Performance Assessment of PRIDE in Manufacturing Environments

Zeid Kootbally * — Tel: (301) 975-3428 — zeid.kootbally@nist.gov
Craig Schlenoff * — Tel: (301) 975-3456 — craig.schlenoff@nist.gov
Raj Madhavan *+ — Tel: (301) 975-2865 — raj.madhavan@nist.gov

* National Institute of Standards and Technology (NIST)
Gaithersburg, MD, USA

+ Computational Sciences and Engineering Division,
Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA

ABSTRACT

This paper describes PRIDE (Prediction in Dynamic Environments), a multi-resolution and hierarchical framework. PRIDE was developed as a test bed to assess the performance of autonomous vehicles in the presence of moving objects in a simulated environment. By simulating scenarios in which moving objects are prevalent, a designer of an autonomous vehicle can test the performance of their path planning and collision avoidance algorithms without having to immerse the vehicle in the physical world. This framework supports the prediction of the future location of moving objects at various levels of resolution, thus providing prediction information at the frequency and level of abstraction necessary for planners at different levels within the hierarchy.

Previous works have demonstrated the reliability of PRIDE to simulate on-road traffic situations with multiple vehicles. To provide realistic scenarios, PRIDE integrates a level of situation awareness of how other vehicles in the environment are expected to behave considering the situation in which the vehicles find themselves in.

In recent efforts, the PRIDE framework has been extended to consider production logistics in dynamic manufacturing environment while focusing on the scheduling of material transportation system. To demonstrate the characteristics of the PRIDE framework, this paper illustrates real-time navigation of Automated Guided Vehicles (AGVs) at different locations in a dynamic manufacturing environment. Moreover, using the high-fidelity physics-based framework for the Unified System for Automation and Robot Simulation (USARSim), this paper analyzes the performance of the PRIDE framework on a set of realistic scenarios.

1. INTRODUCTION

From traditional and well-established applications in the automotive industry to emerging applications such as material handling, palletizing, and logistics in warehouses, the use of mobile robots can increase productivity whilst ensuring personnel safety. Automated Guided Vehicles (AGVs) represent an integral component of today’s manufacturing processes. They are widely used on factory floors for intra-factory transport of goods between conveyors/assembly sections, parts/frame movements, and truck-trailer loading/unloading.

According to Bishop Consulting's report [1] on AGV Industry Next-Generation Technology Priorities, "In the eyes of the system vendors, the most prominent technology development area is in moving from today's AGVs, which require highly structured environments and reference markers installed throughout the plant, to operating in less structured or unstructured environments. In fact, the site preparation required to install these reference markers is a significant portion of the system cost …". To offset prohibitively expensive maintenance and installation costs, and thus expand the AGV's markets and utility beyond what is possible today, it is evident that the dependency on infrastructure is to be minimized (if not eliminated). To achieve this goal and to be able to cope with unstructured, dynamic environments, predicting future positions of moving objects in factory environments are critical enablers for widespread use of AGVs.

The research interest of this paper deals with path planning of AGVs using the PRIDE (PRediction In Dynamic Environments) framework. PRIDE is a multi-resolution, hierarchical framework that provides an autonomous vehicle planning system with information required to perform path planning in the presence of moving objects. PRIDE incorporates multiple prediction algorithms into a single, unifying framework. To date, we have applied this framework to simulate the prediction of the future location of autonomous vehicles during on-road driving.

The PRIDE algorithms are not limited to on-road driving and can be ported to other domains. As such, this paper illustrates how PRIDE has been extended to path planning of AGVs in manufacturing environments. In the factory environments used by PRIDE, the autonomous vehicles are industrial robots (e.g., unit loaders, forklifts) performing collision-free intra-factory activities, including transport of goods between conveyors and assembly sections, parts and frame movements, and truck-trailer loading/unloading.

The remainder of this paper is organized as follows: Section 2 gives an overview of the PRIDE framework. Section 3 describes the different features added and modified in PRIDE to port this framework to manufacturing. Section 4 details a manufacturing moving object ontology (M2O2), an ontology used to plan the paths of the other AGVs. Section 5 discusses the performance of the PRIDE framework through three scenarios involving two AGVs and Section 6 concludes this paper and gives an overview of the future work.

2. The PRIDE Framework

PRIDE is a multi-resolution hierarchical framework that provides an autonomous vehicle planning system with information required to perform path planning in the presence of moving objects. This framework supports the prediction of the future location of moving objects at various levels of resolution. PRIDE is based on the 4D/RCS architecture [2], which provides a reference model for unmanned vehicles on how their software components should be identified and organized.

