Hybrid Interval-Probabilistic Localization in Building Maps

Aaronkumar Ehambram[1], Luc Jaulin[2] and Bernardo Wagner[1]

***Abstract*— We present a novel online capable hybrid interval-probabilistic localization method using publicly available 2D building maps. Given an initially large uncertainty for the orientation and position derived from GNSS data, our novel interval-based approach first narrows down the orientation to a smaller interval and provides a set described by a minimal polygon for the position of the vehicle that encloses the feasible set of poses by taking the building geometry into account using 3D Light Detection and Ranging (LiDAR) sensor data. Second, we perform a probabilistic Maximum Likelihood Estimation (MLE) to determine the best solution within the determined feasible set. The MLE is converted into a least-squares problem that is solved by an optimization approach that takes the bounds of the solution set into account so that only a solution within the feasible set is selected as the most likely one. We experimentally show with real data that the novel interval-based localization provides sets of poses that contain the true pose for more than $99\,\%$ of the frames and that the bounded optimization provides more reliable results compared to a classical unbounded optimization and a Monte Carlo Localization approach.**

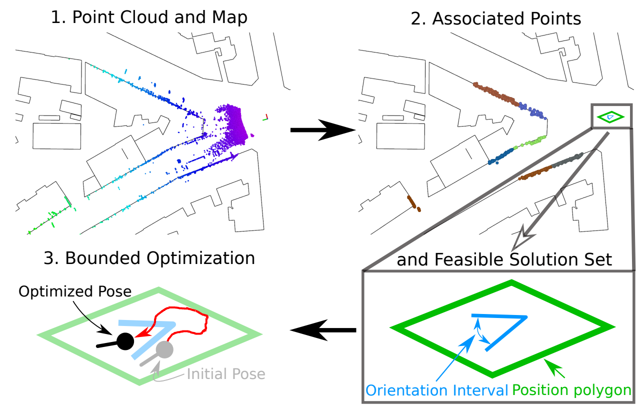


Fig. 1: Overview. From the point cloud, lines are extracted and associated with the facades of the map. We narrow down the feasible set of poses based on the association. The set of positions is described by a polygon and the orientation by an interval. Using a bounded optimization method, we find the most likely solution within the feasible set.

I. Introduction

Localization is one of the core abilities of mobile robotic systems to perform navigation. During the last three decades, there have been a large variety of different solutions. The approaches differ – among other things – in terms of the map, the used sensors and how the measurement uncertainties of the noisy sensors are handled. Uncertainties can either be described probabilistically [1]–[6] or by set-membership approaches [7], [8].

Probabilistic localization methods study the propagation of probabilistic distributions of the sensor noise and map uncertainty to the pose estimation. Teslic et al. [1] solve the localization problem using an Extended Kalman Filter (EKF) – a classical probabilistic approach. The linearization makes the method inherently local [8] and incorrectly distorts the uncertainty estimate of the localization. Also the approximation of all the involved uncertainties by a Gaussian distribution introduces modeling errors.

Monte Carlo Localization (MCL) approaches overcome these problems: The probability density function is represented by maintaining a set of samples that are randomly drawn from it. Hentschel et al. [2] perform MCL with LiDAR sensors in publicly available 2D building maps by tightly integrating GPS information, wheel odometry and inertial data to the sampling procedure. While Hentschel et al. use cadastral maps and OpenStreetMap (OSM) building maps, Floros et al. [3] weight the samples using the road network extracted from OSM road data. However, the major drawback of MCL approaches is that the quality of the solution heavily depends on the number of samples. If the uncertainty is very large, a large number of particles may be required to cover the solution space and therefore can become computationally heavy. Further, due to random sampling of the particles, an unfortunate sequence of samples can cause wrong convergence of the method [9].

In contrast to filtering techniques, Maximum Likelihood Estimation (MLE) approaches seek to find the most likely solution given the measurements and the map. For instance, Vysotska et al. [4] incorporate the constraints drawn from the locally observed facades of the buildings and the OSM building map into a graph optimization problem. Similarly, Boniardi et al. [5] track the pose of a robot in an architectural floor plan using 2D LiDAR solving a graph optimization problem. Wilbers et al. [6] track the vehicle location by matching pole-like landmarks. Although the presented approaches provide accurate results, the authors do not comment on the uncertainty estimates of the poses. As shown in [10], we see two major problems in MLE approaches: First, linearization is necessary for error propagation and leads to approximation errors that are not taken into account. Second, systematic errors that are not modeled lead to erroneous results while the uncertainty estimates of the results can be

This work was supported by the German Research Foundation (DFG) as part of the Research Training Group i.c.sens [RTG 2159].

[1]Real Time Systems Group (RTS), Institute for Systems Engineering, Leibniz Universität Hannover, D-30167 Hannover, Germany. {ehambram,wagner}@rts.uni-hannover.de

[2]ENSTA Bretagne, Lab-STCC, UMR CNRS 6285, Brest, France. lucjaulin@gmail.com

significantly underestimated. Although there are certifiable optimization techniques such as presented in [11], the uncertainty propagation based on linearization is still error-prone for non-linear localization tasks. As a result, these methods lack integrity. We try to compensate for this issue by fusing the MLE with an interval approach as illustrated in Fig. 1.

In contrast to probabilistic approaches, intervals do not need any prior knowledge about the probability distribution. Interval methods do not aim to provide the most accurate solution as probabilistic methods do. Interval methods rather dismiss infeasible parts in the solution set and provide a set that guarantees to contain the true solution if none of the assumptions are violated. Kieffer et al. [8] present the first interval-based localization approach that determines the position and orientation of a robot equipped with ultrasonic sensors given a map containing oriented segments as landmarks. The presented method only considers static localization and is not online capable. The high computational effort mainly results from a recursive set inversion strategy (SIVIA). We bypass this issue by directly taking the bounds into account and we present a novel method that can provide a minimal position polygon. Further, our method is online capable. Langerwisch et al. [7] present an approach for localization in a structured indoor environment using wheel odometry and a 2D LiDAR. The authors solve a Constraint Satisfaction Problem (CSP) by applying a robust version of SIVIA. In contrast to this approach, we seek to find a minimal polygon that encloses the position as illustrated in Fig. 1 instead of an axis-aligned box. We perform the localization in an outdoor environment where more objects occlude the buildings. Further, we use a 3D LiDAR that provides more measurements.

