AGV Global Localization Using Indistinguishable Artificial Landmarks

Davide Ronzoni, Roberto Olmi, Cristian Secchi and Cesare Fantuzzi

Abstract—In this paper we consider the global localization problem for an industrial AGV moving in a known environment. The problem consists of determining the pose of the vehicle without any prior information about its location. The vehicle is supposed to be equipped with a laser scanner that allows to measure the range and bearing of the vehicle with respect to a set of anonymous landmarks. A map with the positions of all landmarks in the environment is available to the localization system. We propose a novel algorithm for AGV self-localization based on landmarks identification that can take into account also false detections, very common in industrial environments. The pose is computed with a single scan (2D), without any sensor fusion. The performance of the proposed strategy is shown both by simulations and experiments on real industrial plants.

I. INTRODUCTION

The navigation system is a key component of a modern Automated Guided Vehicle (AGV) since it strongly determines its flexibility. A navigation system has to be able to resolve the so-called global localization problem. In other words, it has to be able to obtain the pose of the AGV in the plant it is moving in, with no a priori estimate, using only data gathered from the surrounding environment. This problem has to be solved when the AGV is switched on but also when the vehicle “gets lost” for some reasons (e.g. communication faults). Furthermore, an accurate solution of the global localization problem is of paramount importance for properly initializing the navigation filters (e.g. extended Kalman Filters).

The problem of global localization has a rich history in the mobile robotics literature (see e.g. [1], [2], [3]) and several localization strategies have been developed using several kinds of sensors (e.g. laser scanners, sonars, cameras). Many works (see e.g. [4], [5], [6], [7], [8], [9]) exploit natural features (like corners, walls and geometrical shapes) for determining the position of a robot. These approaches are very powerful but they require that a considerable part of the features does not change and that each part of the environment is sufficiently distinguishable.

In industrial applications, a considerable part of the features can indeed change (e.g. a pallet may occlude a landmark and/or change the geometry of the environment) and, furthermore, many parts of the environment can be very similar (e.g. corridors and racks) leading to some ambiguities in the recognition of a zone of the plant. Ceiling structures can partially avoid this problem but that each part of the environment is sufficiently distinguishable.

In industrial applications, a considerable part of the features can indeed change (e.g. a pallet may occlude a landmark and/or change the geometry of the environment) and, furthermore, many parts of the environment can be very similar (e.g. corridors and racks) leading to some ambiguities in the recognition of a zone of the plant. Ceiling structures can partially avoid this problem but that each part of the environment is sufficiently distinguishable.

Artificial landmarks can be designed in order to be uniquely identified by the sensor (see e.g. [6], [11]) and this greatly simplifies the localization task. Nevertheless, in a common industrial environment (e.g. a warehouse) a very large number of landmarks (even thousands) may need to be installed in order to properly cover the area the AGV is moving in and the identification of such a large number of landmarks technically impracticable.

In the majority of AGV systems, the robots are equipped with a laser scanner and the environment is populated by anonymous (e.g. indistinguishable) landmarks (typically reflecting surfaces) [12]. The technology described above provides a high accuracy in the position recovery even in a large-scale plant, differently from RTLS systems as RFID [13] or UWB [14]. The disadvantage of using a Wireless Sensor Network (WSN) for localizing a mobile robot is the relative low accuracy which is insufficient for Automated Guided Vehicles that move with high velocities (e.g. 2 m/s) and very close to human people or shelves. The accuracy can be increased only with a sensor fusion (see e.g. [15]) and moreover more than one anchor has to be recognized by the AGV. NAV200 Laser Positioning System from SICK AG ([16]) is a commercial system that solves the global localization problem but with a map limited to 32 reflectors. Thus a warehouse with thousands of reflectors has to be
subdivided into layers.

When switching on the system, the navigation unit receives as an input the initial position of the robot in the global map. Thanks to this information, during the motion of the AGV, the navigation system is able to correctly associate the landmarks that the scanner detects and to determine the position of the robot by triangulation. The initial pose of the AGV is inserted manually into the navigation system and it has to be re-inserted each time an AGV fails to determine its position and the robot gets lost.

The goal of this paper is to exploit the landmark infrastructure already present in AGV systems for solving the global localization problem and to make automatic the initialization procedure of the navigation system. This will make an AGV system more autonomous and capable of self-recovery after a localization error avoiding any human intervention and/or blocking of the system.

