Mobile Robot Localisation
Using Ultrasonic Sensor Signatures
and Fuzzy ARTMAP Clustering.

Spyros G. TZAFESTAS* and Konstantinos C. ZIKIDIS*.

Abstract Mobile robot positioning is usually based on odometry, in conjunction to a periodic recalibration method, in order to minimise the accumulated estimation errors. This work is a preliminary study of the use of ultrasonic sensor 'signatures' as an independent source of spatial information that can be used to correct the odometry estimation, in combination with fuzzy ARTMAP-type clustering. A car-like mobile robot equipped with an ultrasonic sensor ring tries to localise itself, utilising a fuzzy ARTMAP neural network, which identifies different 'signatures' or patterns of ultrasonic sensor readings and partitions the environment into small areas, according to the perceived patterns. The present study extends our previous work on local and global mobile robot navigation.

I. Introduction

The estimation of position plays a crucial role in many tasks in the field of mobile robotics. The most basic way of position estimation is the use of internal encoders on the wheels of the mobile robot (odometry). The robot calculates its position, taking into account the revolution and the diameter of its wheels, and also its geometry (whether it is holonomic or car-like, etc.). This method is cheap and easy to implement, has a fast response, and yields acceptable results at least for limited amounts of time. Due to these obvious advantages, almost all mobile robots rely on odometry for short term position estimation. However, due to various reasons like wheel slippage or uneven wheel diameter, error accumulates unboundedly and can reach unacceptable levels after a short time. Orientation errors play a more significant role, since an initially small orientation estimation error leads to a large translation error after some meters of straight travel.

It is clear that spatial information independent from the wheel encoders has to be fused with odometry data, in order to monitor the error and either reset the measuring system periodically (recalibration) or to keep the error bounded. Many techniques have been proposed, using for example various external sensors like ultrasonic, laser and

* National Technical University of Athens, Zografou Campus, Zografou 15773, Athens, Greece. E-mail addresses: tzafesta@softlab.ece.ntua.gr, kzikidis@softlab.ece.ntua.gr
optical (camera)[1][2], landmarks, or even active beacons and GPS [3]. Other approaches have tried to improve the dead reckoning estimation accuracy by fusing odometry and other dead reckoning methods, like gyroscope [4], or by making an internal measurement/calibration of the odometry system [5].

In the present work, a mobile robot tries to fuse odometry and information from an ultrasonic sensor ring. The readings from the ultrasonic sensors give a 'signature' of every place. Places with similar 'signatures' can be grouped together. In an initial phase, where the position measurement is assumed to be accurate (e.g., it is periodically reset manually), the robot wanders in its environment and performs clustering in the signature space (or classification) and grouping of places giving similar signatures (corresponding to the same signature cluster). In this way, the robot divides the environment into small areas, associating each of them with a signature cluster. The robot, exploiting the information gathered through the learning phase, can identify the area it is in, using the range readings from the ultrasonic sensor ring. This information is fused to the odometry system, in order to monitor the accuracy of the position/orientation estimation. Computer simulation results for a car-like mobile robot indicate the potentials of this approach, keeping the error bounded for extended runs, even when the error in the odometry measuring system is biased.

In [6], Kurz used a similar system, performing experiments with two learning classifiers, Kohonen's self organising feature map and RCE-Classifier (Restricted Coulomb Energy Classifier). The present study reports preliminary results on the use of the fuzzy ARTMAP-type neural network [7] as the core of the localising system, used to perform partitioning (clustering) in both the environment area and signature spaces, and associate areas with signature clusters. This work extends and is based on our previous work [8][9][10] on the use of separate neural network modules for local navigation, global navigation and map recall. In fact, the local navigation module is always active, in order to avoid collisions as the robot wanders in its environment.
II. The mobile robot model and the experimental environment

The robot model used in the simulation is like a small car, 10 length units long and 3 units wide. The maximum angle of the steering wheels is ±45°. The robot is equipped by a ring of 16 ultrasonic sensors, with a maximum range of 62 units and minimum range 4 units. As mentioned above, independently from the localisation system, a local navigation system is active at all times. This system uses a different set of sensors, and has been analysed in [8][9].