The PRIDE framework provides moving object predictions to planners running at any level of the 4D/RCS hierarchy at an appropriate scale and resolution. The underlying concept of PRIDE lies in the incorporation of multiple prediction algorithms into a single, unifying framework.

At the higher levels of the framework, the prediction of moving objects needs to occur at a much lower frequency and a greater level of inaccuracy is tolerable. At these levels, moving objects are identified as far as the sensors can detect and a long-term (LT) prediction algorithm predicts where those objects will be at various time steps into the future. Higher-level reasoning processes need a global representation of the environment to compute the future location of an AV. PRIDE uses the road network database (RND) [3] to access different information about the road networks, including individual lanes, lane markings, intersections, legal intersection traversability, etc. The lower levels of the framework use estimation theoretic short-term (ST) predictions based on an Extended Kalman Filter (EKF) to predict the future location of moving objects with an associated confidence measure. Complete details on the LT and ST prediction algorithms can be found in previous efforts [4].

PRIDE currently integrates the Mobility Open Architecture Simulation and Tools (MOAST) framework along with the Unified System for Automation and Robot Simulation (USARSim) [5]. This integration provides predictions incorporating the physics, kinematics and dynamics of AVs involved in traffic scenarios. MOAST is a framework that provides a baseline infrastructure for the development, testing, and analysis of autonomous systems1. MOAST implements a hierarchical control technique, which decomposes the control problem into a hierarchy of controllers with each echelon (or level) of control adding additional capabilities to the system. USARSim is a high-fidelity physics-based simulation system that provides the embodiment and environment for the development and testing of autonomous systems. USARSim utilizes high-quality 3D rendering facilities to create a realistic simulation environment that provides the embodiment of a robotic system. The system architecture on the integration of PRIDE with the MOAST and USARSim frameworks is described in previous work [6].

PRIDE also handles drivers' aggressivity. In this context, the aggressivity represents the style and driving preferences of a driver. For example, one would likely assume that a conservative driver will remain in his lane whenever possible and will keep a gap between his vehicle and the leading vehicle. Conversely, an aggressive driver would have a higher probability of changing lanes and would be more apt to tailgate the leading vehicle. One may also find that the aggressivity of the driver may change over time, e.g., the driver can be very aggressive when trying to get to a certain lane, but become more passive when he gets there. The PRIDE framework addresses all the driver types and situations mentioned above. Experiments and corresponding results performed on aggressivity can be found in previous work [7].

3. From On-road to Manufacturing Environments

The PRIDE algorithms are not limited to on-road driving and can be ported to other domains. In this paper, the PRIDE framework deals with path planning in manufacturing environments. In a factory setting, the autonomous vehicles are AGVs (e.g., unit loaders, forklifts) performing collision-free intra-factory activities, including transport of goods between conveyors and assembly sections, parts and frame movements, and truck-trailer loading/unloading. To handle these systems, some features of the current framework have to be modified.

The RND structures must comply with factory settings and safety. New maps are built to accommodate industrial robots requirements. For instance, hazardous areas are specified and loading/unloading stations are setup for goods deliveries around the factory.

---

[1] Autonomous systems in this context refer to embodied intelligent systems that can operate fairly independently from human supervision.

To be able to handle new missions in manufacturing environments, the long-term prediction algorithm is modified as well. During on-road driving, the autonomous vehicles were not asked to reach a specific target. Instead, the PRIDE algorithms focused on the actions the autonomous vehicles would take based on the situations they find themselves in. In manufacturing settings, AGVs have to reach different targets to load and unload supplies for delivery. For better productivity, AGVs have to move along collision-free paths with the minimum time. To reduce the time during missions, PRIDE uses the algorithm of Dijkstra [8] to compute the shortest path from the start position to the target position for each AGV. Figure 1 depicts the process overflow of the PRIDE prediction algorithm.

