A rewritable RFID environment for AGV navigation

Arne Bosien, Marcus Venzke, Volker Turau
Institute of Telematics
Hamburg University of Technology (TUHH)
Hamburg, Germany
{arne.bosien|venzke|turau}@tuhh.de

***Abstract*—The paper analyses potentials, challenges, and applications of using rewritable, stationary RFID tags for marking routes for autonomous vehicles in manufacturing environments. This approach promises to be flexible, robust, and inexpensive. Two algorithms are proposed also considering additional sensors.**

I. Introduction

Automatic Guided Vehicles (AGVs) are robots moving freely in manufacturing environments or warehouses require a means of navigation. Frequently transportation routes are marked with optical lines or electric wires embedded into the floor. More recently, electric wires are replaced by RFID tags returning static IDs in defined positions. All approaches share the inflexibility if routes have to be changed or require to determine the exact locations of all RFID tags.

This paper proposes to apply rewritable RFID tags for marking AGV transportation routes for increasing flexibility. It analyses potentials and applications, proposes two suitable algorithms, one incorporating an additional sensor, and points out challenges faced when implementing the idea. Furthermore, we describe a first prototype.

II. Navigation techniques

One approach for free-ranging is satellite navigation in combination with radio communication. Vehicles detect their own position by means of GPS and send it to a server via WLAN. The central control system takes care for computing the path an AGV has to follow and for management of all vehicles.

Another flexible solution is the localisation using Cartesian guidance with a grid build of optical or magnetic nodes. Every time a vehicle reaches a node it adjusts heading and speed to fulfil the received instructions [1]. The management is also done by a central authority.

Up to now RFID tags are commonly used as mobile devices to track or identify objects. This can be observed frequently when tags are attached as replacement for traditional bar-codes for clothing or containers etc.

Beyond this it is possible to use RFID tags stationary which can be utilised for localisation issues. Every tag can be identified clearly and has a limited range. With the aid of a map with positions of each tag, the position can be determined precisely. For blind humans this is applied to guide in Laveno Mombello (Italy) [2]. Alternatively the position can be stored directly on the tag avoiding the need for a map.

This technique has been introduced as *Free Ranging On Grid* (FROG) at the *Europe Combined Terminals* (ECT) in Rotterdam [3] and is also used at the *Container Terminal Altenwerder* (CTA) in Hamburg. For that purpose about 12,000 tags are spread in the CTA's pavement [4]. Since it is required to determine the geographic position of each tag, this environment is very complex to install.

Researchers at ETH Zurich have suggested a similar approach. They advocate *massively-redundant* tag distributions where RFID tags are deployed in a highly redundant fashion over large areas or objects surfaces [5]. They consider the possibility to leave a virtual trace on a RFID tagged floor space but focus on applications where they use *virtual tags*. A virtual tag is adopted to solve the problem of limited storage capacity and the feasibility that a tag might be readonly. Therefore, a background infrastructure is introduced which maps the ID of each *physical tag* to a virtual tag.

Like the previously described technologies this leads to the requirement of a reliable backend which must not fail.

III. Approach

Using a rewritable RFID environment for AGV navigation is inspired by ants using pheromones to mark ant trails. Marking takes place after an ant has discovered a source of food on its way back to the ant-hill. This allows other ants to follow the path to the food.

A similar approach can be applied for AGVs that move in an environment containing many rewritable RFID tags (e.g. as a grid embedded into the floor). Commercially available tags have about a hundred registers that can be written with an RFID reader attached to the AGV. Markers written into nearby tags are used similar pheromones used by ants to mark paths. This also allows storing additional information relevant for other AGVs.

In a manufacturing environment a human operated car can be used to mark paths for AGVs. While driving on a desired path, the car writes markers into the tags allowing automatic vehicles to follow later. In the same environment several paths are possible by writing different markers. This offers alternative routes and enables varying paths for different classes of vehicles.

The approach provides a lot of flexibility. If the production line is altered, new routes can be established with low effort. Overlapping paths are possible if desired. There is no need for maintaining a map of all tags. Wireless interfaces for communication with servers are superfluous; enabling its adoption in areas where radio fails or is prohibited. Having a distributed architecture only depending on the reliable communication with RFID tags makes the approach highly robust. It is inexpensive, because of decreasing tag prices and the simple installation by spreading out tags randomly without storing their locations.