The experimental environment is depicted in Fig. 1, with the robot and the receptive field of the ultrasonic sensor ring, to compare the sizes. Its dimensions are 640×380 length units. The obstacles are located in such a way that will cause similar signatures in many areas of the environment, in order to test the ability of the sensor fusion system to distinguish between areas associated with the same signature cluster.

III. Preprocessing of the ultrasonic sensor ring range readings

In every time step, the sensor ring yields 16 range values. In order to use these values in the clustering algorithm, a simple preprocessing is performed: the sensor ring is undergone a virtual rotation, until it is directed into a known direction (rotation into a reference orientation). In this way, the effects caused by the robot rotation are eliminated and the classification of signatures is independent from robot orientation. Of course, this supposes that the robot orientation is known. During the initial phase the position estimation system is supposed to be accurate. After this phase, the virtual rotation uses the estimated robot orientation.

In [6], in order to make the signature classification even more autonomous, a second preprocessing method was also used, yielding similar results: rotation into the most occupied orientation. In this way, at every time step the ring is virtually rotated to the direction that seems most occupied by obstacles. This method was not tested in the present study.

IV. The fuzzy ARTMAP system

The fuzzy ARTMAP algorithm [7] is a neural network architecture for incremental supervised learning of recognition categories. It includes two fuzzy ART modules, ART₁ and ART₂ that are linked together via an inter-ART module called map field. Each of the fuzzy ART modules receives a stream of input patterns and automatically creates recognition categories (or hyperboxes or clusters or classes). These recognition categories start as points in the input space and increase in size to incorporate new points that are presented to the input (while the respective weights representing the hyperboxes decrease), until the whole input space is covered. The maximum size of these hyperboxes and implicitly the number of the required hyperboxes can be adjusted.
by a parameter called "vigilance". A similar parameter exists also in the map field module. ART_a and ART_b are associated with the environment and the ultrasonic signatures respectively. The dynamics of this kind of network are far too complex to be analysed in this paper and the reader is referred to [7].

Figure 2. The fuzzy ARTMAP system has partitioned the environment, allocating all available categories (700 in these experiments). Areas with the same greyscale level are associated with the same ART_b category.

During the initial (learning) phase, the input to ART_a is the actual position of the robot (which is supposed to be more or less accurate), namely the vector \((x, y)\) where \(x\) and \(y\) are the robot position coordinates. ART_a automatically partitions the environment into areas (rectangles), the maximum size of which is controlled by the baseline vigilance \(\rho_a\), which is the initial value of \(\rho_a\). The input to ART_b is the ultrasonic sensor range pattern \((s_1, s_2, \ldots, s_{16})\), where \(s_i\) is the reading from the \((i)th\) sensor, virtually rotated to the reference orientation. ART_b creates clusters (hyperboxes) in the signature space, the maximum size of which depends on the vigilance parameter \(\rho_b\). The map field module represents the association between the clusters of ART_a and ART_b. The key idea of the fuzzy ARTMAP algorithm is that the category creation in ART_a is controlled by the required accuracy in ART_b. Therefore, if the selected category in ART_a is associated with the wrong category in ART_b, match tracking is initiated: the value of the vigilance parameter \(\rho_a\) in ART_a is increased and a new search starts, until another category is selected which predicts the correct category in ART_b. This category can possibly be a new one (previously uncommitted).

During the main phase, there is no input presented to ART_a. ART_b is presented with the ultrasonic sensor range pattern, virtually rotated to the reference orientation. ART_b comes up with a selected category (signature cluster). Since there was a learning phase, this is supposed to be an already known category, and not a new one. The map field module indicates a number of ART_a categories (areas) associated with the specific
ARTb class. In practice, with the environment of Fig. 1, in most cases there is a many-to-one relation between ARTa and ARTb classes.