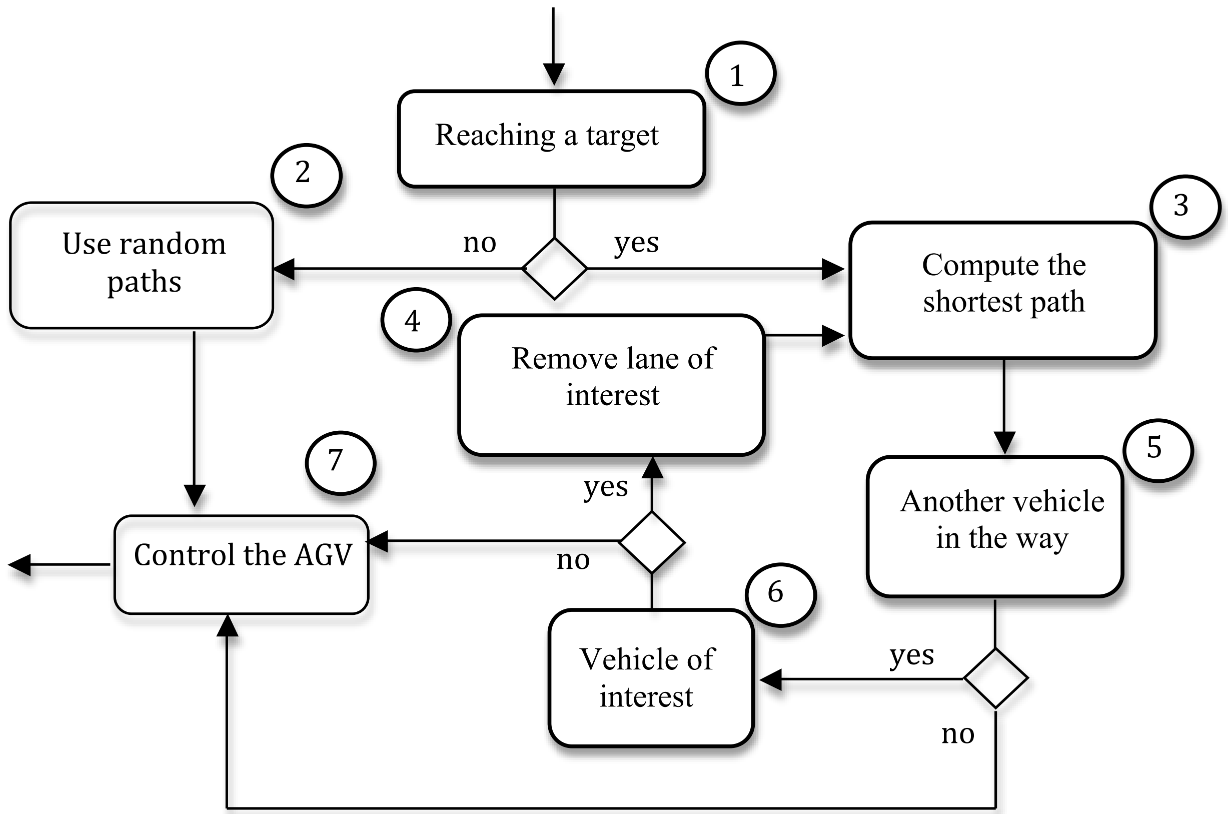

Figure 1. Process overflow of the long-term prediction algorithm.

As illustrated in Figure 1, the PRIDE algorithms check if the current vehicle has to reach a target (1), which is specified in a xml file. If a target is not specified, the vehicle chooses random paths (2) and the PRIDE algorithms send waypoints to control the AGV (7). If the AGV has to reach a target, the Dijkstra's algorithm is used to compute the shortest path from the current position of the AGV to the target (3). If another vehicle is within the computed shortest path (5), its category and its employment role in the factory is retrieved from a manufacturing moving object ontology. According to the category and the role of the vehicle, the decision to re-compute another path or not is made. Another path is re-computed when a vehicle of interest is met (6). For instance, if the AGV meets a forklift lifting some supplies, the PRIDE algorithms will re-compute another path minus the lane (lane of interest) in which the vehicle of interest is situated (4). The lane of interest is removed from the RND only for the current AGV at this time. During the next loop, the PRIDE algorithms check again if there is a vehicle of interest within the path of the AGV. The next section details the manufacturing moving object ontology used by PRIDE.

4. A MANUFACTURING MOVING OBJECT ONTOLOGY

The goal of this Manufacturing Moving Object Ontology (M2O2) is to provide a neutral knowledge representation (the data structures) capturing relevant information about moving objects on the shop floor and their capabilities. This information will be fed into the moving object prediction algorithms to give them relevant information about moving objects they will encounter on the shop floor to allow them to make better predictions as to where those objects will be at points in the future. This knowledge representation must be complete, unambiguous, and flexible enough to adapt as the moving object requirements evolve. As such, we have chosen to use an ontological approach to representing these requirements.