Hybrid approaches combine probabilistic and interval methods. Ashokraj et al. [12] combine the EKF with the approach presented in [8] by correcting the EKF solution if it lies outside the box estimated by the interval approach. Neuland et al. [9] use the interval-based localization results to determine the feasible solution set and generate particles within the sets – similar to MCL – to determine the best solutions.

In this paper, we present a hybrid interval-probabilistic approach – as presented in Fig. 1. The method fuses our novel interval-based localization approach, that provides a minimal polygon for the position and a narrow interval for the orientation of the vehicle, and an MLE approach, that finds the most likely solution within the feasible rigidly bounded solution set by solving a least-squares problem. Our contributions in this paper are two-fold. We aim to (i) determine the vehicle pose described by a minimal position polygon exploiting building geometries and the interval orientation; (ii) augment the set-membership approach by an MLE method that seeks to find the best solution from the feasible set.

II. Interval Analysis

In the following, a short introduction to interval analysis is given based on [13]. Interval analysis is a part of set theory extending real arithmetic operators to intervals. An interval $[x] = [\underline{x}, \bar{x}]$ is defined by its lower bound $\underline{x}$ and upper bound $\bar{x}$. When expressing measurements using intervals, we assume rigid and guaranteed bounds of the measurements. This kind of uncertainty description is intuitively stated by us humans when we for example talk about a distance that is accurate up to $\pm 1\,\text{m}$. In contrast to probabilistic approaches, we are not able to access the most likely value but we can guarantee the enclosure of the true value x^* by the interval so that $x^* \in [x]$ holds.

Classical real arithmetic operators, as well as elementary functions, can be extended to interval arithmetic utilizing inclusion functions. The goal of our work is to estimate the pose of the robot taking the building map into account. Hence, we seek to find the pose estimate $\mathbb{X}$ based on the measurements $\mathbb{Y}$ by solving the set inversion problem

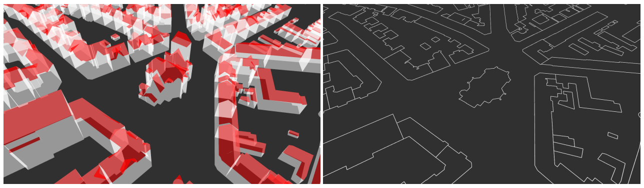
$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\} = \mathbf{f}^{-1}(\mathbb{Y}), \tag{1}$$

where $\mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^2$ represents a function that describes the expected measurement $\mathbb{Y}$ given the robot pose $\mathbb{X}$. To solve (1), a CSP can be formulated: Starting with an initial interval for $\mathbb{X}$, interval computations are employed to discard infeasible parts based on $\mathbb{Y}$. Hence, the set of poses $\mathbb{X}$ is contracted to a set that is consistent with the constraints – which is the basic idea of contractor programming.

III. Task Description, Notations and Assumptions

A. Task Description

The problem addressed in this paper is to determine the set of all feasible poses $\mathbf{p} \in \mathcal{P}$ and the most likely pose $\mathbf{p}^* \in \mathcal{P}$ of the vehicle in a given map with building footprints using a 3D LiDAR. The pose $\mathbf{p} = \begin{pmatrix} t_x & t_y & \theta \end{pmatrix}^T$ consists of two translation and one orientation parameters describing the pose in 2D. A set of poses $\mathcal{P} = \{\Phi, [\theta]\}$ is described by a polygon Φ that defines the set of translations and an interval $[\theta]$ that defines the set of orientations. We describe Φ by an ordered set of corner points $\mathcal{C} = \{c_1, \dots, c_n\}$ with $c_i = \begin{pmatrix} x & y \end{pmatrix}^T$ for $n, i \in \mathbb{N}$ and $x, y \in \mathbb{R}$, so that the border of Φ is defined by the line segments between the corner points. A feasible position $\begin{pmatrix} t_x & t_y \end{pmatrix}^T \in \Phi$ is defined by a point that lies within or on the border of Φ. Note that if the bordering line segments are axis oriented and the polygon Φ simplifies to a rectangle, the polygon can be represented by two intervals so that $\begin{pmatrix} [t_x] & [t_y] \end{pmatrix}^T \subseteq \Phi$ holds. That means the polygon can be described by an interval vector (box) and the tools provided by interval analysis are applicable. We assume that the initial set of poses of the vehicle is known with a large uncertainty. That means the initial orientation interval $[\theta_{\text{init}}]$ and the position polygon Φ_{init} are large. The higher the initial pose uncertainty is, the more local measurements can be matched to facades in the vicinity. As it will be explained in Section V, the proposed method in this paper is not able to cope with ambiguous localization results. Hence, we assume the initial pose uncertainty to be large, but small enough to unambiguously match local data to map facades. Further, without loosing generality we assume



(a) 3D City Model. (b) Footprints of the buildings.

Fig. 2: Building Map. The CityGML LOD2 model is shown in Fig. 2a. The according footprints are depicted in Fig. 2b. OSM provides maps similar to the one presented in Fig. 2b.

$\Phi_{\text{init}} \supseteq \begin{pmatrix}[t_{x,\text{init}}] & [t_{y,\text{init}}]\end{pmatrix}^T$ to be a box.
A polygon is minimal with respect to a set $\mathcal{S}$, if there exists no other polygon that has a smaller area for a finite number of corners. The goal of our interval-based method is to determine the minimal position polygon Φ and the orientation $[\theta]$ that enclose all feasible vehicle poses, given the map and a LiDAR scan. The goal of the bounded MLE is to find the most likely solution within the feasible sets.