We consider an AGV moving in an industrial environment (e.g. a warehouse) and equipped with a laser scanner. A set of anonymous reflectors, easily detectable (but not identifiable) by the laser scanner, are placed at known positions in the environment. This scenario is illustrated in Fig. 1. For each detected reflector the scanner measures the range and bearing with respect to its local reference frame and a global map of the environment, where the position of the landmarks is indicated, is available. We will exploit a result from cartography ([17]) to develop a new algorithm for landmark identification and association. This strategy will be further extended in order to take into account data corruption which may be due either to detection of false landmark (e.g. a reflective surface which is not a landmark is detected by the laser scanner) or to the non-detection or a real landmark (e.g. because of obstruction or removal). Once the landmarks are properly associated and the corrupted data properly handled, the global position of the robot will be computed.

The paper is organized as follows. In Sec. II a formal description of the problem is given. Then, in Sec. III the matching task is considered for the simplified case when only measurement noise is considered. Inliers and outliers are taken into account in Sec. V. Experimental results both on a real plant and with simulations are presented in Sec. VI. Finally, in Sec. VII some conclusions are drawn and some future work is addressed.

II. Overview of the Problem

We consider a set \( N \) consisting of \( n \) artificial reflectors, namely a set of anonymous landmarks disposed in known locations in the environment. The robot is initially placed somewhere in the map but it lacks the knowledge of its whereabouts. The sensor used by the robot is a laser scanner that is able to detect a set \( M \) consisting of \( m \) reflectors. Elaborating the range and bearing information, it is possible to establish their position in a laser-centered coordinate frame. In general \( n \geq m \) since the robot will detect just a portion of the landmarks in the environment.

The global localization problem can be decomposed into two subproblems. The first one is the landmark matching (i.e., finding a one-to-one mapping between the \( m \) detected reflectors and the corresponding points in \( N \)). The second task is to determine the robot pose \( P = (X, Y, \phi) \) after the matching has been done, where \((X,Y)\) and \(\phi\) are the Cartesian coordinates and the orientation of the robot with respect to a global frame.

In order to find a correspondence between the set of landmarks detected by the robot and the landmarks in the map, we will exploit the Euclidean distance since it is a geometric relationship that is invariant with respect to translations and rotations [9]. Thus, given two landmarks represented in the laser frame, their distance is the same as that they have in the map. Obviously the coordinates of the landmarks obtained by the measurements are affected by an error due to the laser scanner precision. In practice the range and bearing errors must be accommodated when computing the matching of the two sets of points. In order to take into account the sensor precision, we consider a tolerance \( T \) when comparing the distance between couples of points. The value of \( T \) has to be chosen according to the maximum error in the sensor measurement. Larger values of this parameter cause an increase of corresponding sets of points, consequently an increase of algorithm iterations. Furthermore, it is possible that the scanner detects reflecting surfaces (e.g. pallets) which are not landmarks. These “fake reflectors”, that we will call outliers, have to be properly handled by the matching algorithm in order to avoid non sensible results. We will call inlier each detected landmark corresponding to a reflector in the map.

The initial disposition of the \( n \) landmarks in the map is a sort of art. The more the landmarks, the bigger is the set of detected reflector and, consequently, the easier is to determine the position of the robot. On the other hand, the outliers become harder to handle during the matching process. In the paper, we assume that the number and the position of the reflectors in the environment has been already determined. A map of the landmarks is available and, furthermore, the distance between any pair of landmarks is greater than the tolerance \( T \) in order to eliminate possible ambiguities in the landmark detection. We finally assume that the landmarks disposition allows to detect, from every position of the map, a number of landmarks sufficient to localize the AGV.
The online phase of the matching algorithm is described in Alg. 1 where, given a set \( A, \sharp A \) denotes its cardinality.

**Algorithm 1 Matching algorithm without outliers**

**Require:** Set \( M \) and Set \( N \) and \( M \\
1: Z = \text{maxdist}(M) \\
2: F = f(Z, N) \\
3: \text{if} \; \sharp F = 1 \; \text{then} \\
4: \quad \text{return} \; F \\
5: \text{else} \\
6: \quad \text{while} \; \sharp F > 1 \; \text{do} \\
7: \quad \quad \text{if} \; \sharp Z = \sharp M \; \text{then} \; \text{return} \; \emptyset \\
8: \quad \quad \text{end if} \\
9: \quad P = \text{select}(M) \\
10: Z = \text{append}(Z, P) \\
11: F = f(Z, N) \\
12: \text{end while} \\
13: \text{return} \; F \\
14: \text{end if}

The algorithm requires the set \( M \) of the detected landmarks together with their position in a laser-centered frame, the set \( N \) of the landmarks installed in the environment together with their position in a global coordinate frame and the data structure \( M \) computed in the offline phase.

