinematic model is defined by the physical dimensions of the robot such as the diameters of the wheels, steering angle offsets, among other parameters, depending on the robot's steering geometry [1]. So, odometry calibration can be used to estimate more accurate kinematic parameters (relative to measuring them) and reduce the odometry error.

As for localization methods using laser scanners (e.g., map matching or laser odometry), it is required to know the relative sensor's pose – the so-called extrinsic parameters – to transform all measurements into a common coordinate frame for tasks like localization or sensor fusion. The extrinsic sensor parameters are defined by 3 DoFs (Degrees of Freedom) for translation and 3 DoFs for rotation (e.g., yaw, pitch, and roll components). However, the estimation of these parameters may have errors associated with them. For example, errors of 1° in the orientation of a safety laser can lead to position errors up to 0.5m at a 30m range. The 0.5m error can put in danger the robots or persons moving through the environment by not detecting correctly if an obstacle is in the imminence of colliding with the robot [3]. So, extrinsic sensor calibration is crucial to reduce the errors associated with the transformation of the sensors' data into a common coordinate frame.

The literature already has several methods proposed for odometry calibration. The ones intended for calibrating both the odometry and the extrinsic parameters of a 2D laser scanner were proposed by Censi *et al.* (2013) [4], Kallasi *et al.* (2017) [5], and Galasso *et al.* (2019) [6]. These methods assume that the laser scanner is horizontally mounted on the robot (zero pitch and roll – 3 DoFs). Furthermore, the works [4,5,6] are intended for a specific steering geometry. As for extrinsic sensor calibration for the setup robot–laser scanner, Gao and Spletzer [3] and Underwood *et al.* (2010) [7] estimated the 6 DoFs of a 2D laser scanner using the distance error of known features (retro-

reflective tape on vertical poles and a vertical pole perpendicular to the ground, respectively), and Li *et al.* (2019) [8] focused on 3D laser scanners. However, the works [3, 7, 8] did not calibrate the odometry of the robot.

This work proposes an automatic calibration method for calibrating the odometry of wheeled robots and the extrinsic parameters of 2D laser scanners simultaneously. The proposed calibration method assumes that the relative position error of the robot's odometry and a map matching algorithm – using the Iterative Closest Point (ICP) [9] algorithm to match the 2D laser scanner point clouds – should converge to 0 optimizing the odometry and extrinsic sensor parameters. This work has the following contributions:

- generic odometry calibration algorithm only requiring the definition of the kinematic model for the intended steering geometry (method was tested only with a tricycle robot);
- extrinsic sensor calibration of 2D laser scanners assuming that the sensor is horizontally mounted on the robot (similar to [4, 5, 6]);
- optimization of the odometry and extrinsic parameters of a 2D laser scanner to reduce the quadratic relative position error using a derivative-free optimization algorithm (DIRECT [10]);
- automatic calibration procedure that only requires the operator to drive the robot through circles with increasing and/or decreasing radius in both clockwise (CW) and counterclockwise (CCW) directions.

Our previous works [11, 12, 13] are the basis for the discussion of the related work providing a comprehensive literature review on odometry and extrinsic sensor calibration methods. To the best of our knowledge, the proposed method is the first to optimize the odometry and the extrinsic sensor parameters simultaneously independently of the steering geometry of the robot.

The rest of the paper is organized as follows. Section 2 presents the related work. Section 3 formulates the kinematic model of a tricycle robot, the extrinsic parameters of a 2D laser scanner, and the proposed calibration algorithm. Section 4 analyses the results obtained from the experiments made. Section 5 presents the conclusions of the work performed and future developments for the proposed method.

2 Related Work

Odometry calibration of wheeled robots is vastly studied in the literature. Most of the existent works use an external tracking system (relative to the robot) to provide ground-truth data to the calibration algorithm. Still, three methods focused on calibrating the odometry and extrinsic sensor parameters of a 2D laser scanner simultaneously. These three calibration methods assumed that the sensor is horizontally mounted on the robot (zero pitch and roll) simplifying the extrinsic sensor calibration problem from 6 DoFs to 3 DoFs: 2 for translation (x and y-axis) and 1 for orientation (rotation of the coordinate frame). First, Censi *et al.* (2013) [4] calibrated the odometry of a differential robot (estimation of the diameter of the wheels and the wheelbase) and the extrinsic parameters of a 2D laser scanner (3 DoFs) by solving the constrained least-squares problem using Lagrange multipliers. Censi *et al.* [4] did not separate the two calibration problems. Next, Kallasi *et al.* (2017) [5] and Galasso *et al.* (2019) [6] also used the least-squares algorithm for the calibration algorithm. These methods differ from Censi *et al.* [4] in the independent formulation of the calibration algorithm for estimating the odometry and the extrinsic sensor parameters. Also, the works of Kallasi *et al.* [5] and Galasso *et al.* [6] only differ from each other on the steering geometry used to formulate the odometry calibration algorithm: [5] is for tricycle robots, and [6] is for Ackerman or dual-drive robots. However, Kallasi *et al.* [5] and Galasso *et al.* [6] did not estimate the distance between the front and the rear wheels for the intended steering geometries because these methods assumed that this distance is known. The works [4, 5, 6] do not require initial estimations for the calibration algorithms.