Storing additional data on RFID tags during the production process enables additional usage scenarios. AGVs can follow a human operated car constituting a virtual train. This includes the communication of current speed to avoid crashes with the vehicle in front. If an AGV detects strong pollution, a damaged roadway, or anther vehicle that has broken down, it may mark the section as closed. The same may be done if it detects a failure in its own system before shutting down. Other vehicles can securely drive past or use an alternative route.

The approach also allows controlling the placement of AGVs on parking spaces or at service points. Before leaving the path, a vehicle marks the position as taken by writing a marker on the nearest RFID tag. Other vehicles then know that the position is occupied and continue to the next.

IV. Challenges

The major challenge is developing algorithms that allow an AGV to follow a marked path without loosing it. The mechanism needs to work reliably even in case of crossings caused by loops or crossing paths.

Another challenge is to deal with the limited storage space in RFID tags. Sophisticated algorithms thus need to minimise their memory consumption. One approach is based on forecasting the pursued path. Affording multiple paths is a further characteristic, that has to be faced.

If paths are marked during the production process, it is desirable to implement a kind of ageing allowing to forget old paths and freeing tags from old data. Alternatively algorithms are required for explicitly deleting old paths. Of course ageing must not apply to permanently required paths.

Requirements to be covered subsequently are enabling fault-tolerance in case of RFID tag failures and smooth movements. Furthermore, the number of read and write operations should be minimised allowing a high speed of AGVs.

V. First Algorithm

This section explains a first sketch of an algorithm for marking several paths (teaching mode) and for following it by AGVs (following mode). It is assumed that AGVs and vehicles for teaching mode have an RFID reader as the only sensor. The motor control allows an imprecise estimation of the resulting robot movements.

In the teaching mode a human controlled vehicle drives along the desired path. It marks it on the RFID tags by writing a path identifier and an increasing sequence number on the group of tags newly observed at discrete points in time (see figure 1). The sequence number is initialised to 0. The vehicle performs an inventory operation on the RFID reader for discovering the RFID tags in range and adds a marker consisting of path identifier and sequence number on each tag. The IDs of all tags are stored in a ring buffer of size b. The vehicle then moves on, increases the sequence number and repeats inventory operation and writes markers. Now markers are only written to tags

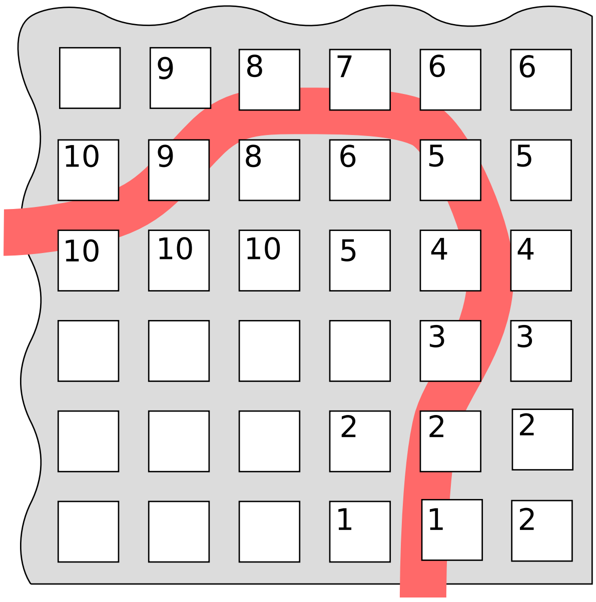

Fig. 1. Single path of robot in RFID environment: only sequence numbers are shown, the path identifier has been left out

not contained in the ring buffer. This is continued until the end of the path. Finally, a marker informing about the end is written to all tags found in an inventory operation.

In the following mode an AGV is asked to navigate on one of the marked paths. It is assumed that the vehicle knows the path identifier and is placed in a position where it is able to read at least one tag of the path. The AGV starts with an inventory operation and determines the highest sequence number s_0 with matching path identifier. In each step it then searches for a sequence number s_{i+1} with $s_i < s_{i+1} < s_i + m$ where m is a small integer greater than 1. m is the number of sequence numbers that can be skipped allowing faster movements. It must not be too high; otherwise the AGV might erroneously bend off if the path crosses itself or continue with another segment of the path that is close to the current position. For searching the next sequence number, the AGV performs inventory operations after moving forward as the first try, then searches to the left, right, and also in the opposite direction if necessary, until a suitable tag is found. If no more tag contains a marker of the path, the area of searching is increased. The AGV stops when the end marker is found.