In this context of this paper, an ontology can be thought of as a knowledge representation approach that represents key concepts, their properties, their relationships, and their rules and constraints. Whereas taxonomies usually provide only a set of vocabulary and a single type of relationship between terms (usually a parent/child type of relationship), an ontology provides a much richer set of relationship and also allows for constraints and rules to govern those relationships. In general, ontologies make all pertinent knowledge about a domain explicit and are represented in a computer-interpretable fashion that allows software to reason over that knowledge to infer addition information.

By taking an ontological approach, we provide for:

- Less ambiguity in term usage and understanding
- Explicit representation of all knowledge, without hidden assumptions
- Conformance to commonly-used standards
- Available of the knowledge source to other arenas outside of the moving object prediction domain
- Availability of a wide variety of tools (reasoning engines, consistency checkers, etc.)

To date, very little work has been performed in developing the M2O2, apart from initial planning and preliminary surveys. It is expected that the M2O2 will be based on a number of existing technologies, including:

- OWL (Web Ontology Language) – OWL is a World Wide Web Consortium (W3C) recommendation (as of February 10, 2004). It defines terms commonly used in creating a model of an object or process, including classes/subclasses, properties/subproperties, property restrictions, and instances [9].

- OWL-S (Web Ontology Language – Services) – OWL-S is an OWL-based web service ontology, which describes the properties and capabilities of services in an unambiguous, computer-interpretable form. It was developed by the DARPA Agent Markup Language (DAML) Program. OWL-S is an upper ontology intended to be extended to meet specific applications [10].
- Protégé – Protégé is an open source ontology editor developed at Stanford University. It supports class and property definitions and relationships, property restrictions, instance generation, and queries. Protégé accommodates plug-ins, which are actively being developed for areas such as visualization and reasoning [11].

Though not yet developed, it is expected that the M2O2 will contain the following types of information:

- Relevant dimensions of the moving object
- Mobility characteristics of the object (e.g., how fast it can go, it's turn radius, etc.)
- Specific moving restrictions expected to be place on the moving object (e.g., staying on the right side of the travel lanes, staying in pedestrian walkways, etc.)
- Typical movement patterns (e.g., a forklift might first align itself with a pallet, drive towards the pallet until it is direct next to it, lift the load, drive backwards, etc.)

A more detail paper will be published on this topic as more progress is made.

5. PERFORMANCE OF PRIDE

In this section we analyze the performance of the PRIDE framework using two AGVs on a factory floor. The first AGV, a Unit Loader (Figure 2 left) is asked to reach a target while the second AGV, a Forklift (Figure 2 right) performs various actions in each scenario. For each scenario we only vary the role of the Forklift. Figure 3 represents the layout (map) of the factory used to run the scenarios presented in this paper. The start positions are the same for the Unit Loader and the Forklift in each scenario. The start and the target positions (denoted by *) are depicted in Figure 3.

Figure 2. Unit Loader (left) and Forklift (right) simulated in USARSim.