As for extrinsic calibration-only methods that estimate the laser's pose relative to the robot coordinate frame, Gao and Spletzer (2010) [3] relaxed the non-linear calibration problem to a Second-Order Cone Program (SOCP). The SOCP minimized the distance error of known features (retro-reflective tape on vertical poles) while being able to add constraints to the problem (e.g., calibration tolerances). The work proposed by Gao and Spletzer [3] estimates the 6 DoFs of a 2D laser scanner. Due to the type of features used, [3] requires distance and remission values from the laser. Another extrinsic 2D laser calibration method was proposed by Underwood *et al.* (2010) [7]. This work used the MATLAB `fmincon` function to minimize the distance error of known features: a vertical pole and the ground (it is assumed that the pole is perpendicular to the ground). The vertical pole was used to provide distance error along the x and y-axis, and the ground provided an error measure along the z-axis. The works [3, 7] have two common requirements: the robot 6 DoFs relative to the world (e.g., external tracking system) and an initial estimation for the extrinsic parameters. Lastly, Li *et al.* (2019) [8] proposed an extrinsic calibration method for 3D laser scanners focused only on the transformations between the laser and the vehicle's frames. First, it ignored the translation vector to compute the rotation matrix relative to the robot. And then, the translation between frames is estimated. Li *et al.* [8] requires ground plane without slope (z-axis assumed to be perpendicular to the floor), presence of perpendicular planes to the floor in the scene, and the heading direction of the vehicle be perpendicular to the wall. However, [8] does not require initial estimation.

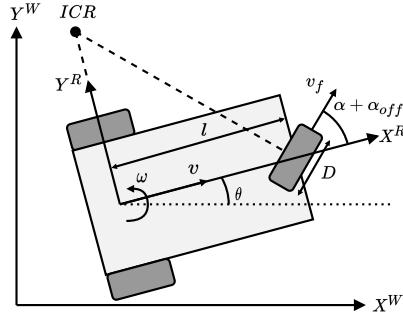


Fig. 1. Tricycle Steering Geometry

In summary, the existent literature is limited for the case of calibrating the odometry and the extrinsic parameters of a 2D laser scanner simultaneously. Even though Censi *et al.* [4] formulated this problem for differential robots, its work cannot be adapted to other steering geometries. Kallasi *et al.* [5] and Galasso *et al.* [6] did not consider the distance between the front and rear wheels for Ackerman, tricycle, and dual-drive robots. Also, the manual process of calibrating the odometry and extrinsic parameters is a tiresome process requiring at least one hour for each robot and an experienced operator to perform it [5]. So, automatic calibration algorithms for estimating the odometry and extrinsic parameters are desirable for mobile robots.

3 Method

3.1 Odometry equations for tricycle robots

First, let us define the forward kinematics of the tricycle robot (steering geometry used in the experiments). The tricycle geometry illustrated in figure 1 always has only one Instantaneous Center of Rotation (ICR), given that tricycle robots have only one steering wheel. The kinematic parameters of tricycle robots are the diameter of the front-driven wheel (D), the steering angle offset (α_{off}), and the distance between the front and rear wheels (l).

The linear displacement of the front-driven wheel at a time instant k ($\Delta d_{f,k}$) is defined in equation 1. It depends on the diameter of the wheel (D), the gear reduction ratio (n), the encoders' resolution (C_e), and the impulses increment ($\#i_k$) measured by the respective encoder.

$$\Delta d_{f,k} = \frac{\pi D}{nC_e} \#i_k \quad (1)$$

Next, the wheel's linear displacement ($\Delta d_{f,k}$) and measured steering angle (α_k) compute the robot's linear (Δd_k) and angular ($\Delta\theta_k$) displacements, as illustrated in equations 2 and 3 [14]. The robot's linear and angular displacements are relative to its local coordinate frame ($\{X^R, Y^R\}$). The steering angle offset (α_{off}) influences both displacements, offsetting the value of the real steering angle of the robot ($\alpha_k + \alpha_{off}$).

$$\Delta d_k = \cos(\alpha_k + \alpha_{off}) \cdot \Delta d_{f,k} \quad (2)$$

$$\Delta\theta_k = \frac{\sin(\alpha_k + \alpha_{off})}{l} \cdot \Delta d_{f,k} \quad (3)$$