B. Sensor Error Model

Due to various effects like different reflectivities of surfaces, humidity and imperfect calibration, the LiDAR measurements can be inaccurate. We claim intervals to be the appropriate tool to model the uncertainty, as such effects cannot be modeled explicitly. The detailed description of the LiDAR error model can be found in [14]. As a result, we model each measurement provided by the LiDAR by a box $[\mathbf{m}_l] = \begin{pmatrix}[x] & [y] & [z]\end{pmatrix}^T$ which is guaranteed to contain the truly measured 3D point.

C. Map

In this paper, we want to use the CityGML LOD2 city model [15] and OpenStreetMap (OSM) [16]. While the LOD2 city model is only available for dedicated cities, OSM provides worldwide geo-spacial data.
The LOD2 city model shown in Fig. 2a is a generalized 3D model of the buildings based on cadastral information and data from airborne laser scanning. As we use the building map for localization and due to the height of the buildings in our data sets, we are not able to constrain the elevation of the vehicle. For that reason, we simplify the LOD2 city model to the building footprints as shown in Fig. 2b and use that map for 2D localization.
OSM directly provides the building footprints. The major difference between the LOD2 map and OSM data is the accuracy: Although the authors of the LOD2 map do not give explicit values on the accuracy, it can be assumed to be close to the used cadastral data (standard deviations between 2 to 10 cm) with an unknown generalization error. Since OSM is open-source where different contributors are involved, no globally valid accuracies can be assumed. Nonetheless, our experience shows that we can provide reliable upper and lower bounds for the uncertainty for the used datasets. As a consequence, we assume a map that consists of building footprints with uncertainties for which we can give an upper and lower bound. A building is modeled by multiple facades where each facade forms a line segment in the map. A line segment is defined by a pair of points $\mathbf{a}_{m,1}$ and $\mathbf{a}_{m,2}$ described in the map frame m. We further represent the line by its implicit form

$$\mathbf{n}_m^T \cdot \mathbf{a} - d_m = \begin{pmatrix}\cos(\alpha_m) & \sin(\alpha_m)\end{pmatrix} \cdot \begin{pmatrix}x \\ y\end{pmatrix} - d_m = 0. \quad (2)$$

The orientation of the facade is described by α_m while the distance of the line is defined by d_m. As we assume interval uncertainty for the facades, each corner point inflates to a box and the line parameters also inflate to intervals. Consequently, in this paper the i-th facade in the map is fully described by $F_i = \{[\mathbf{a}_{m,1}], [\mathbf{a}_{m,2}], [\alpha_m], [d_m]\}$.

IV. System Overview

A. Sensor Hardware

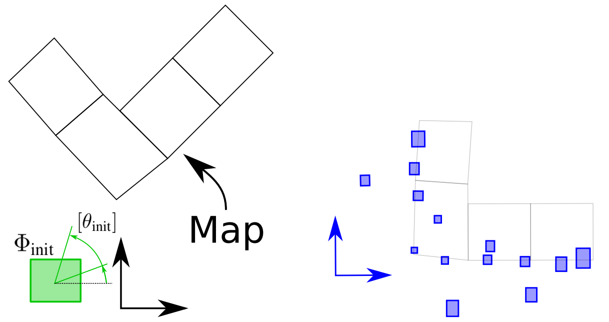
The proposed method is validated with real data on (i) an author-collected dataset acquired by the the same sensor system that was presented in [10], [14] and (ii) on the KITTI datasets [17]. The author-collected dataset provides LiDAR scans from a Cepton Vista 8800 solid-state LiDAR for the localization. GNSS measurements are used for the initial estimate of the vehicle pose. Ground truth measurements are obtained with the Riegl VMX-250 Mobile Mapping System. The KITTI datasets provide data from a Velodyne HDL-64E.

B. Method Overview

We assume the initial pose of the vehicle to be known up to a large uncertainty as explained in III. Our method consists of two steps: First, we narrow down the initial orientation and position uncertainty to feasible sets that are consistent with the map and local measurements provided by the LiDAR. For this purpose we use interval analysis.
Second, based on the sets we perform an MLE. Up to this step, we just assumed the measurement errors to be bounded. But from this step onward, we approximate all measurement uncertainties to be Gaussian distributed. This assumption leads to a least-squares optimization [18] that we solve by utilizing the Levenberg-Marquardt algorithm. As we manage to determine rigid bounds in the first step, we select an initial guess of the pose from the sets. We further modify the optimization so that for none of the updates the solution lies outside of the feasible sets.

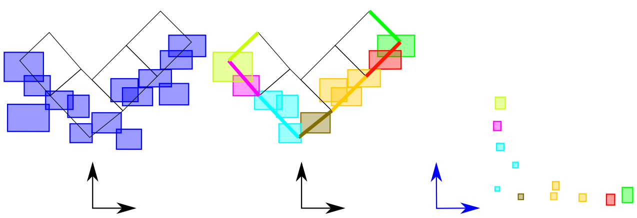
V. Interval-based Localization

The LiDAR provides 3D boxes as measurements. Since we assume a 2D building map, we first project the boxes to the floor by omitting the elevation axis. To determine which of the building facades are seen, we first perform an interval-based line selection. Based on the line selection we deduce which local measurements are associated with the facades. As facades consist of elements such as balconies, ornaments and projections we need to extract lines from the projected local LiDAR boxes. Therefore, we introduce a novel interval-based Hough Transformation (iHT) based on [19]. This method provides the local lines with interval uncertainty.



(a) Map and initial pose. (b) Local measurement boxes.

Fig. 3: Exemplary scenario. Fig. 3a shows a map and the initial pose estimate. Map components are colored black. Fig. 3b shows the local measurements which are colored blue.



(a) Transform to m. (b) Intersect with F_i. (c) Association in l.