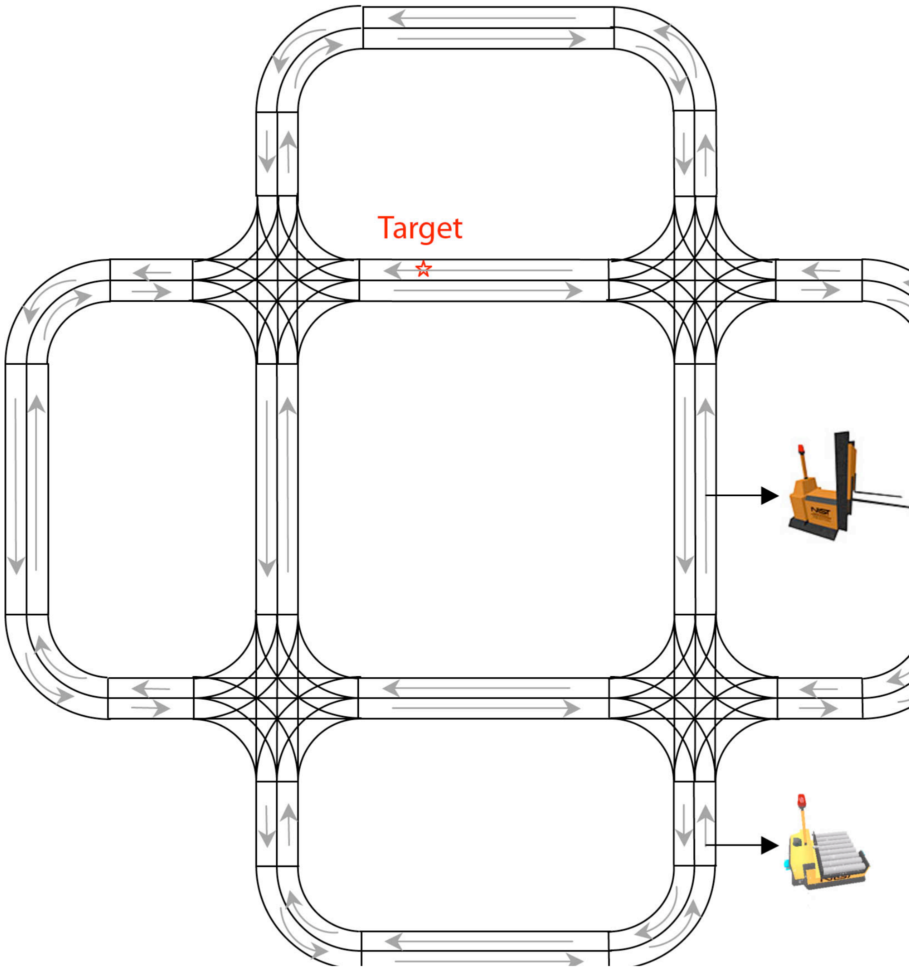

Figure 3. Start positions of the AGVs and target position. The arrows depicted in each lane represent the direction of traveling in the corresponding lane.

5.1. SCENARIO 1

In the first scenario, the Unit Loader has to reach the unload station (target on Figure 3). The Forklift is placed in the way the Unit Loader is supposed to follow. In this scenario, the role of the Forklift is to simply drive in the factory and no lifting is required.

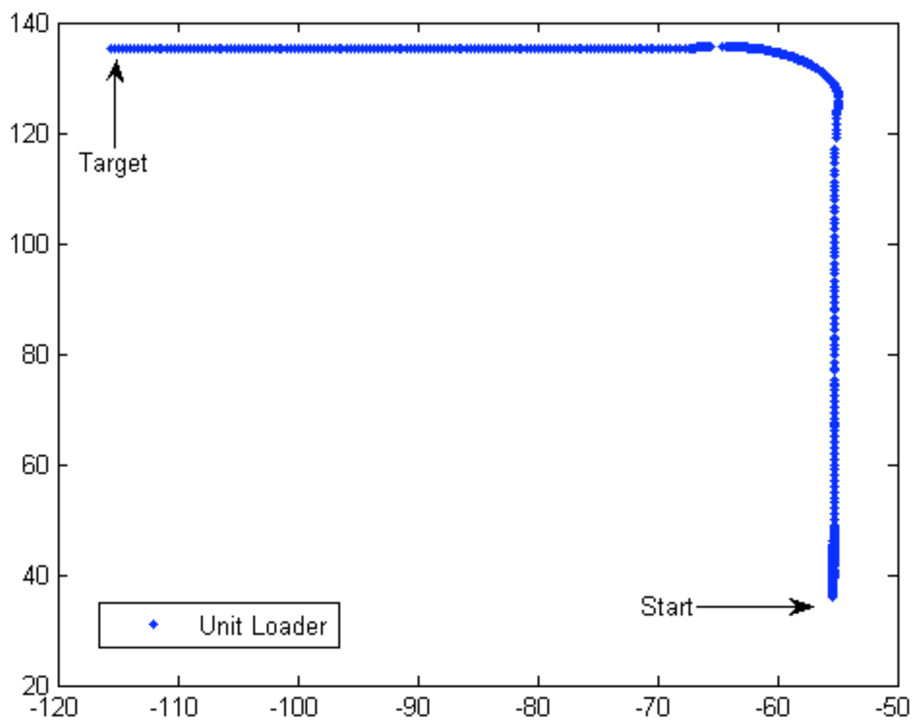

Figure 4. Current positions for the United Loader.

Figure 4 depicts the current positions of the Unit Loader while driving toward the target. The Unit Loader takes the shortest path even though the forklift is in the way. As specified in section 3, a vehicle of interest is defined by its category and its role. In this scenario, the Forklift is asked to drive in its current lane. The Unit Loader does not recognize the forklift as a vehicle of interest from the M2O2 and thus, the Unit Loader follows the shortest path. To better situate the path of the Unit Loader in the factory, a map and the path of the Unit Loader is represented in Figure 5.