Finally, the robot's pose relative to the world frame ($\{X^W, Y^W\}$) can be computed by the odometry equations using a discrete approximation. This approximation is illustrated in the set of equations 4 for tricycle robots (and also applicable to differential drive and Ackerman steering geometries). Note that these equations compute the odometry pose at a time instant j relative to a previous non-consecutive instant i .

$$\begin{aligned} x_{odo,j} &= x_{odo,i} + \sum_{k=i+1}^j \Delta d_k \cdot \cos(\theta_{k-1} + \Delta\theta_k/2) \\ y_{odo,j} &= y_{odo,i} + \sum_{k=i+1}^j \Delta d_k \cdot \sin(\theta_{k-1} + \Delta\theta_k/2) \\ \theta_{odo,j} &= \theta_{odo,i} + \sum_{k=i+1}^j \Delta\theta_k \end{aligned} \quad (4)$$

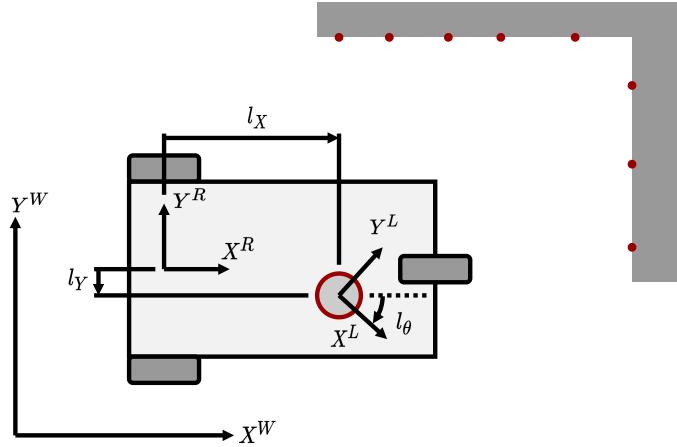


Fig. 2. Extrinsic Parameters of a 2D Laser Scanner with 3 DoFs

The odometry relative displacement between two different time instants i and j ($\Delta x_{odo,u}, \Delta y_{odo,u}, \Delta \theta_{odo,u}$) of a robot is defined in equation 5. This displacement is computed relative to the robot coordinate frame at the time instant i ($\{X^{R,i}, Y^{R,i}\}$). The index u represents different values of the time instants i and j .

$$\begin{aligned} \Delta x_{odo,u} &= x_{odo,j} - x_{odo,i} \\ \Delta y_{odo,u} &= y_{odo,j} - y_{odo,i} \\ \Delta \theta_{odo,u} &= \theta_{odo,j} - \theta_{odo,i} \end{aligned} \quad (5)$$

3.2 2D laser point clouds matching

A 2D laser scanner retrieves point clouds from the environment in which the sensor is inserted. These point clouds \mathcal{P}^L are relative to the laser's coordinate frame ($\{X^L, Y^L\}$). The extrinsic sensor parameters allow the transformation of the point clouds ($\mathcal{P}^L \rightarrow \mathcal{P}^R$) into the robot's local coordinate frame ($\{X^R, Y^R\}$). The parameters can be defined by 6 DoFs: 3DoFs for translation and 3 DoFs for rotation (e.g., yaw, pitch, and roll components). In this work, we will consider the same simplification as in works [4, 5, 6], i.e., the 2D laser scanner is assumed to be horizontally mounted relative to the robot (the most usual pose for navigation laser scanners of mobile robots in industrial environments). This simplification reduces the number of extrinsic parameters from 6 to 3: 2 DoFs for translation (l_x and l_y offsets in the robot's local frame) and 1 DoF for orientation (rotation angle l_θ of the laser coordinate frame relative to the robot's local one), as illustrated in figure 2. Equation 6 computes the transformation of a 2D laser point cloud in the laser's frame (\mathcal{P}^L) into the robot's local frame (\mathcal{P}^R). The homogeneous transformation that defines the transformation from the laser to the robot's coordinate frame (H_L^R) depends on the 3 DoFs considered in this work for extrinsic sensor calibration, as illustrated in equation 7.

