Coordination of AGVs in an Industrial Environment

(Demo Paper)

D. Herrero-Perez
Department of Information and Communication Engineering, University of Murcia
30100 Murcia, Spain
dherrero@dif.um.es

H. Martinez-Barbera
Department of Information and Communication Engineering, University of Murcia
30100 Murcia, Spain
humberto@um.es

ABSTRACT

This demonstration aims to show the decentralized coordination of the navigation system of highly autonomous Automated Guided Vehicles (AGVs) integrated in a real industrial environment. These vehicles are used for goods delivery tasks between different points of the production system. The *navigation* part consists on calculating and following the trajectory to reach the goal, which is calculated considering the plant layout and recalculated to avoid the non-modeled obstacles when they are sensed during the navigation. The *coordination* part is based on a decentralized architecture where each vehicle broadcasts the information about its state in the working environment, and by combining all these states in a local way, each AGV decides which action to take.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed AI – Coordination, Intelligent agents, Multiagent systems.

Keywords

Industrial Robots, Multi-Robot Systems, Path Planning, Automated Guided Vehicles

1. INTRODUCTION

When there are several robots working, either collaborating or not, in the same environment performing tasks which have dependences between them, it is mandatory to coordinate them. In our case, we show several automated guided vehicles performing transportation tasks in a highly dynamic environment. These vehicles are used for goods delivery tasks between different points of a production system, being able to navigate up to one meter per second. The coordination aims to avoid possible collisions between vehicles and deadlock cases, maximizing the performance of the global system, while keeping it fully decentralized.

In order to perform the transportation requests, the vehicles need a representation of the environment, which is used for path planning. There are two fundamental paradigms for path-planning: the grid based paradigm and the topological paradigm. The grid map approach presents several problems, above all the space and time complexity representing a difficulty for the planning task. On the other hand, topological maps are more compact permitting fast planning and joint use with high level planners and traffic coordination system. But this kind of maps presents other problems like the maintenance in large environments. Because we have to deal with a relative large environment, a reasonable solution is the use of both approaches, using a topological map for a high level description and a grid map for a local description of the environment including currently sensed obstacles. In particular, we are using a two-level topological map for i) storing the relevant places that the AGV could need to reach, and ii) representing how to move from different zones or rooms to others.

Cite as: Coordination of AGVs in an Industrial Environment (Demo Paper), D. Herrero-Perez and H. Martinez-Barbera, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008,Estoril,Portugal,pp.1709-1710.

2. NAVIGATION

The AGVs are not navigating with fixed path, but they calculate the path dynamically using a search algorithm D* over a fuzzy grid map, what makes the system more flexible and facilities the implantation and the starting of the whole system. The location of each AGV is broadcast for the coordination between vehicles. This information is also used for mapping the fuzzy grid map, and then considering the other vehicles as static obstacles. In order to include the other vehicles in the fuzzy grid map, we are modeling each AGV as a bounding box, Figure 1, considering this place to generate the path of the robot, i.e., avoiding that the path generated cross the area occupied by the other vehicles.

When the vehicle detects a possible collision with an obstacle or other vehicle, an avoidance obstacle behavior is activated. We are using a reactive method named *Polar Kinematics Bug*, which is an extension of the method *PolarBug* adapted to the configuration of the vehicles used, which are tricycle type. This method makes many assumptions to simplify computations and provide a fast solution because obstacle avoidance in industrial robot has logically strong real time restrictions.

Because of the angle of view of the laser rangefinder is limited to 180 degrees and this device is installed in the frontal part of the vehicle, some lateral obstacles are not visible and the vehicle could collide. In order to avoid this problem, a local buffer is used to store some close points that are not visible from new locations, and these point are likewise handled. There are elements of the AGV which are not visible using the laser, such as the fork, because the laser rangefinder is installed in the forward of the vehicle. This problem is solved using the location information which is

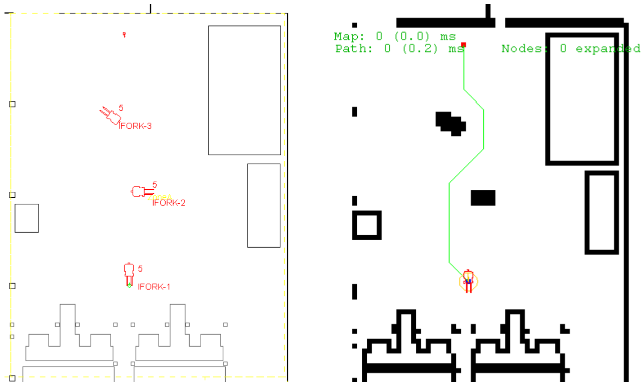

Figure 1: Path planning using D* method and considering the other AGVs as static obstacles

broadcast by each vehicle, the robot's location is modeled as a bounding box, rectangle, and this is handled as an obstacle by the obstacle avoidance method.