The main advantage of this approach is its requirement for only one sensor, the RFID reader. Several crossing paths are supported using path identifiers. A path can cross itself if the intermediate section is large enough, that is, if the difference of sequence numbers is greater than m.

The disadvantages result from the lack of information of the AGV's orientation, relative position to the path, and the imprecise estimation of movements. In case of loosing the path searching requires driving to the left or to the right, not allowing a smooth correction of direction. The AGV might also need to drive backward to get back to the path. As a result, the path is not followed directly, but with a lot of search movements. Another disadvantage is the huge amount of records read from the tags, each requiring an individual, time consuming read operation. Both disadvantages decrease the maximum driving speed considerably.

Also a high tag density is required, otherwise the AGV cannot decide if it is off the path if no tag is reachable.

VI. Compass, Odometry, Additional Readers, and Dynamic Range

Additional sensors can be used to optimise the algorithm and to make it more reliable.

A. Compass

A compass allows to detect the current direction of the vehicle. This can be applied to reproduce rotary motions on the one hand. On the other hand algorithms can write directions on the tags. However, this consumes more space on the tags and thus reduces the number of markers that can be written.

A magnetic compass suffers from disturbances of the terrestrial magnetic field. In industrial environments this may be caused by steely objects or electrical machines. An alternative is the usage of a gyroscopic compass. These do not rely on the magnetic fields but maintain direction.

B. Odometry

Odometry enables to detect changes of the position and direction of a vehicle by analysing the movements of its wheels. In contrast to a compass this method cannot offer an absolute position nor absolute direction, but measurements relative to its previous position. Measurement errors are accumulated.

For example AGVs can use odometry to verify the distance they have to drive back if they have lost the path. In teaching mode it is possible not to write every tag but to mark tags in a certain distance.

Therefore, odometry offers a lot of advantages, but a precise realisation is much more complex than usage of a compass.

C. Dynamic range

If the reader uses constant transmission power, the robot doesn't get any information about the distance between reader and tags. This is useful for determining the deviation to the path to follow. For this purpose the reader increases the transmission power, does an inventory, and checks which tags have newly appeared. Readers allowing to vary the transmission power are on the market.

D. Additional readers

To distinguish the cases in which the vehicle is on the desired path, on its left, or on its right additional readers can be used. These may be mounted on both sides in the front of the robot.

Furthermore it can be reasonable to place a reader in the back of the AGV. While driving forward an algorithm can decide to store information after the tags in front have been passed. This allows the rear reader to write tag identifiers seen by the front reader.

VII. Optimised algorithm with compass

A second algorithm shall illustrate optimisations possible with a magnetic compass as additional sensor. It is an extension of the algorithm described in section V. In addition to the path identifier and the sequence number the teaching vehicle can also store the direction it is driving, as illustrated in figure 2). This allows an AVG to drive into the direction of tags with higher sequence numbers without searching.

The direction is only stored, when the driving direction is changed. Thus in the following mode, AGVs do not need to search for every tag, but follow the direction to the next. Because of the inaccuracy of the compass, AGVs can still loose the path requiring additional search operations.

To perform less write operations on tags, it is conceivable to only mark tags when changing the driving direction. Intermediate tags are skipped. This requires a precise estimation of the direction in teaching and following mode and poses high demands on the reproducibility of the robot's movements. The AGV cannot determine if it lost the path or the direction has not change yet. To avoid this every n-th tag can be marked.

VIII. Experimental platform

To evaluate the proposed concept and algorithms, first experiments with AGVs in a rewritable RFID environment have been conducted. The test environment consists of several mats with 100 writable RFID tags arranged in a grid as shown in figure 3. Two experimental AGVs have been build, each equipped with a single RFID reader and an electronic compass. One is connected to a PC by cable to simplify development, debugging and power supply. The other uses Bluetooth and batteries.