$$\mathcal{P}^R = H_L^R \cdot \mathcal{P}^L \quad (6)$$

$$H_L^R = \begin{bmatrix} \cos(l_\theta) & -\sin(l_\theta) & 0 & l_x \\ \sin(l_\theta) & \cos(l_\theta) & 0 & l_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Then, we can use point clouds matching algorithms to estimate the robot's relative position ($\Delta x_{las,u}$ and $\Delta y_{las,u}$) and angular ($\Delta \theta_{las,u}$) displacements from two laser point clouds acquired at different i and j time instants ($\mathcal{P}^{R,i}$ and $\mathcal{P}^{R,j}$, respectively). The Iterative Closest Point (ICP) [9] algorithm available in the Point Cloud Library (PCL) [15] was applied to minimize the difference between these two laser point clouds and estimate the homogeneous transformation that transforms one onto the other. In this work, the point cloud acquired at the time instant i ($\mathcal{P}^{R,i}$) is the target one, i.e., the ICP [9] optimizes the homogeneous matrix $H_{R,j}^{R,i}$ that transforms $\mathcal{P}^{R,j}$ into $\mathcal{P}^{R,i}$, as illustrated in equation 8. The main reason is that the homogeneous matrix $H_{R,j}^{R,i}$ depends on the relative position ($\Delta x_{las,u}$ and $\Delta y_{las,u}$) and angular ($\Delta \theta_{las,u}$) displacements of the robot, as presented in equation 9.

$$\mathcal{P}^{R,i} = H_{R,j}^{R,i} \cdot \mathcal{P}^{R,j} \quad (8)$$

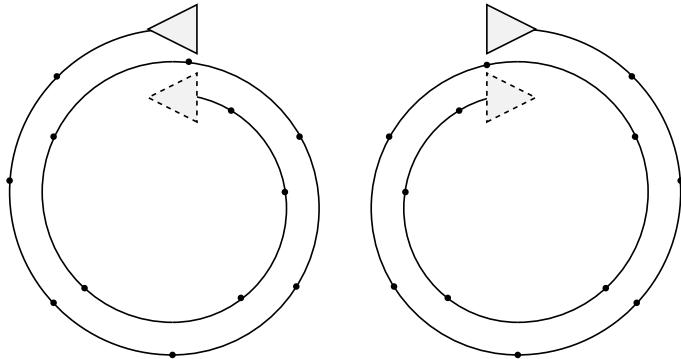


Fig. 3. Calibration Path Used in this Work, Kallasi *et al.* [5] and Galasso *et al.* [6]

$$H_{R,j}^{R,i} = \begin{bmatrix} \cos(\Delta\theta_{las,u}) & -\sin(\Delta\theta_{las,u}) & 0 & \Delta x_{las,u} \\ \sin(\Delta\theta_{las,u}) & \cos(\Delta\theta_{las,u}) & 0 & \Delta y_{las,u} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

3.3 Calibration algorithm

As explained in Sections 3.1 and 3.2, it is possible to estimate the robot's position and angular relative displacements with odometry ($\Delta x_{odo,u}, \Delta y_{odo,u}, \Delta\theta_{odo,u}$) or laser scan matching ($\Delta x_{las,u}, \Delta y_{las,u}, \Delta\theta_{las,u}$) within a time interval $t \in [i, j]$. So, the two estimation methods could be compared for adjusting the odometry and extrinsic sensor parameters. The method proposed in this section formulates a cost function equal to the sum of quadratic position errors ($\sum_{u=1}^N \varepsilon_{pos,u}^2$), as illustrated in equation 10. The position error compares the position displacement estimated by odometry to the laser scan matching's estimation. It is assumed that the global minimum of the cost function should correspond to the true values of the odometry (D, l, α_{off}) and extrinsic sensor parameters (l_x, l_y, l_θ).

$$\min_{D, l, \alpha_{off}, l_x, l_y, l_\theta} \sum_{u=1}^N \varepsilon_{pos,u}^2 = \sum_{u=1}^N (\Delta x_{odo,u} - \Delta x_{las,u})^2 + (\Delta y_{odo,u} - \Delta y_{las,u})^2 \quad (10)$$

However, the gradient of the cost function formulated in equation 10 could be difficult to obtain directly. The main reason is that the relative position displacement of the robot estimated by laser scan matching uses already an optimization algorithm (ICP [9]). So, gradient-based optimization algorithms such as Levenberg-Marquardt or gradient descent could not be suitable for simultaneous odometry and extrinsic sensor calibration. Our calibration method used a derivative-free optimization algorithm to minimize the cost function illustrated in equation 10. The algorithm chosen was the DIRECT global optimization algorithm [10] because this algorithm is available in the NLOpt C++ open-source library [16] for nonlinear optimization. The DIRECT [10] algorithm requires defining the parameters' initial estimation, their upper and lower bounds, and the maximum of iterations.

In terms of a calibration path, the proposed method is not path-specific. However, the choice of the calibration path should be one that makes observable all the odometry and extrinsic sensor parameters considered in the optimization. Given that a circular-based calibration path with decreasing and/or increasing radius proved already in the works [5, 6] to satisfy the observability requirement, we suggest also using this type of calibration path with our method. The circular-based calibration path is illustrated in figure 3.