Firstly (Line 1), the pair of landmarks in \( M \) characterized by the maximum distance are selected and stored in the list \( Z \). This operation is performed by the function \( \text{maxdist} \). If more pairs of landmarks are characterized by the same maximum distance, the function \( \text{maxdist} \) just picks one. The set \( F \) is then computed by means of the map \( f \) (Line 2) that is defined by:

\[
f : (P_1, \ldots, P_z, N) \mapsto \{(p_1, \ldots, p_z) \in N \times \cdots \times N \}
\]

where \( z = \sharp Z \). Loosely speaking, given a list of \( z \) landmarks in the laser centered reference frame, the function \( f \) determines the groups of \( z \) landmarks in the global frame whose pairwise distances are the same as those of the landmarks in the laser centered frame. The image of \( f \) is built performing a search in the data structure \( M \).

The cardinality of \( F \) is equal to the number of possible correspondences of the set of landmarks \( Z \) in the environment. In Fig. 2 there is an example of correspondence sets with \( \sharp F = 11 \) and \( \sharp Z = 2 \). Using the function \( \text{maxdist} \) a list \( Z = \{P_1, P_2\} \) has been built. Then in the global reference frame all the correspondence sets (couple of points) to \( P_1 \) and \( P_2 \) have been computed through \( f(Z, N) \). A matching between the detected reflectors and the landmarks in the environments is obtained when \( \sharp F = 1 \). In Fig. 2 it can be seen that if we consider only two reflectors, there are 11 possible correspondences.

If a pair of landmarks in \( M \) is sufficient for the matching then the corresponding pair of matched landmark is returned (Lines 3 - 4).

If \( \sharp F > 1 \) it means that considering two of the \( m \) detected landmarks is not sufficient for univocally determining the position of the landmarks in the global frame. Thus, the algorithm enters in the loop (Line 6) where at each step the list \( Z \) is augmented. Before starting the iteration, the algorithm checks whether some new measurements to consider are available (Line 7). If this is not the case, the algorithm returns the empty set, meaning that the algorithm has failed to match the detected reflectors. This is never the case when there are not outliers but it may be the case when the algorithm will be used for dealing with the presence of outliers in Sec V.

The function \( \text{select} \) chooses from the set \( M \) a new landmark that is not already in the list \( Z \) (Line 9). The algorithm in [17] at this step chooses \( P \) as the closest point to \( P_1 \) but in the proposed method simply the first point of the list \( M \) is chosen. The choice of the subsequent point influences the convergence speed of the algorithm, but it depends on the number of possible correspondences to the list \( Z \) in the environment, and it could not be a priori known.

The new landmark is then added to the list \( Z \) (Line 10) and the set of correspondences to the new list of landmarks is done through the function \( f \) (Line 11). In case \( \sharp F > 1 \) a new loop is done, otherwise algorithm exits the loop and returns the set of corresponding landmarks in the global reference frame (Line 13). \( \sharp F \) is decreasing at each step because each iteration considers more landmarks than in the iteration before.

In Fig. 3, when considering 4 landmarks, the cardinality of \( F \) is equal to 1. Thus there is only one correspondence set to \((P_1, P_2, P_3, P_4)\).

At each iteration, only the correspondence sets obtained in the previous iteration are considered for building the image of \( f \). In this way the computational burden of the algorithm is much lower than what it would be if starting from scratch.
In order to complete the matching, the landmarks of the set $M$ that haven’t been considered when the algorithm stops have to be associated to their corresponding landmarks in map. This is done by augmenting the list $Z$ with the remaining landmarks and by computing $f(Z, N)$.

IV. LOCALIZATION OF THE AGV

Once the matching problem has been solved, it is possible to exploit the information about the relative and the absolute position of the landmarks for computing the pose of the AGV with respect to the global reference frame. This is equivalent to computing the configuration of the AGV with respect to the global reference frame. Let $(x'_i, y'_i)$ be the coordinates of the $i^{th}$ landmark with respect to the laser centered reference frame and $(x_i, y_i)$ be the coordinates of the corresponding landmark (computed using Alg. 1) in the global reference frame. By simple computations, it can be seen that the following relation holds:

$$
\begin{bmatrix}
    x'_1 - y'_1 & 1 & 0 \\
    y'_1 & x'_1 & 0 \\
    x'_2 - y'_2 & 1 & 0 \\
    y'_2 & x'_2 & 0 \\
    \vdots & \vdots & \vdots \\
    x'_m - y'_m & 1 & 0 \\
    y'_m & x'_m & 0 \\
\end{bmatrix} =
\begin{bmatrix}
    \cos \phi & \sin \phi & X \\
    \cos \phi & \sin \phi & Y \\
\end{bmatrix}
\begin{bmatrix}
    x_1 \\
    y_1 \\
    x_2 \\
    y_2 \\
    \vdots \\
    x_m \\
    y_m \\
\end{bmatrix}
$$