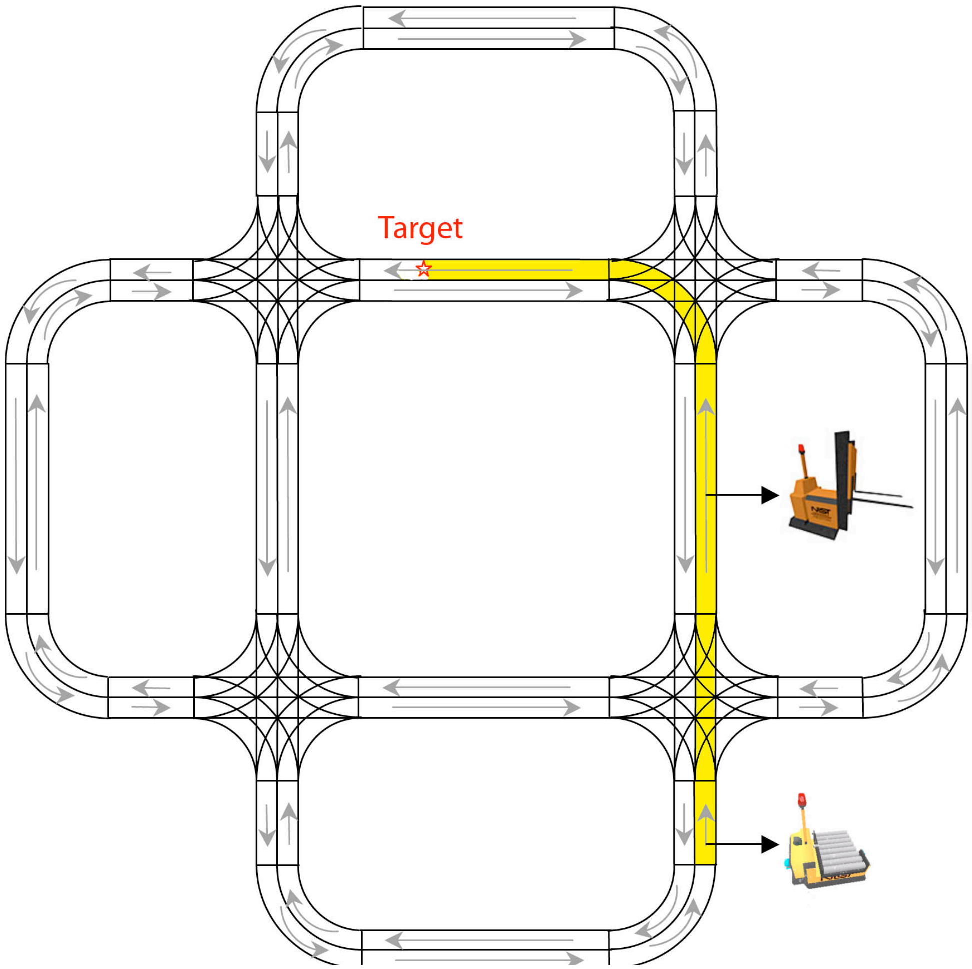

Figure 5. Shortest path followed by the Unit Loader.

5.2. Scenario 2

In the second scenario, the role of the Forklift has changed from the first scenario. Instead of only driving in the factory, the Forklift has new assignments, lifting and transporting supplies from a specific position to a target location. Figure 6 depicts the current positions of the Unit Loader from its start position to the target position.