As for defining the time instants i and j , Kallasi *et al.* [5] and Galasso *et al.* [6] specified that the start and end of 180° curves of the calibration path (estimated with odometry data) correspond to instants i and j , respectively. This metric is more suitable for paths highly dependent on angular movement. Although we used the same type of calibration path, we defined that the robot's position should be measured (the cost function defined in equation 10 does not require the robot's orientation data) after a certain value of linear displacement (accumulation of Δd_k). We consider this metric more suitable than the one used in [5, 6] because most of the paths for mobile robots always perform at least linear motion. Also, we change the definition of the time instants i and j to the one illustrated in figure 4, i.e., each time interval considered in the cost function represents the path between the initial position and the data acquisition points along the path defined by the accumulative linear displacement metric.

A final observation is that even though this work only tested the calibration algorithm with only one steering geometry, the proposed calibration method could be adapted to other steering geometries. The only requirement is that the forward kinematics are defined as dependent on the odometry parameters.

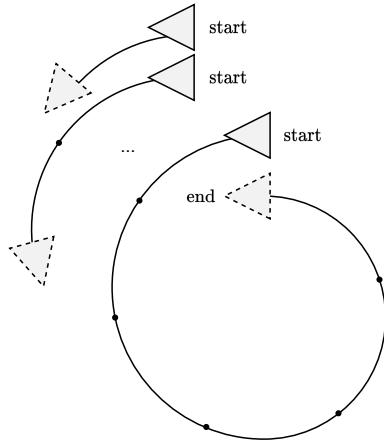


Fig. 4. Definition of Path Segments Considered in this Work

3.4 Calibration procedure

Next, it is enumerated the steps required to perform our calibration algorithm with the circular-based calibration path:

1. Run the robot through a curvilinear motion in CW direction:
 - Set the initial radius taking into consideration the available space
 - Reduce the curvature radius (similar to a spiral)
 - Save the current timestamp after the robot has gone through 0.5m linear displacement (considering the odometry estimation)
2. Repeat step 1 in CCW direction (do not stop the robot) with a increasing curvature radius
3. Perform the optimization procedure
4. Adjust the robot's odometry and the laser's extrinsic parameters given the result from the previous step

Note that the calibration procedure should be performed with a low nominal velocity. This requirement intends to avoid not only wheels slipping – it would induce error for the odometry data – but also to reduce possible noise in the laser point clouds data (e.g., due to the vibration of the robot's motion).

4 Experiments

4.1 Robot used in the experiments

The proposed calibration algorithm was tested on a tricycle robot: the Jarvis mobile platform from INESC TEC - Institute for Systems and Computer Engineering, Technology and Science illustrated in figure 5a. This robot has the INESC TEC Navigation Stack implemented on the Robot Operating System (ROS) framework. The odometry data is published on the topic `motors_outputs` at a frequency rate of 25Hz. The Jarvis mobile platform has two 2D laser scanners, although only one of them is used for navigation. The one used for navigation is a Sick Nav350 2D LiDAR sensor that publishes the laser data on the topic `scan_nav` at an 8Hz frequency rate. In figure 5b, it is possible to observe the area used to perform the calibration procedure proposed in this work. This area provides a structured and almost static environment (to reduce possible noise in the laser point clouds data).

4.2 Experimental results

The experimental results obtained are presented in table 1. All the experiments performed the calibration procedure described in Section 3.4. Also, the same initial estimation and parameters' range was used for the experiments: $D = 0.1522\text{m} \pm 0.01\text{m}$, $l = 0.5891\text{m} \pm 0.025\text{m}$, $\alpha_{off} = -56.72^\circ \pm 2.5^\circ$, $l_x = 0.1940\text{m} \pm 0.02\text{m}$, $l_y = 0\text{m} \pm 0.02\text{m}$, and $l_\theta = 0^\circ \pm 2.5^\circ$. This initial estimation was obtained by performing a manual calibration procedure specific to tricycle steering geometry. A maximum of 200 iterations was set for the DIRECT [10] global optimization algorithm.

Analyzing the results presented in table 1, the first observation is that the cost function always decreased after calibrating the parameters. This observation indicates that the error of the two localization methods reduced relative to each other. A comparison example from before and after calibration is presented in figure 6. Figure 6b shows that