In order to compute the pose of the robot that best matches (in a least square sense) the detected landmarks, we compute the Moore-Penrose pseudo inverse of the matrix $A$:

$$
H = \begin{bmatrix} \cos \phi & \sin \phi & X \\ \cos \phi & \sin \phi & Y \end{bmatrix}^T = A^+b = (A^T \cdot A)^{-1} \cdot A^Tb
$$

Visually, the MOORE-PEARSE pseudo inverse of the matrix A:

$$
H = \begin{bmatrix} \cos \phi & \sin \phi & X \\ \cos \phi & \sin \phi & Y \end{bmatrix}^T = A^+b = (A^T \cdot A)^{-1} \cdot A^Tb
$$

Once $H$ is known, it is possible to recover $b$ by simple trigonometry.

The accuracy of the robot pose estimation can be increased by considering that the range and bearing measurements are affected by an error that is different for each reflector. Once a correspondence between the detected reflectors and the landmarks in the global reference frame has been found, it is possible to discard the reflectors whose measurement errors are greater than a threshold as in Fault Detection and Exclusion algorithms in the field of GPS RAIM [18], and then determine the pose of the AGV.

V. HANDLING OUTLIERS

The localization method outlined in Sec.III and in Sec.IV cannot be applied to real cases yet. In fact, when an AGV is moving in an industrial environment (e.g. a warehouse), it is possible that the laser scanner detects reflecting surfaces (e.g. pallets) which are not landmarks. These outliers, have to be properly handled by the matching algorithm in order to avoid non sensible results. In fact, if outliers are considered as real landmarks, the matching algorithm may fail to converge or it may lead to a corresponding set of landmarks in the global map which is not coherent with the real position of the robot.

In this section we propose an algorithm, that exploits Alg. 1, which can detect the presence of outliers among the measurements and which can discard them for the computation of the matching.

Since outliers and inliers are identical from the laser scanner point of view, it is impossible to distinguish them from the measurements. Thus, in order to be able to detect the presence of the outliers, some geometric assumptions on the disposition of the landmarks are necessary. We define $k$ as the minimum number of detected inliers for which there exists a unique correspondence in the global reference frame independently of the position of the AGV. We assume that at least $k$ landmarks have always to be detected. This assumption is always satisfied in real plants because it simply means that the landmarks are installed in such a way that the robot can be always localized. Furthermore, we indicate with $N_o$ the number of outliers and with $m$ the number of detected reflectors and we assume that:

$$N_o \leq \frac{m}{k}
$$

This assumption is stronger than the previous one and it basically requires that there are not too many outliers among the detected reflectors. This is reasonable in practice. In fact, if there are too many outliers, the information content of the inliers is lost. Furthermore, the environment and the landmarks installation should be such to make the number of outliers much smaller than the number of inliers. From the experience of the AGV manufacturer we are collaborating with, the maximum amount of outliers can be estimated as 10%.

The algorithm that we propose is reported in Alg. 2. Besides the inputs required by Alg. 1, the algorithm requires the value $k$ which, similarly to $\mathcal{M}$, can be computed in an offline phase, after the landmarks have been installed in the environment, using Alg. 1.

Firstly (Line 1), a set $W$ consisting of $m$ values is created, where $m = \mathbf{z}^T \mathcal{M}$. Each element of the weight set $W$ is relative to a detected point and represent the probability that the
specific point is an outlier. Initially no informations on the probability that each point is an outlier are available, thus all the elements of $W$ are initially set to zero.

Then the algorithm enters in a loop (Line 2). At each iteration, the elements of $M$ are sorted in ascending order with respect to their weights\(^1\) and stored in a list $S$ (Line 3). The function choose picks the first $k$ elements in $S$ and stores them in another list $K$ (Line 4). At this point, the algorithm tries to achieve the matching by Alg.1 using only the measurements stored in $K$ (Line 5). If no outliers are stored in $K$, then a unique correspondence in the global map of the landmarks is obtained. Nevertheless, Alg. 1 tries to obtain the matching with as less measurements as possible and it may happen that, because of the outliers, a single correspondence with the global map is obtained. Thus, Alg. 2, besides checking that a unique correspondence has been obtained, checks also if all the measurements in $K$ can be correctly matched in the global map (Line 6). The function associate starts from $F$ and it returns true if all the measurements in $K$, but those used for obtaining $F$, can also be correctly associated to landmarks in the global map. If at least one outlier is in $K$, this is impossible and the function returns false. If the test in Line 6 is true, then the algorithm returns $F$ which is the correct matching (Line 7) for the set of measurements. If the test fails it means that at least one outlier is in the set $K$. Thus, the weight of all the elements of $K$ is increased of a fixed amount $\delta > 0$ (Line 9). The weight of a measurement is proportional to its probability of being an outlier. At each iteration, the algorithm picks the $k$ elements that have the lower weights, namely the lower probability of being outliers.