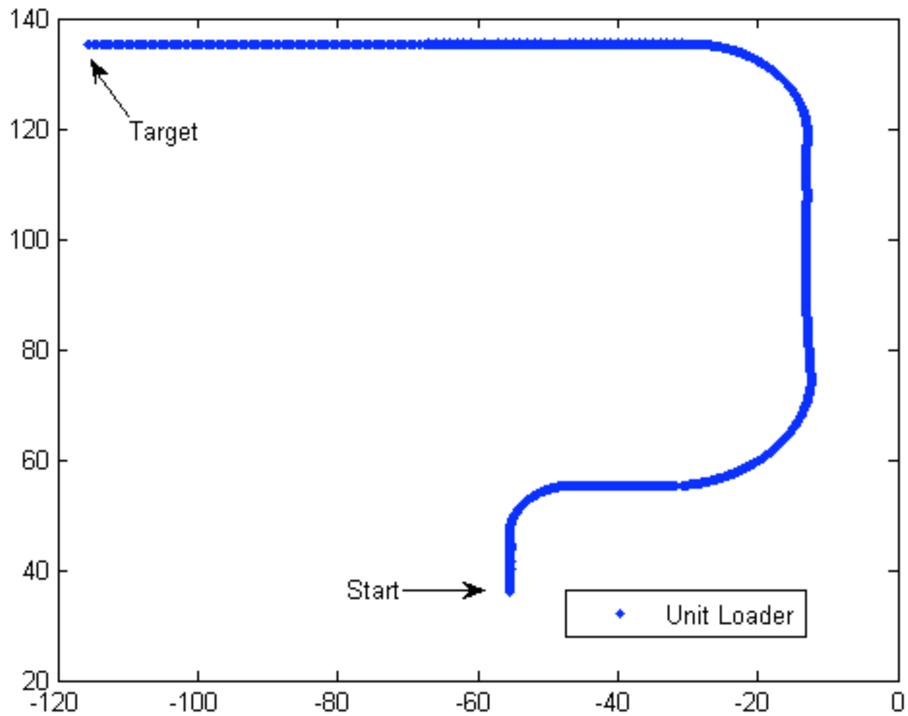

Figure 6. Current positions for the Unit Loader.

In this scenario, the path taken by the Unit Loader is different from the path generated in scenario 1. The Forklift is placed at the same position as in scenario 1, however, the role of the Forklift is different from the previous scenario. The Forklift has to lift and transport supplies, which classifies the Forklift as vehicle of interest according to the M2O2. Since a vehicle of interest is in the path planned for the Unit Loader, the PRIDE algorithms recompute the shortest path after removing lanes of interest (Steps 6, 4 and 3in Figure 1). To better situate the path of the Unit Loader in the factory, a map and the path of the Unit Loader is represented in Figure 7.

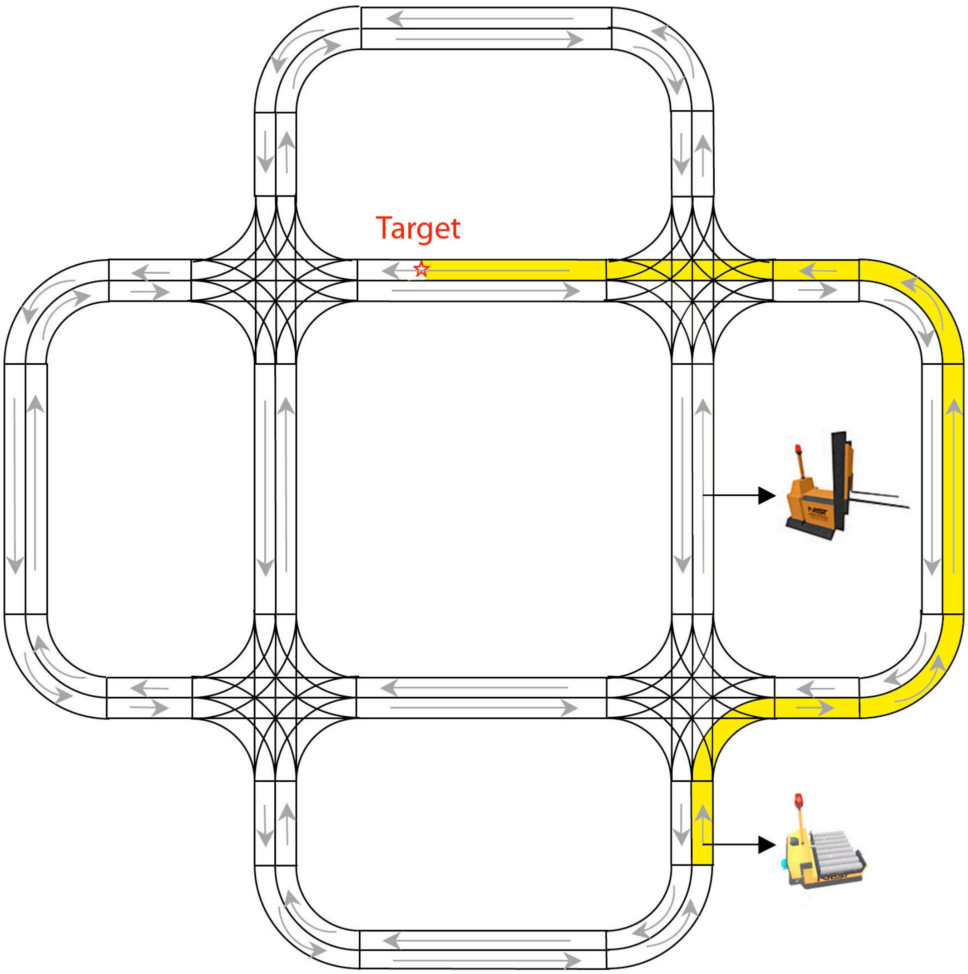

Figure 7. Shortest path followed by the Unit Loader after identification of a vehicle of interest.