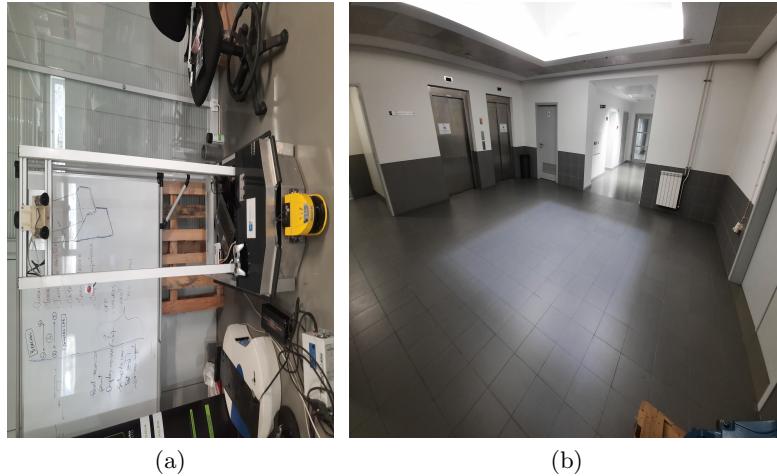


Fig. 5. Experimental Conditions Considered in this Work: (a) Tricycle Robot Used in the Experiments; (b) Available Area for the Calibration Procedure

Table 1. Synthesis of the Experimental Results Obtained with the Proposed Calibration Algorithm

i	D (m)	l (m)	α_{off} (°)	l_x (m)	l_y (m)	l_θ (°)	$\sum_u \varepsilon_{pos,u}^2$ (m ²)	D (m)	l (m)	α_{off} (°)	l_x (m)	l_y (m)	l_θ (°)	$\sum_u \varepsilon_{pos,u}^2$ (m ²)
1	0.152200	0.589100	-56.720	0.194000	0.000000	0.000	0.704002	0.152941	0.594656	-56.720	0.201407	0.019259	-1.111	0.349786
2	0.152200	0.589100	-56.720	0.194000	0.000000	0.000	0.880756	0.152941	0.596507	-56.720	0.211778	0.019259	-1.111	0.340010
3	0.152200	0.589100	-56.720	0.194000	0.000000	0.000	0.806411	0.154669	0.597125	-56.720	0.182148	-0.014815	-0.185	0.433930
4	0.152200	0.589100	-56.720	0.194000	0.000000	0.000	1.211787	0.156644	0.605767	-56.720	0.180667	0.000000	0.000	0.677617
5	0.152200	0.589100	-56.720	0.194000	0.000000	0.000	0.746577	0.152941	0.590952	-56.720	0.176222	-0.017778	-1.111	0.448665

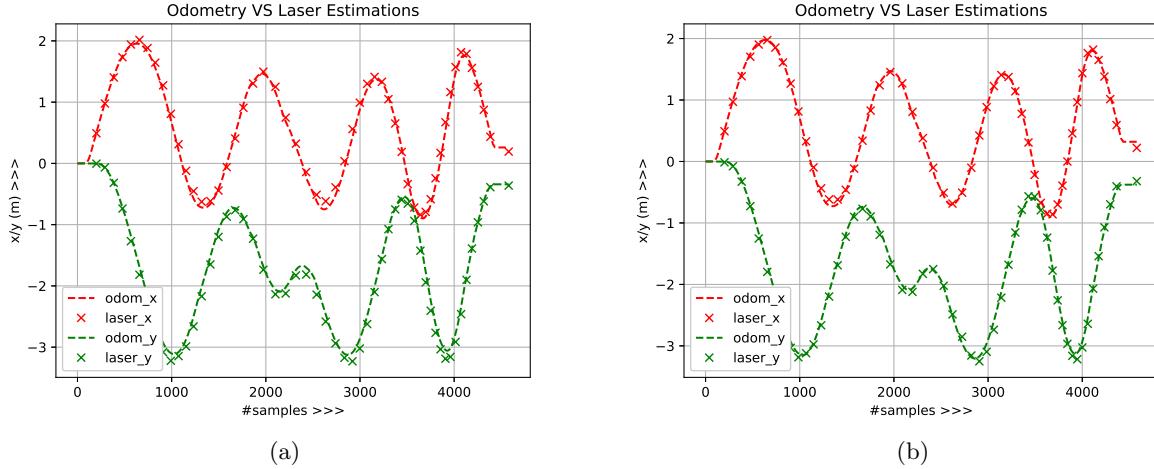


Fig. 6. Odometry VS Laser Matching Relative to the Experiment $i = 2$: (a) Initial Estimation; (b) Calibrated Parameters

the position errors improved slightly with the calibrated parameters. This result is more evident if you look to the extremes of x and y data over time.