V. Odometry-Ultrasonic sensor fusion: The error correction algorithm

Suppose that $x$ and $y$ are the actual robot position coordinates and that $x_n$ and $y_n$ are the robot position coordinates estimated by the odometry system. The ultrasonic sensor range pattern has been presented to ARTb which has selected a category. According to the map field module, a number of ARTa classes predict this ARTb class. In other words, a similar ultrasonic sensor reading pattern has been observed in a number of known areas. The next step is to choose the one that is closest to the estimated robot position (in terms of Euclidean distance). Suppose that this is the area corresponding to the class with index $i$. Suppose also that the coordinates of the center of this area is $a_{i1}$ and $a_{i2}$. At every time step, the position estimation values $x_n$ and $y_n$ are corrected according to the following rules:

\[
\begin{align*}
\text{if } |a_{i1}(t) - x_n(t)| < 50 \text{ then } x_{n}(t+1) &= c_{gain} \cdot a_{i1}(t) + (1 - c_{gain}) \cdot x_n(t) \\
\text{if } |a_{i2}(t) - y_n(t)| < 50 \text{ then } y_{n}(t+1) &= c_{gain} \cdot a_{i2}(t) + (1 - c_{gain}) \cdot y_n(t)
\end{align*}
\]

where $c_{gain}$ is the correction gain taking values in $(0,1)$ and adjusts how much the estimated value will approach the center of the chosen area. The condition in the first part of each rule examines whether the center of the selected area is far away, in which case it is considered as a possibly wrong selection and is ignored.

In this way, the position estimation is monitored and corrected at every time step. However, a correction in the orientation estimation is also necessary. This is a somehow more difficult task, because there is no independent information about the true orientation directly available, as in the case of the position. The orientation correction has to be calculated from two consecutive robot positions. For the position of the robot, the estimated and corrected coordinates are used. The robot calculates a better estimate of the orientation (supposed to be closer to the actual one) taking into account the speed, the angle of the steering wheel and of course the geometry of the robot. If $f_n$ is the orientation estimated by the odometry system and $f_c$ the orientation calculated from two consecutive estimated and corrected robot positions, the orientation estimation is corrected according to the following rule, in a way similar to the position correction rules:

\[
\text{if } |f_{c}(t) - f_n(t)| < 25 \text{ then } f_{n}(t+1) = c_{gain} \cdot f_{c}(t) + (1 - c_{gain}) \cdot f_n(t)
\]

The position and orientation corrections are neither smooth, nor lead the estimated position $(x_n, y_n)$ and orientation $f_n$ always towards the actual position $(x, y)$ and orientation $f$, unless the category classes in the fuzzy ARTMAP system are infinitely many. However, the corrections are based on information independent from the odometry measurement that does not drift with time. When applied with a small gain (e.g., with value between 0.05 and 0.1) it keeps position and orientation errors within bounds, even for extended runs and for noisy and biased odometry measurements.
Table 1. The parameters of the fuzzy ARTMAP system.

<table>
<thead>
<tr>
<th>Module</th>
<th>Parameter</th>
<th>Value</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>ART&lt;sub&gt;a&lt;/sub&gt;</td>
<td>choice parameter: ( a_a )</td>
<td>0.1</td>
<td>A small but positive choice parameter favours the choice of &quot;smaller&quot; categories.</td>
</tr>
<tr>
<td></td>
<td>baseline vigilance: ( \rho_a )</td>
<td>0.95</td>
<td>Such a large baseline vigilance value does not allow large category size (forcing the partition of the environment into small areas).</td>
</tr>
<tr>
<td></td>
<td>learning rate: ( \beta_a )</td>
<td>0.1</td>
<td>Such a small learning rate allows smaller overlap and better category allocation.</td>
</tr>
<tr>
<td></td>
<td>max. number of categories</td>
<td>700</td>
<td>During the learning phase, almost all available categories are committed.</td>
</tr>
<tr>
<td>ART&lt;sub&gt;b&lt;/sub&gt;</td>
<td>choice parameter: ( a_b )</td>
<td>0.0</td>
<td>This is the so called conservative limit. If the problem allows a choice parameter with the value of zero, we have minimum recoding.</td>
</tr>
<tr>
<td></td>
<td>vigilance: ( \rho_b )</td>
<td>0.7</td>
<td>This is a moderate value, leading to efficient allocation of categories in the signature space.</td>
</tr>
<tr>
<td></td>
<td>learning rate: ( \beta_b )</td>
<td>1.0</td>
<td>This value causes &quot;one shot&quot; learning: learning is fast and stable.</td>
</tr>
<tr>
<td></td>
<td>max. number of categories</td>
<td>60</td>
<td>During the learning phase, almost all available categories are committed.</td>
</tr>
<tr>
<td></td>
<td>map vigilance: ( \rho_{ab} )</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>field learning rate: ( \beta_{ab} )</td>
<td>0.9</td>
<td></td>
</tr>
</tbody>
</table>