Fig. 4: As shown in Fig. 4a, the boxes are inflated by the transformation from l to m due to the initial pose uncertainty. In Fig. 4b those boxes with a non-empty intersection with a facade are associated with it. Based on the association, Fig. 4c shows the associated local box clusters.

Based on the lines, we then contract the orientation and determine the position polygon.
For reader convenience, we introduce an exemplary scenario in Fig. 3 to visually describe the steps in our approach

A. Map Line Selection

In this step, we determine which of the local measurements can be potential boxes on the facades. Therefore, we first transform the local boxes to the map frame by using the initial estimate $[\theta_{\text{init}}]$ and Φ_{init}. As a result, the transformation for each local measurement box $[\mathbf{m}_l]$ is described by

$$\mathbf{m}_m = \begin{pmatrix} \cos(\theta_{\text{init}}) & -\sin(\theta_{\text{init}}) \\ \sin(\theta_{\text{init}}) & \cos(\theta_{\text{init}}) \end{pmatrix} \cdot \mathbf{m}_l + \begin{pmatrix} t_{x,\text{init}} \\ t_{y,\text{init}} \end{pmatrix}, \quad (3)$$

for $\mathbf{m}_m \in [\mathbf{m}_m]$, $\theta_{\text{init}} \in [\theta_{\text{init}}]$, $\mathbf{m}_l \in [\mathbf{m}_l]$ and $\begin{pmatrix} t_{x,\text{init}} & t_{y,\text{init}} \end{pmatrix} \in \Phi_{\text{init}}$. We determine $[\mathbf{m}_m]$ by applying interval operations in Eq. 3. Due to the size of the initial pose intervals, this operation will inflate the measurement boxes $[\mathbf{m}_m]$ in the map frame. For the scenario depicted in Fig. 3 the result of the transformation is shown in Fig. 4a.
As a next step, we check which of the transformed boxes $[\mathbf{m}_m]$ have a non-empty intersection with the line segments of the map facades. Therefore, for a given facade $F_i = \{[\mathbf{a}_{m,1}], [\mathbf{a}_{m,2}], [\alpha_m], [d_m]\}$ a box $[\mathbf{m}_m]$ that satisfies both

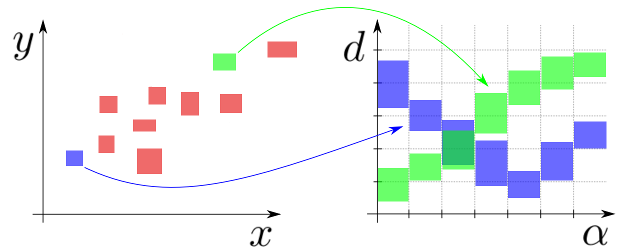


Fig. 5: Interval-based Hough Transformation. From a set of boxes, an accumulator is filled using the angle-distance line parametrization. Each angle interval $[\alpha]$ generates a distance interval $[d]$ for a measurement $[\mathbf{m}]$.

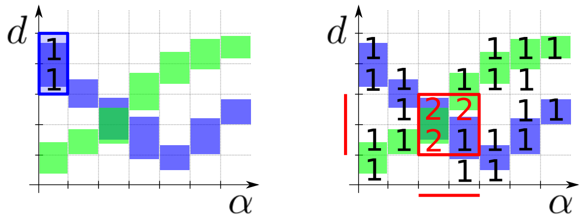
constraints

$$\begin{cases} \text{i)} \begin{pmatrix} \cos([\alpha_m]) & \sin([\alpha_m]) \end{pmatrix} \cdot [\mathbf{m}_m] \cap [d_m] \neq \emptyset \\ \text{ii)} ([\mathbf{a}_{m,1}] \sqcup [\mathbf{a}_{m,2}]) \cap [\mathbf{m}_m] \neq \emptyset \end{cases} . \quad (4)$$

has non-empty intersection with F_i. The first constraint defines that $[\mathbf{m}_m]$ has to intersect with the line spanned by the facade. The second constraint defines that $[\mathbf{m}_m]$ has to lie between the corner points of the facade segment. If a box $[\mathbf{m}_m]$ has a non-empty intersection with F_i, we can associate the corresponding measurement as a potential measurement of the facade F_i. In Fig. 4b the association of the intersecting transformed boxes is visualized. Those boxes that do not intersect with any of the facades can be ignored. Due to the association, we also know which local measurement is associated with which facades in the map (see Fig. 4c). One measurement can be associated with multiple facades. As a result, to each potentially seen facade in the map, we generate an associated box cluster.

B. Interval-based Hough Transformation

The above-described procedure may associate measurements to the facade that do not belong to the bare plane. For example balconies, ornaments and close objects may be associated. We need to extract a line segment with the highest support from each associated box cluster. Furthermore, we need to consider the uncertainty of the extracted lines using intervals. Therefore, we introduce a novel interval-based Hough Transformation (iHT) similar to [19].
A line is parametrized by the angle and the distance as described in Eq. 2. As a result, the Hough space is spanned by those two parameters and each measurement box contributes to the Hough space accordingly. Similar to the classical HT and in contrast to [19] we approximate the Hough space by a quantized accumulator as shown in Fig. 5. While the classical HT uses scalar values, our variables are described in the interval domain and applying Eq. 2 provides an interval for the distance $[d]$ for a given angle interval $[\alpha]$ and a given measurement $[\mathbf{m}] = \begin{pmatrix} [x] & [y] \end{pmatrix}$ (see right side of Fig. 5). A distance interval $[d]$ can partially overlap with a quant in the accumulator. To ensure an efficient accumulator representation we overestimate the uncertainty by taking the lowest upper bound and the highest lower bound according to the quantization that includes the interval as shown in



(a) Intervals in the accumulator. (b) Interval line extraction.

Fig. 6: Interval line with the highest support from the accumulator. Fig. 6a shows the approximation of the upper and lower bound of an interval. Fig. 6b shows the accumulator values. The interval line corresponds to the angle and distance intervals hull enclosing the highest accumulator values.