In case $N_o$ outliers are present in $M$, the worst case is when only one outlier per iteration is considered. In this case, recalling Eq. 4, the algorithm converges after $N_o + 1$ iterations. Differently from other outliers exclusion algorithms based on RANSAC [19], the proposed method does not require a predetermined number of iterations and it stops automatically when the correct solution is reached. This implies fewer iterations and less elaboration time in the AGV context, where few outliers are detected compared to the number of detected landmarks.

VI. EXPERIMENTAL TESTS

The algorithms proposed in the paper have been evaluated by two different kinds of experiments. The first set of experiments have been done in a real industrial context using real data. The second set of experiments consists of simulation tests carried on using a real, larger, map of landmarks installed on a real automatic warehouse.

In the real experiments, we used a layout containing 32 indistinguishable artificial reflectors, disposed on the pillars of a 3000 m\(^2\) area warehouse. For each of the 10 tests done, a number of $7 \sim 12$ reflectors were captured using a laser scan mounted on a AGV positioned in different locations. We use the Range Finder SICK NAV300 to recognize the reflector: the angular resolution is 0.1 degrees, the range systematic error is $\pm 10$mm and the range statistical error is $\pm 10$mm. The range and bearing data relative to each detected landmark were then sent to an Intel T2400 laptop for the computation of the vehicle localization using the algorithms proposed in the paper. The estimated position is compared with the manually extracted positions using standard statistical methods (Tab. I). On the AGVs will be mounted Beckhoff PC with Windows XP Embedded.

The Localization system got correct match and location, independently of the number of detected reflectors and of the presence of one outlier. It took at maximum 0.038 seconds and in most cases less than 0.01 seconds.

The aim of the second set of experiments is to validate the matching method for industrial environments larger and with more outliers. Simulated results are obtained using a real industrial reflector map with 450 reflectors disposed in a 50000 m\(^2\) area (Fig. 4). For each test, the AGV is put in a random location in the map and the set of landmarks detected consists of those within a maximum range of 50 m from the laser scanner. We have introduced a measurement error using the datasheet of the laser scanner that is actually mounted on the AGVs. The data collected are consistent with those detected in real applications.

We have considered five different scenarios: $S_1$, $S_2$, ..., $S_5$. For each scenario $S_i$, the robot is set in a random location and 10\(^i\) landmarks are detected. Different numbers of outliers have been considered for each $S_i$. For each scenario, at least the 80% of the detected reflectors are inliers; this is consistent with real world applications involving AGVs moving in automatic warehouses.

The total number of simulations is 2400. In all the simulations the algorithm proposed in the paper achieves

\[ \text{Position} \quad \text{Mean Error} \quad \text{Error Variance} \]

\[ \begin{array}{lll}
\text{Position} & 13.4 \text{ mm} & 21.8 \text{ mm} \\
\text{Orientation} & 1.72 \times 10^{-3} \text{ rad} & 1.17 \times 10^{-4} \text{ rad} \\
\end{array} \]

\[ \text{TABLE I} \]

LOCALIZATION ERRORS IN REAL TESTS.
The algorithm solves the robot wake-up problem and it can be used for the initialization of the navigation system of the AGV. Future work will deal with the extension of the work in this paper in order to exploit it for navigation of an AGV in the environment.

ACKNOWLEDGMENT

The authors wish to greatly acknowledge the company Elettric80 s.p.a. (www.elettric80.com) which supported this research.

REFERENCES


VII. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a new method for localizing an AGV endowed with a laser scanner and located in an environment populated of anonymous landmarks. We have proposed a landmark matching method that takes into account measurements errors and false detection, due to reflecting surfaces present in the environment. The strategy has been validated by experiments in real industrial environments and by simulation on real plants maps.

The algorithm reaches the correct solution in less than 0.3 seconds even when outliers are considered. The proposed algorithm solves the robot wake-up problem and it can be correct matching and localization. The computation times are reported in Fig. 5. It can be seen that the elaboration time remains small even in case only few landmarks are detected and some outliers are present. It can also be noted that the computational time decreases the bigger is the number of detected reflectors and that it strongly depends on the percentage of the outliers with respect to the detected landmarks.