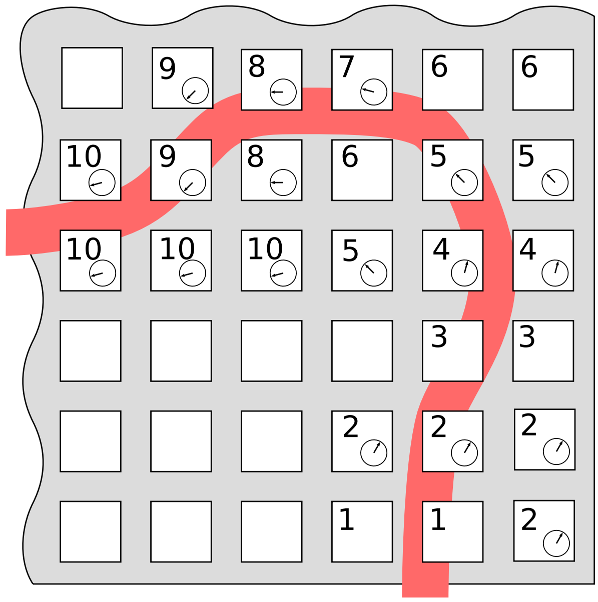

Fig. 2. Path of robot in RFID environment produced by second algorithm: the arrows show the direction written on the tag

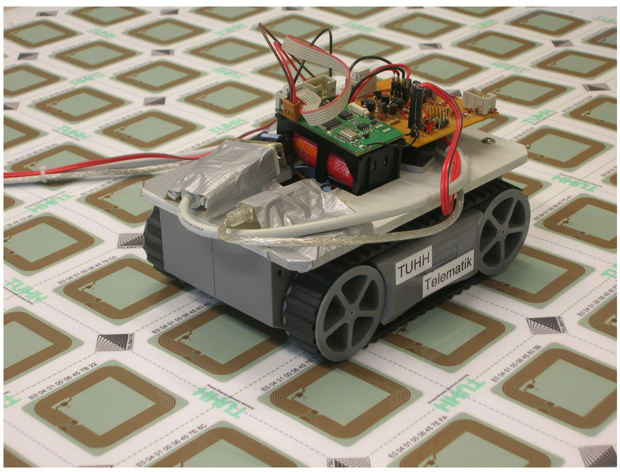

Fig. 3. AGV in RFID environment

The used RFID readers, integrated into the AGV's chassis, returns a maximum of four tags in a single inventory operation and supports a reading distance of up to 10 cm. It can be accessed via serial interface (see figure 4). Compass and motors are controlled by a microcontroller also accessed via serial interface.

A graphical interface for controlling AGVs (*teaching mode*) has been developed. It also provides a sim-

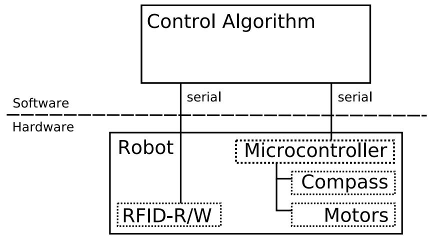

Fig. 4. Hardware setup

ulation environment for testing control algorithms. So far only teach/follow scenarios have been analysed.

The simulation software has been created to ease of development, testing and debugging. It allows the complete simulation of the hardware including RFID reader, compass, RFID tags, movement of the robot. It also simulates inaccuracies of the sensors and read/write errors of the RFID hardware. This allows to test algorithms without a physical robot. Teams having only one robot can work in parallel.

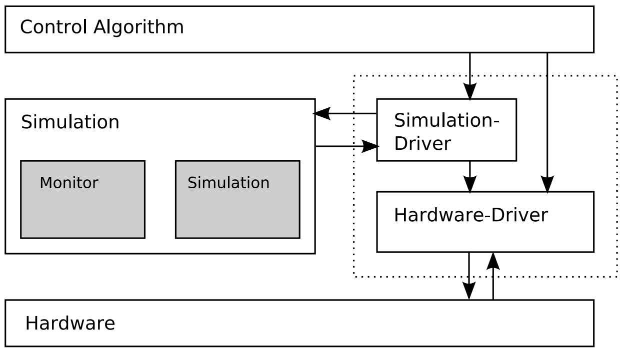

Fig. 5. Software Architecture

A driver concept as depicted in figure 5 is used to encapsulate the communication with the hardware. Drivers provide an API to access a robot. The architecture supports three usage scenarios:

A. Pure simulation

In pure simulation mode the simulation driver passes all request from the control algorithm to the simulator, which generates suitable responses with respect to the current state. As shown in figure 6 the simulator also displays tags that have been read or written and the location of the AGV in the RFID environment. To support this, the software can load an XML file with IDs and positions of the tags on the mat in use.