Fig. 6a. If the accumulator is filled for all the measurements of the cluster, we will get a similar accumulator as shown in Fig. 6b. Now we search for a set of lines that have the highest support. Therefore, we take the accumulator cell with the highest support v (high accumulator cell value) and consider all those cells that have an accumulator value higher than $0.95v$. In the accumulator, those cells will be close together for good line structures as illustrated in Fig. 6b. For bad configurations where the box cluster does not form a line-like structure, the peak in the accumulator will not be significant. That is why we only take clusters into account that have enough measurements and have a significant peak so that the $0.95v$ is 5 times higher than the average accumulator cell value. This selection procedure makes the line extraction robust against artefacts on the facades for real data as we only select those cells that have a high support. We then take the hull of all selected cells in the Hough space, so that we get an interval for the angle $[\alpha]$ and for the distance $[d]$ as illustrated in Fig. 6b. These parameters describe the set of lines with the highest support and that incorporate the uncertainty of contributing measurements.

We can restrict the region in the Hough space based on the initial pose estimate and the associated map line. This reduces computation costs as the accumulator can be set smaller. On the downside, we are implicitly deciding to which facade a box cluster has to belong. Consequently, ambiguous localization is suppressed. However, this approach works well in practice with real data, if the initial pose uncertainty is small enough so that the matching is not ambiguous.

As a result, for each box cluster the iHT determines the set of lines described by the angle $[\alpha_l]$ and the distance $[d_l]$ that have the highest support. Further, we can determine the subset of boxes that are consistent with this parameter set using the accumulator as we track which of the boxes contribute to which cell. Consequently, by choosing the two boxes $[\mathbf{a}_{l,1}]$ and $[\mathbf{a}_{l,2}]$ among the subset that have the highest distance to each other we can describe the line as a line segment – similar to the facade F_i. As a result, the iHT provides a local line segment $L_i = \{[\mathbf{a}_{l,1}], [\mathbf{a}_{l,2}], [\alpha_l], [d_l]\}$ that is associated to the facade F_i.

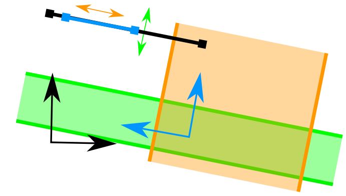


Fig. 7: Constraints provided by matched line segments. The map facade F_i is colored black, the local line segment L_i is blue. The position is constrained along the normal of the line segment (green) and along the line due to finite segment lengths (orange). Each line segment match provides two pairs of parallel lines that border the feasible position.

C. Orientation Contraction

For each associated pair L_i and F_i the orientation is determined by the difference between the line angles $[\alpha_m]$ and $[\alpha_l]$. For n associated pairs, the orientation interval is determined by

$$[\theta] = [\theta_{\text{init}}] \bigcap_{i \in \{1,\ldots,n\}} [\alpha_m] - [\alpha_l]. \tag{5}$$

Please note that the implementation of the operation has to consider the modulo-2π-problem and is omitted here due to lack of space.

D. Position Polygon

While the line angles $[\alpha_m]$ and $[\alpha_l]$ mainly determine the orientation of the local frame in the map frame, the distance of the local frame to the map frame origin along the normal vector of F_i is not affected. Consequently we can simplify the orientation interval $[\alpha_m]$ of F_i to a scalar orientation by taking the mid $\alpha_{m,\text{mid}}$. As a result, F_i is approximated by a set of parallel lines. This enables us to draw two geometric constraints for each associated line segment L_i and F_i as shown in Fig. 7. First, the position is constrained along the normal vector of the facade F_i (green). Consequently, the position $\begin{pmatrix} t_x & t_y \end{pmatrix}^T$ has to satisfy

$$\begin{pmatrix} \cos \alpha_{m,\text{mid}} & \sin \alpha_{m,\text{mid}} \end{pmatrix} \cdot \begin{pmatrix} t_x \\ t_y \end{pmatrix} = d_m - d_l = d_\perp, \tag{6}$$

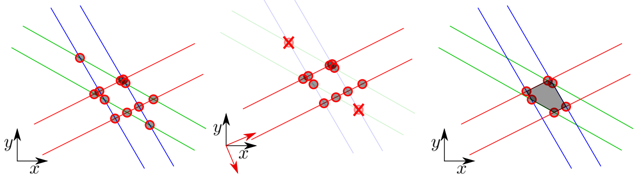
for unknown $t_x \in [t_x]$ and $t_y \in [t_y]$ and known $d_m \in [d_m]$, $d_l \in [d_l]$ and $d_\perp \in [d_\perp]$. That means the position can be bordered by an upper and lower line, that are parallel to the facade. The position has to be within the stripe as illustrated with a green stripe in Fig. 7.

Second, the finite length of the line segments constrains the position along the facade F_i (see Fig. 7, orange stripe). We define

$$[d_{g,\|}] = [\mathbf{n}_g^T] \cdot [\mathbf{a}_{g,1}] \sqcup [\mathbf{n}_g^T] \cdot [\mathbf{a}_{g,2}], \tag{7}$$

for $\mathbf{n}_g = \begin{pmatrix} \cos \alpha_g + \frac{\pi}{2} & \sin \alpha_g + \frac{\pi}{2} \end{pmatrix}^T$ and $g \in m, l$, $\alpha_g \in [\alpha_g]$ and $\mathbf{n}_g \in [\mathbf{n}_g]$ based on the corner points of the line segments L_i and F_i. The position has to further satisfy

$$\begin{pmatrix} \cos\left(\alpha_{m,\text{mid}} + \frac{\pi}{2}\right) & \sin\left(\alpha_{m,\text{mid}} + \frac{\pi}{2}\right) \end{pmatrix} \cdot \begin{pmatrix} t_x \\ t_y \end{pmatrix} = d_\|, \tag{8}$$



(a) Intersections. (b) Deletion. (c) Polygon.