Furthermore, a brief analysis of repeatability can be performed over the calibrated parameters. Even though we have only 5 experiments, the purpose of this analysis is to evaluate the variation of the extrinsic parameters for different experiments with the same configuration (no changes over the real odometry and extrinsic sensor parameters). The following results are relative to the mean and standard deviation relative to each calibrated parameter:

$$-\mu_D = 0.154027\text{m}; \sigma_D = 0.001643\text{m}$$

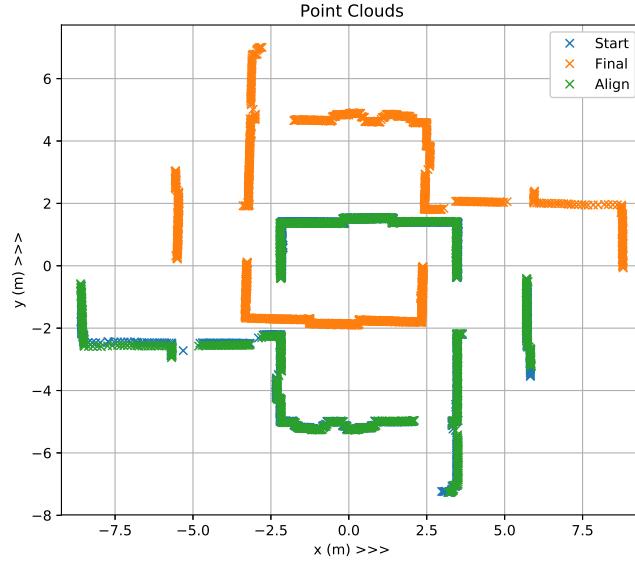


Fig. 7. 2D Laser Point Clouds Matching Using the ICP [9] Algorithm

- $\mu_l = 0.597001\text{m}$; $\sigma_l = 0.005458\text{m}$
- $\mu_{\alpha_{off}} = -56.720^\circ$; $\sigma_{\alpha_{off}} = 0^\circ$
- $\mu_{l_x} = 0.190444\text{m}$; $\sigma_{l_x} = 0.015346\text{m}$
- $\mu_{l_y} = 0.001185\text{m}$; $\sigma_{l_y} = 0.017820\text{m}$
- $\mu_{l_\theta} = -0.704^\circ$; $\sigma_{l_\theta} = 0.562^\circ$

Although we do not have ground-truth data for the odometry parameters, the experimental standard deviation results show that the proposed method can estimate the odometry parameters with high repeatability. Indeed, the standard deviation for D and l was lower than approximately 0.005m, and it was 0° for α_{off} . As for the extrinsic parameters of the laser scanner, the proposed method could not achieve the same results as for the odometry parameters in terms of standard deviation. Given that the laser matching estimation depends on the ICP performance, it could be related to matching errors. In figure 7, it is possible to evaluate an example for point clouds matching where we want to match the *Final* to the *Start* point clouds (the matching cloud is represented by the *Align* one). The *Start* and *Final* point clouds would represent the homogeneous matrices $\mathcal{P}^{R,i}$ and $\mathcal{P}^{R,j}$, respectively. Even though it seems to have achieved a good matching, if we look closer to the area of $x \in [-10, -5]$ and $y \in [-4, 0]$, it seems that the *Align* point cloud has still a misalignment in terms of rotation. So, the parameters of the ICP [9] algorithm should be modified to improve the point clouds matching performance.

Lastly, note that it was considered only a maximum of 200 iterations for the DIRECT [10] global optimization algorithm. Given that the calibration problem presented in this work has 6 dimensions (equivalent to the number of parameters to be estimated), the maximum number of iterations could be lower than desired. However, the execution of the calibration algorithm with 200 iterations requires approximately 1 hour to estimate the calibrated parameters. So, it was not feasible to increase this number.

5 Conclusions and Future Work

Wheeled odometry and laser scan matching can be used to obtain localization estimations for the robot. However, each of these localization methods requires knowing a certain set of parameters. In the case of a tricycle robot, wheeled odometry depends on the diameter of the front-driven wheel, the distance between the front and rear wheels, and the steering angle offset. As for a 2D laser scanner, even though it would be required 6 DoFs to describe the sensor's relative pose to the robot's frame, it could be simplified into 3 DoFs (x and y offsets in the robot's local frame, and the orientation angle of the laser relative to the robot's frame) if we considered that the sensor is horizontally mounted on the robot.

This work proposes a calibration algorithm for estimating both odometry and extrinsic sensor parameters simultaneously. A cost function was formulated that compares the linear displacements estimated by wheeled odometry

and laser scan matching (the latter obtained with the ICP [9] algorithm). The cost function was optimized using the DIRECT [10] global optimization algorithm. The preliminary results presented in this report showed that indeed the proposed calibration reduces the error between odometry and laser scan matching. The output of the work presented in this report was the development of a ROS node that will be integrated in the INESC TEC Navigation Stack for mobile robots. This node will allow the automatic calibration of wheeled odometry and 2D laser scanner systems. Previously, the calibration was performed by an expert operator requiring at least 1 hour to conclude the procedure. It should be noted that the experimental results presented in this report showed that the proposed method achieves higher odometry and laser matching convergence than the manual procedure.