In the main area of the user interface the tag matrix and the simulated robot with its compass are displayed. The identifier and the contents of each tag are

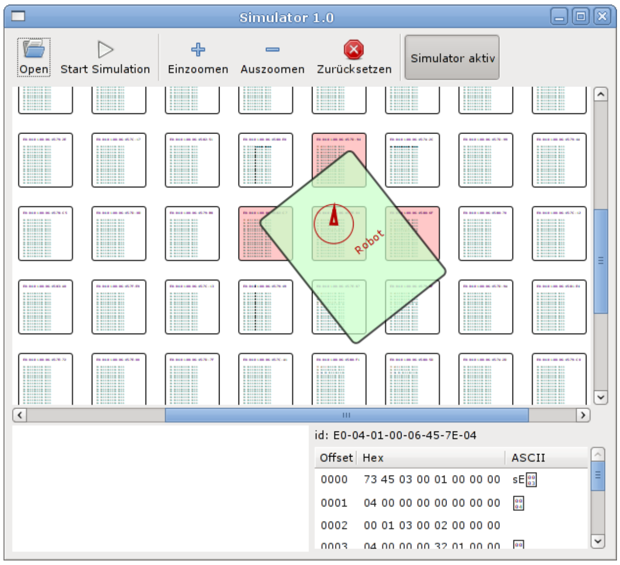

Fig. 6. Screenshot of monitoring interface

permanently shown but only readable if zoomed in. The currently accessed tags are highlighted. When a tag is selected, the content of this tag is also displayed in the lower screen area.

B. Monitoring

When testing an control algorithm with real hardware, it is useful to visualise the tags that have been read and the data that been written to a tag. The simulation driver forwards all requests to the hardware driver and relevant information is transmitted to the simulation software. The hardware driver communicates with the real robot and is not aware of the simulation software.

C. Native execution

To enable the control algorithm to run on a robot autonomously, the program can be build without simulator. Therefore, the API of simulation driver and hardware driver are identical.

The ability to abstract from the underlying hardware allows to compile identical code for different platforms. For this purpose only the implementation of the API has to be adopted for each platform.

IX. Conclusion and Future Work

Using an environment with a dense grid of rewritable RFID tags for the navigation of AGVs is a promising, yet challenging, topic. Several routes can be marked quickly with a vehicle writing markers into the tags. This provides flexibility if new routes are required in a dynamic production process or in case of failure.

Experiments have shown, that a single RFID reader does not enable smooth movements. It is thus unsuitable for manufacturing scenarios. First experiments with a magnetic compass as additional sensor suggest that it enables important improvements, but also exposed its delicateness for disturbances of the terrestrial magnetic field frequently existing in manufacturing environments.

This paper has started research in a new field, but the main work still lies ahead. A quantitative analysis of the proposed algorithms is necessary to get a deeper insight into the current shortcomings. This will allow a better understanding of the potentials of the additional sensors discussed in section VI and may lead to new algorithms. In particular it can be expected that configurations with two or more RFID readers will outperform the algorithms discussed in this paper.

An issue not discussed yet is the management of the storage on the tags. Most RFID tags only allow to read the data in blocks of four bytes. Hence, the organisation of data has to be aligned according to these boundaries (i.e. path identifier and sequence number should be stored in the same block). Otherwise the number of read operations will be unnecessary high, leading to a low driving speed. To reduce the number of bits stored on a tag, a suitable coding must be used. In particular this applies to additional data stored when using a compass or odometry.

Algorithms can also be developed for other scenarios as proposed in section III for constituting a virtual train, marking forbidden regions etc. The RFID reader used in the prototype is able to read four tags in its environment at a time. It is currently not known, whether readers surpassing this value, would allow for improved algorithms.

Acknowledgement

We want to thank Mr. Tiedemann from Herma GmbH for providing us RFID tags for our testing environment.

References

[1] R.H. Hollier, Ed., *Automated Guided Vehicle Systems*, Springer, 1987.

[2] Morgen E. Peck, "RFID Tags Guide the Blind," *IEEE Spectrum*, vol. 1, 2008.

[3] "Texas Instruments RFID Tracks Containers Throughout Europe," *RFID News, Texas Instruments*, p. 10, 2000.

[4] "Der Dynamik des globalen Handels begegnen," *RFID-Newsletter 01/2006, Metro AG*, pp. 8–9, 2006.

[5] Jürgen Bohn and Friedemann Mattern, "Super-distributed rfid tag infrastructures," in *Proceedings of the 2nd European Symposium on Ambient Intelligence (EUSAI 2004)*, Eindhoven, The Netherlands, Nov. 2004, number 3295 in Lecture Notes in Computer Science (LNCS), pp. 1–12, Springer-Verlag, Springer-Verlag.