Fig. 8: Determine the polygon from a set of stripes. Fig. 8a shows all intersections of the bordering lines of the stripes. By iteratively deleting the points if they do not lie within the stripes as illustrated in Fig. 8b, the polygon corners are determined as shown in Fig.8c, that define the borders of the feasible solution set.

for known $d_{\|} = d_{m,\|} - d_{l,\|}$, $d_{m,\|} \in [d_{m,\|}]$ and $d_{l,\|} \in [d_{l,\|}]$ and unknown $t_x \in [t_x]$ and $t_y \in [t_y]$. This constraint again introduces a stripe represented by an upper and lower line, that are perpendicular to the facade. As a result, each associated facade gives a stripe (pair of bordering lines) parallel and perpendicular to the facade (see Fig. 7). The intersection region of all the stripes represents the region of feasible positions.

We can formulate Eq. 6 and 8 for multiple facades as an interval linear equation system

$$\begin{pmatrix} \cos(\alpha_1) & \sin(\alpha_1) \\ \vdots & \vdots \\ \cos(\alpha_n) & \sin(\alpha_n) \end{pmatrix} \begin{pmatrix} t_x \\ t_y \end{pmatrix} = \begin{pmatrix} d_1 \\ \vdots \\ d_n \end{pmatrix}, \qquad (9)$$

where each facade contributes two rows and $t_x \in [t_x]$ and $t_y \in [t_y]$ are unknowns and $d_1 \in [d_1]$ to $d_n \in [d_n]$ are described by known intervals. In interval analysis, efficient tools exist to determine the smallest axis-aligned box $\begin{pmatrix}[t_x] & [t_y]\end{pmatrix}^T$ enclosing the solution space that satisfies Eq. 9 [13]. But we want to determine the exact, minimal polygon described by the corner points. Therefore, we propose a novel method illustrated in Fig. 8 exploiting the parallel border lines of each stripe.

Each row in Eq. 9 corresponds to a stripe that is bordered by an upper and lower line defined by the interval distance $[d_i]$. We first determine all possible intersections between the bordering lines as shown in Fig. 8a. These points are inserted into a set of potential corner points $\mathcal{C}$. As a next step, we select the first stripe and define a new coordinate frame that has the same origin as the map frame, but the x-axis is oriented along the normal vector $\begin{pmatrix}\cos(\alpha_i) & \sin(\alpha_i)\end{pmatrix}^T$ of the facade. The corresponding rotation matrix is fully determined by the angle α_i. We then transform all the points in $\mathcal{C}$ to this coordinate frame. Points lie within the stripe only if the x-component of the transformed points have a non-empty intersection with $[d_i]$. All those points that have an empty intersection, are deleted from $\mathcal{C}$ as illustrated in Fig. 8b. This procedure is performed for each row of the system 9 – that means for each stripe. The remaining set $\mathcal{C}$ defines the corner points of the polygon Φ as shown in Fig. 8c.

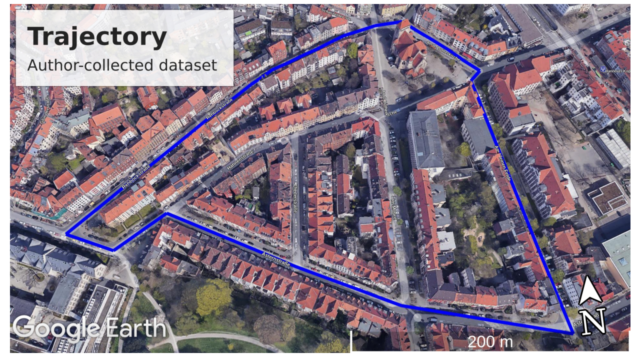


Fig. 9: Trajectory to the author-collected dataset T_1.

VI. Least Squares Optimization with Rigid Bounds

The interval-based localization provides an interval $[\theta]$ for the orientation and a polygon Φ for the position. Given the feasible solution set, the goal is to determine the most likely solution. To perform an MLE, we need to assume an error distribution. As the classical approaches do, from now on we also assume a Gaussian distribution for the errors. As a consequence, the MLE can be formulated as a least-squares problem [18]. The objective function, that we want to minimize for each measurement point of each facade is

$$f(\mathbf{p}) = \begin{pmatrix} \cos(\alpha_m) \\ \sin(\alpha_m) \end{pmatrix}^T \cdot \left(\mathbf{R}(\theta) \cdot \mathbf{m}_l + \begin{pmatrix} t_x \\ t_y \end{pmatrix}\right) - d_m, \quad (10)$$

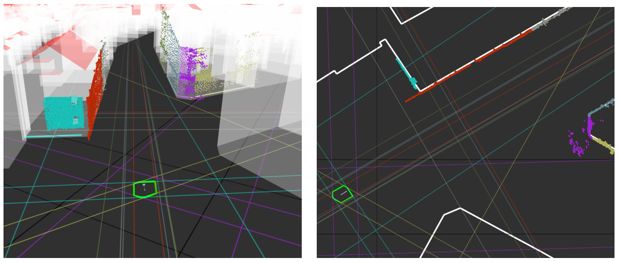
where $\mathbf{R}(\theta)$ defines the 2D rotation matrix and $\mathbf{m}_l$ a measurement point that lies on the facade F_i. The iHT provides the association between measurement and facade. We define $\mathbf{f}(\mathbf{p})$ as the function that stacks $f(\mathbf{p})$ of all measurements to all facades. Let the overall function be $F(\mathbf{p}) = \mathbf{f}(\mathbf{p})^T \cdot \mathbf{f}(\mathbf{p})$. The goal is to find a pose so that

$$\mathbf{p}^* = \arg\min_{\mathbf{p} \in \mathcal{P}} F(\mathbf{p}). \qquad (11)$$

As a consequence, all measurements are equally weighted. To solve the optimization problem we utilize the Levenberg-Marquardt algorithm. To account for the rigid bounds defined by Φ and $[\theta]$, we modify the update step: If an update leads to a pose that does not lie within the feasible solution set, we recompute the update with a higher damping factor. By doing that, we can assure for each step that the solution is always within the feasible solution set.