VI. Computer simulation

In the learning phase, the robot wanders in the environment of Fig. 1 (the local navigation module guides the robot to randomly assigned virtual targets). The fuzzy ARTMAP system performs clustering in both spaces of environment areas and signature patterns, and associates areas with signature classes. An efficient category creation would look like a uniform allocation of small environment areas, each of them predicting a different category in the signature space. However, practice shows that this is not possible. ART<sub>a</sub> categories cannot be allocated uniformly, because there may be large regions where the same signature is perceived (for example in free space), while as the robot approaches obstacles, the perceived ultrasonic range pattern changes rapidly, leading to a more dense allocation of environment areas. On the other hand, a high vigilance value in the ART<sub>b</sub> module would trigger the creation of much too many categories in the ART<sub>a</sub> module.

Practically, a fast and stable category allocation in the ART<sub>b</sub> module helps the ART<sub>a</sub> module to allocate categories efficiently. Therefore, the robot is allowed to
perform a brief exploration of its environment with only the ARTb module active, before the activation of the full ARTMAP system. In this way, the ARTb module has allocated a few categories in the signature space and exhibits a more stable behaviour when the ARTa module is switched on.

The parameters in each fuzzy ART module are: the choice parameter $\alpha$ which takes usually small, positive values and favours the selection of 'smaller' categories, the vigilance parameter $\rho$ taking values in $[0,1]$ which adjusts the maximum size of the categories, and the learning parameter $\beta$, also taking values in $[0,1]$, which adjusts the learning speed. The parameters used in the fuzzy ARTMAP and some remarks on the selection of their values are shown in Table 1.

In the beginning, the robot is left to run freely, until the behaviour of the ARTb module seems to stabilise (the rate of category allocation has dropped). Then the full ARTMAP system is turned on. The ARTa module begins partitioning the environment into small areas. After a while (1 minute on a Pentium-class PC), almost all available categories have been used in both fuzzy ART modules. The learning phase has come to its end and the localisation system is ready. In Fig. 2, one can see an environment partition to 700 categories (areas).

It is reminded that during this initial phase the robot is supposed to have accurate position estimation. This requirement can be relieved, taking into account that only the center of each ARTa category is required in the main phase. The coordinates of the center can be roughly estimated with the use of an averaging method, taking into account all measurements relative to the corresponding ARTa category. However, this technique was not tested in the simulation.

In the main phase, at every time step a normally distributed random variable is added to the odometry estimation values (position/orientation), multiplied by a factor of 0.2. A bias of 0.01 is also added to the x-coordinate and the orientation parameter $f$. The correction gain has a small value between 0.05 and 0.1, as mentioned before. Using these values, the position estimation error without any correction grows very rapidly, as can be seen in the left part of Fig. 3.

With the proposed localisation system turned on, the robot was left to run for a long time and the difference between the actual and estimated position and orientation is
observed. The difference decreases when the robot passes through a 'higher information' region, where there are many ART\textsubscript{a} categories predicting different ART\textsubscript{b} categories, and increases when the robot is in regions where many ART\textsubscript{a} categories predict the same ART\textsubscript{b} category (for example in free space). During the whole time, the error remains bounded, as indicated (for a very small travel) in the right part of Fig. 3.

VII. Conclusion

A localising system based on ultrasonic sensor information and the use of a fuzzy ARTMAP was proposed and tested by computer simulation, yielding promising results. The robot is equipped with an ultrasonic sensor ring and odometry encoders. During a learning phase, the fuzzy ARTMAP system partitions the environment into areas associated with similar ultrasonic sensor readings. After this initial phase, the ultrasonic sensor reading pattern is fed to the fuzzy ARTMAP system, which comes up with a number of the corresponding environment areas. A correction algorithm, fuses this information to the odometry data, correcting the robot position and orientation estimates. Simulation results indicate that with the help of the proposed system the error between the actual and estimated position/orientation values remain always bounded.

Future work will concentrate on further study on this idea, experimenting on the use of the localising system right from the beginning, without any learning phase, possibly by employing a more effective sensor fusion algorithm, like Kalman filtering.

VIII. References