As future work, it will be evaluated the repeatability of the parameters' estimation with the proposed method with more datasets (e.g., other types of paths and more experiments), higher number of maximum iterations for the DIRECT [10] algorithm, different configurations of the laser (change the real relative pose of the laser), and possibly with other steering geometries (differential drive and omnidirectional). Given that having access to ground-truth data of the odometry and extrinsic sensor parameters could not be possible, the repeatability analysis would evaluate if the proposed algorithm estimates the intended parameters with a minimal variation. Also, further improvements to optimize the computational performance of the proposed algorithm will be made to reduce the time required to obtain the calibrated parameters. One of the improvements could be task parallelization for the optimization algorithm.

Acknowledgments

I thank the Centre for Robotics in Industry and Intelligent Systems (CRIIS) from INESC TEC – Institute for Systems and Computer Engineering, Technology and Science for making available the mobile robotic platform used in the experiments. Also, I thank the mobile robotics team from CRIIS for all the support given throughout the development of the work presented in this report.

References

1. M. Spong, S. Hutchinson, M. Vidyasagar, *Robot Modeling and Control*, 1st edn. (John Wiley & Sons, 2005)
2. J. Borenstein, L. Feng, *Measurement and correction of systematic odometry errors in mobile robots*, IEEE Transactions on Robotics and Automation **12**(6) 869–880 (1996). DOI 10.1109/70.544770
3. C. Gao, J. Spletzer, *On-line calibration of multiple LIDARs on a mobile vehicle platform*, in *2010 IEEE International Conference on Robotics and Automation* (2010), pp. 279–284. DOI 10.1109/ROBOT.2010.5509880
4. A. Censi, A. Franchi, L. Marchionni, G. Oriolo, *Simultaneous Calibration of Odometry and Sensor Parameters for Mobile Robots*, IEEE Transactions on Robotics **29**(2) 475–492 (2013). DOI 10.1109/TRO.2012.2226380
5. F. Kallasi, D. Lodi Rizzini, F. Oleari, M. Magnani, S. Caselli, *A novel calibration method for industrial AGVs*, Robotics and Autonomous Systems **94** 75–88 (2017). DOI 10.1016/j.robot.2017.04.019
6. F. Galasso, D. Lodi Rizzini, F. Oleari, S. Caselli, *Efficient calibration of four wheel industrial AGVs*, Robotics and Computer-Integrated Manufacturing **57** 116–128 (2019). DOI 10.1016/j.rcim.2018.11.005
7. J. Underwood, A. Hill, T. Peynot, S. Scheding, *Error modeling and calibration of exteroceptive sensors for accurate mapping applications*, Journal of Field Robotics **27**(1) 2–20 (2010). DOI 10.1002/rob.20315
8. N. Li, T. Luo, B. Su, *Fast Extrinsic Calibration for 3D LIDAR*, in *Proceedings of the 2019 4th International Conference on Automation, Control and Robotics Engineering* (Association for Computing Machinery, New York, NY, USA, 2019), no. 15 in CACRE2019. DOI 10.1145/3351917.3351986
9. P. Besl, N. McKay, *A method for registration of 3-D shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(2) 239–256 (1992). DOI 10.1109/34.121791
10. D. Jones, C. Perttunen, B. Stuckman, *Lipschitzian optimization without the Lipschitz constant*, Journal of Optimization Theory and Applications **79**(1) 157–181 (1993). DOI 10.1007/BF00941892
11. R.B. Sousa, M.R. Petry, A.P. Moreira, *Evolution of odometry calibration methods for ground mobile robots*, in *2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)* (2020), pp. 294–299. DOI 10.1109/ICARSC49921.2020.9096154
12. R. Sousa, M. Petry, A. Moreira, *Extrinsic sensor calibration methods for mobile robots: a short review*, in *CONTROLO 2020*, ed. by J. Gonçalves, M. Braz-César, J. Coelho (Springer International Publishing, Cham, 2021), pp. 559–569. DOI 10.1007/978-3-030-58653-9_54
13. R.B. Sousa, Odometry and extrinsic sensor calibration on mobile robots. Master's thesis, Faculty of Engineering of the University of Porto (FEUP), INESC TEC – Institute for Systems and Computer Engineering, Technology and Science, Porto, Portugal (2020). URL <https://doi.org/10.13140/RG.2.2.27052.28802>
14. B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, *Robotics: modelling, planning and control*, 1st edn. (Springer-Verlag, London, 2009). DOI 10.1007/978-1-84628-642-1
15. R. Rusu, S. Cousins, *3D is here: Point Cloud Library (PCL)*, in *2011 IEEE International Conference on Robotics and Automation* (2011), pp. 1–4. DOI 10.1109/ICRA.2011.5980567
16. S. Johnson. The NLOpt nonlinear-optimization package. URL <http://github.com/stevengj/nlopt>. Accessed on 2021-05-28