VII. Experiments

In this paper we present the test and evaluation results for two different datasets: The author-collected dataset was acquired in an urban region in Hanover, Germany. The corresponding trajectory T_1 is shown in Fig. 9 and measures a length of 1.25 km. The KITTI dataset 0027 [17] was acquired in a residential area in Karlsruhe, Germany. The trajectory has a total length of 3.71 km. The LiDARs were operated at 10 Hz for both datasets. Our method is implemented in C++ language using Robot Operating System (ROS). All computations are performed on a single consumer-grade



(a) Complex geometry 3D (b) Complex geometry 2D

Fig. 10: An exemplary configuration in shown in the 3D side view (Fig. 10a) and in the 2D bird's view (Fig. 10b). The extracted facade points using the iHT and the stripes corresponding to the facades are colored accordingly.

Method	Hybrid Localization		MCL
Encloses ground truth pose	99.8 %		–
Average radius of smaller side	0.91 m		–
Average radius of larger side	4.2 m		–
Average orientation interval radius	1.8°		–
Optimization	Bounded	Unbounded	–
Translation RMSE	**0.237** m	0.241 m	0.81 m
Average orientation error	**0.339°**	0.343°	0.75°
Largest translation error	**2.83** m	2.994 m	4.67 m
Largest orientation error	**2.921°**	5.935°	6.1°
Result outside feasible pose sets	0.0 %	0.725 %	–

TABLE I: Evaluation of T_1 with LOD2 Map.

laptop (Intel Core i7-9850H @ 2.6 GHz × 12, 64 GB RAM). Fig. 10 depicts the localization result for an exemplary scenario in T_1. The complex building geometry leads to differently oriented stripes so that the intersection reduces to a comparatively small area.

To evaluate the localization performance of our method, we need to quantify the accuracy. The orientation accuracy can be measured by the radius of the interval. To measure the size of the position polygon we fit an enclosing rotated rectangle with the smallest area. The width and the height approximate the position accuracy.

The localization is evaluated for the author collected trajectory T_1 with the LOD2 map (Table I) and OSM data (Table II). For the KITTI dataset 0027 we only use the publicly available OSM data (Table III).

For the LOD2 map, we bound the horizontal position uncertainty with ± 0.9 m and the facade orientation uncertainty with $\pm 3.5°$. The uncertainty of the OSM map can be

Method	Hybrid Localization		MCL
Encloses ground truth pose	99.8 %		–
Average radius of smaller side	1.79 m		–
Average radius of larger side	5.23 m		–
Average orientation interval radius	2.39°		–
Optimization	Bounded	Unbounded	–
Translation RMSE	**0.615** m	0.645 m	0.85 m
Average orientation error	**0.818°**	0.843°	0.83°
Largest translation error	5.03 m	8.137 m	**3.5** m
Largest orientation error	6.54°	6.543°	**3.92°**
Result outside feasible pose sets	0.0 %	4.178 %	–

TABLE II: Evaluation of T_1 with OSM data.

Method	Hybrid Localization		MCL
Encloses ground truth pose	100 %		–
Average radius of smaller side	1.94 m		–
Average radius of larger side	3.48 m		–
Average orientation interval radius	2.7°		–
Optimization	Bounded	Unbounded	–
Translation RMSE	**1.12** m	1.3 m	1.18 m
Average orientation error	**1.149°**	1.78°	1.2°
Largest translation error	4.4 m	6.5 m	**3.5** m
Largest orientation error	9.99°	81.7°	**2.4°**
Result outside feasible pose sets	0.0 %	11.7 %	–

TABLE III: Evaluation of KITTI 0027 with OSM data.

bounded by ± 1.8 m and $\pm 4.5°$ for the trajectory T_1 in Hanover. For the KITTI dataset we found larger inconsistencies with the corresponding OSM data. That is why we need to take higher uncertainties for this experiment into account which we bound with ± 2.5 m and $\pm 6.0°$. The initial pose is known up to an accuracy of ± 5 m and $\pm 10°$. The tables I, II and III show the results for different trajectories and maps. To evaluate the impact of the bounded optimization we compare the localization error between the optimization results for the bounded and unbounded case. We also compare our hybrid localization method with an MCL approach. For the MCL the number of particles is set to 400 and we initially distribute the particles uniformly in the initial pose intervals to maintain comparability between the methods. To compare the localization results we take the weighted average particle as a reference for each time step after the resampling step.

In the upper part of the tables the quantitative evaluation of the feasible polygon pose sets determined by our interval approach is presented. The determined sets enclose the ground truth poses for more than 99.8 % of the frames. The very rare case, that the ground truth is not enclosed by the determined set is mainly caused by the error-prone extraction of the local facades from the box clouds: Shifted line structures in the box clouds are identified in the iHT as local facades and are incorrectly matched.

The accuracy of the localization is represented by the average radius of the smaller and larger side of the oriented rectangles that enclose the polygons and the average radius of the orientation intervals. As expected, the uncertainty of the pose estimates with the OSM map is higher. Note that the average radius of the smaller side directly corresponds to the horizontal position uncertainty of the respective maps. The average radius of the larger side corresponds to the initial position uncertainty as the initial uncertainty defines the upper limit.