When there is a conflict in the navigation of several AGVs, this is solved using re-planning and obstacle avoidance, hence, obstacle avoidance is critic in this application.

3. DECENTRALIZED COORDINATION

In order to coordinate the robots, we are using a decentralized architecture in which AGVs navigate independently and broadcast the information about their state. The path planning method, which considers the other vehicles, and the obstacle avoidance are enough to avoid collisions between vehicles, but they are not for avoiding deadlocks: AGV traffic jams where two vehicles are waiting each other for finishing a task while they stand still. Thus a set of coordination rules are mandatory. Each AGV periodically broadcasts over a wireless link a coordination tuple, which contains the following information: location, priority, and critical zone that is occupying.

The priority of each vehicle is calculated considering its current task; tasks with higher priorities are the tasks that imply carrying a pallet and the change of zone. The state of the vehicle also influences in its priority. The possible states of the vehicle are: stopped, waiting, giving way and normal state.

Due to the nature of a decentralized architecture, it is difficult to avoid collisions between vehicles when they are all moving independently , i.e., they can suddenly change goal point or current path. Because of this, we only allow the navigation when the distance between vehicles is larger than a given safety distance. Therefore, each vehicle calculates the distance to the others using the broadcast locations, and then decides if it should give way according to the priorities of the others. In case of having the same priorities, i.e. their assigned tasks are of the same complexity and relevancy, the tie is broken by using their network addresses, i.e. the lower one gets the priority. When all the vehicles inside a hazardous zone are stopped, the one with the higher priority is able to reach its destination point by re-planning and obstacle avoidance. Despite the method seems to have all but one AGV non-navigating, it is important to note that this is only applied when they are very close each other, and during normal operation this only occurs a few times.

Figure 2: Some custom modified AGVs in an industrial environment

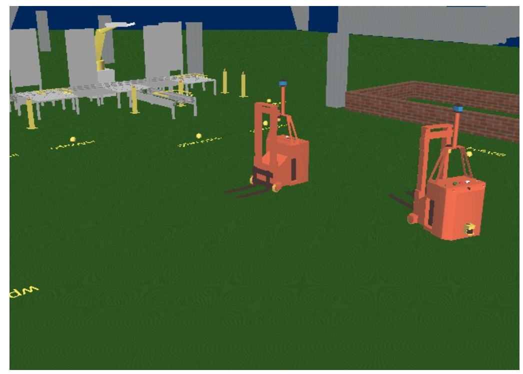

Figure 3: Simulated conflict between AGVs.

When a vehicle is performing a docking or undocking operation, it needs some free space to perform the maneuver. In case those two vehicles compete for the same maneuvering area, i.e. two close-by docking places, a deadlock is possible. To avoid it, these maneuvering areas are manually designated as critical zones, i.e. mutual exclusion zones. The first vehicle that enters one of these critical zones broadcasts a coordination tuple to indicate that it is the only vehicle permitted in that zone. The other competing AGVs stay still outside the critical zone until the one inside it leaves the zone, which is designated broadcasting a coordination tuple. The same method is used for doors and narrow passageways, which are designated as critical zones.

4. DEMOSTRATION

Unfortunately, it is not possible to transport the AGVs to the conference. Thus our demonstration consists on the presentation of video, in which several conflicts between AGVs are generated and handled. In addition, a simulation is presented, where we have a user interface to generate the conflicts between the AVGs and we are able to monitor the resolution of these conflicts.

5. ACKNOWLEDGMENTS

This work has been supported by CICYT project DPI2004-07993-C03-02. The authors would like to thank the company Frutas El Dulze S.L. and Ingenia tecnologia S.L. for granting access to the facilities required for this research.