The localization results are depicted in the bottom part of the tables. A small percentage of the results of the classical unbounded optimization lie outside the feasible set. Inspecting the largest errors in those cases, we can see that our bounded approach prevents the most likely solution from significantly diverging from the correct solution. Although this happens for the high quality LOD2 map in very rare cases (0.725 %), we can see an improvement of the Root Mean Square Error (RMSE) for the translation and the average error of the orientation for the proposed bounded optimization. When the low quality OSM data is used, it is more likely that the unbounded optimization diverges as shown in Table II and III. We found that the divergence of the optimization is caused by mainly three factors: First, inhomogeneous distribution of the LiDAR points on different facades leads to unbalanced weighting in the optimization; second, incorrectly associated points; and third, offset errors in the map. Especially in the OSM data based experiments, large parts of the map suffers from a position offset leading to partially inconsistent buildings compared to local data. This is also the reason why the percentage of the results that lie outside of the feasible solution set for the unbounded optimization is the highest for those experiments.

The KITTI dataset has inconsistencies due to which we have to increase the uncertainty. As we cannot identify if those inconsistencies originate from error-prone ground truth poses or from deviations in the OSM data, we encounter higher localization errors in Table II compared to III. In all the experiments our method outperforms the unbounded optimization method and the MCL regarding the average error. But in Table II and III the largest errors are smaller for the MCL. We believe that this is due to the fact that we compare to the weighted average pose, where particles that are far off compensate for each other especially for low quality maps.

For those sections where no facades are seen, the initial pose estimate for the according time stamp can not be further narrowed down to a smaller set. This also applies to the MLE method since no data association is possible. As a result, for those sections the pose of the vehicle can only be tracked by odometry information or other methods like i-VIL SLAM [10].

A. Runtime

The proposed method has an average execution time of $0.098\,\mathrm{s}$. As we have a frame rate of $10\,\mathrm{Hz}$, our method is online capable. Note that the orientation contraction and polygon estimation only needs $44\,\mathrm{ms}$ while the map line selection with $0.041\,\mathrm{s}$ and the iHT with $0.046\,\mathrm{s}$ take the most time. In contrast to our method, the MCL with 400 particles needs $1.3\,\mathrm{s}$ for the same dataset and therefore is not online capable. Reducing the number of particles reduces the execution time, but the localization quality worsens.

VIII. Conclusions and Future Work

We present an online capable hybrid interval-probabilistic localization approach that uses LiDAR and publicly available 2D building maps. For this purpose, a novel interval-based localization approach is introduced. It provides an interval for the orientation and a minimal polygon for the position of the vehicle that represent the feasible set of poses taking the local LiDAR data and the map including their uncertainties into account. Furthermore, using an MLE we determine the most likely pose within the feasible set by applying a modified Levenberg-Marquardt algorithm that considers the rigid bounds provided by the interval method. We demonstrate that the bounded optimization is more reliable than the classical unbounded optimization. Compared to the MCL approach, our method provides lower average localization errors, provides feasible sets for the localization instead of a sampled representation and is online capable. On the downside, with the presented approach ambiguous localization in a global sense is not possible yet. The main issue with our current approach is the association of the local measurements to the map facades. In future work, we want to overcome this problem so that a global localization with possibly multiple disconnected solution sets is possible.

References

[1] L. Teslić, I. Škrjanc, and G. Klančar, "Ekf-based localization of a wheeled mobile robot in structured environments," *Journal of Intelligent & Robotic Systems (JINT)*, vol. 62, no. 2, pp. 187–203, May 2011.

[2] M. Hentschel and B. Wagner, "Autonomous robot navigation based on openstreetmap geodata," in *13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Sep. 2010, pp. 1645–1650.

[3] G. Floros, B. van der Zander, and B. Leibe, "Openstreetslam: Global vehicle localization using openstreetmaps," in *2013 IEEE International Conference on Robotics and Automation (ICRA)*, May 2013, pp. 1054–1059.

[4] O. Vysotska and C. Stachniss, "Exploiting building information from publicly available maps in graph-based slam," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 4511–4516.

[5] F. Boniardi *et al.*, "Robust lidar-based localization in architectural floor plans," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 3318–3324.

[6] D. Wilbers, C. Merfels, and C. Stachniss, "Localization with sliding window factor graphs on third-party maps for automated driving," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 5951–5957.

[7] M. Langerwisch and B. Wagner, "Guaranteed mobile robot tracking using robust interval constraint propagation," in *Intelligent Robotics and Applications*, C.-Y. Su, S. Rakheja, and H. Liu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 354–365.

[8] M. Kieffer *et al.*, "Robust autonomous robot localization using interval analysis," *Reliable Computing*, vol. 6, no. 3, pp. 337–362, Aug. 2000.

[9] R. Neuland *et al.*, "Hybridization of monte carlo and set-membership methods for the global localization of underwater robots," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2014, pp. 199–204.

[10] A. Ehambram *et al.*, "Interval-based visual-inertial lidar slam with anchoring poses," in *2022 IEEE 18th International Conference on Robotics and Automation (ICRA)*, (accepted) 2022.

[11] D. M. Rosen *et al.*, "Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group," *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 95–125, 2019.

[12] I. Ashokaraj *et al.*, "Sensor based robot localisation and navigation: using interval analysis and extended kalman filter," in *2004 5th Asian Control Conference (ASCC)*, vol. 2, July 2004, pp. 1086–1093.

[13] L. Jaulin *et al.*, *Applied Interval Analysis with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer London Ltd, Aug. 2001.

[14] A. Ehambram, R. Voges, and B. Wagner, "Stereo-visual-lidar sensor fusion using set-membership methods," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, Aug 2021, pp. 1132–1139.

[15] G. Gröger *et al.*, *OGC City Geography Markup Language (CityGML) Encoding Standard*, 2nd ed. Open Geospatial Consortium, 2012.

[16] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, Oct 2008.

[17] A. Geiger *et al.*, "Vision meets Robotics: The KITTI Dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[18] V. Kreinovich and S. Shary, "Interval methods for data fitting under uncertainty: A probabilistic treatment," *Reliable Computing*, vol. 23, pp. 105–140, Jul. 2016.

[19] L. Jaulin and S. Bazeille, "Image shape extraction using interval methods," *IFAC Proceedings Volumes*, vol. 42, no. 10, pp. 378–383, 2009.