6. Conclusion And Future Work

This paper described the PRIDE framework, which predicts the future location of moving objects in environments for the purpose of path planning for autonomous ground vehicles. To date, PRIDE has been involved for on-road driving. The work presented in this paper discusses the performance of PRIDE for manufacturing. Automated Guided Vehicles (AGVs) represent an integral component of today's manufacturing processes. They are widely used on factory floors for intra-factory efforts. Different features of PRIDE have been modified to comply with manufacturing environments, such as reaching a target using the shortest path. A new feature of PRIDE is its ability to use a manufacturing moving object ontology to identify vehicles of interest. Through different scenarios using the Mobility Open Architecture Simulation and Tools (MOAST) and the Unified System for Automation and Robot Simulation (USARSim), preliminary results have demonstrated reasonable performance of the PRIDE algorithms for path planning in factory settings.

Although substantial progress has been made in designing and porting the PRIDE framework to manufacturing, there is still much to be done. More complicated situations must be considered in

PRIDE. Different robots should be modeled in MOAST and USARSim, security areas should be setup in virtual environments and the interaction between humans and AGVs should be taken into account. In order to phase the PRIDE algorithms from a simulation environment to a functioning autonomous vehicle, there is a necessity on using previously gathered data to see how well the system performs in predicting the future location and behavior of other AGVs in manufacturing areas. Once the algorithms are validated through this experiment, they will begin to be transitioned to an autonomous vehicle for intra-factory driving.

BIBLIOGRAPHY

[1] "Bishop Consulting, AGV Industry Next-Generation Technology." Contract 2005-08-08-01. KT Consulting, 2006.

[2] Albus, J. (2002). 4D/RCS: A Reference Model Architecture for Intelligent Unmanned Ground Vehicles. *Proceedings of SPIE Aerosense Conference*, (pp. 303-310). Orlando.

[3] Kootbally, Z., Schlenoff, C., & Madhavan, R. (2007). A Brief History of PRIDE. *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, (pp. 40-47). Gaithersburg.

[4] Schlenoff, C., Balakirsky, S., Barbera, A., Scrapper, C., Ajot, J., Hui, E., et al. (2004). *The NIST Road Network Database: Version 1.0.* National Institute of Standards and Technology, Gaithersburg.

[5] Schlenoff, C., Kootbally, Z., & Madhavan, R. (2007). Driver Aggressivity Analysis within the Prediction In Dynamic Environments (PRIDE) Framework. *Proceedings of the 2007 SPIE Defense and Security Symposium.*

[6] Schlenoff, C., Madhavan, R., & Kootbally, Z. (2006). PRIDE: A Hierarchical, Integrated Prediction Framework for Autonomous On--Road Driving. *Proceedings of the 2006 International Conference on Robotic Applications (ICRA)*, (pp. 2348-2353).

[7] Scrapper, C., Balakirsky, S., & Messina, E. (2006). MOAST and USARSim: A Combined Framework for the Development and Testing of Autonomous Systems. *Proceedings of the 2006 SPIE Defense and Security Symposium.*

[8] Dijkstra's Algorithm: http://en.wikipedia.org/wiki/Dijkstra%27s_algorithm

[9] Harmelen, F. and D. McGuiness. OWL Web Ontology Language Overview. <http://www.w3.org/TR/2004/REC-owl-features-20040210/>.

[10] Stanford Medical Informatics. 2005. <http://protege.stanford.edu/>.

[11] The OWL Services Coalition . "OWL-S 1.0 Release." 2003. <http://www.daml.org/services/owl-s/1.0/owl-s.pdf>.

BIOGRAPHY

Zeid Kootbally received his Ph.D. in computer science in 2008 from the University of Burgundy, France, for a dissertation on “Moving Object Predictions in Dynamic Environments for Autonomous Ground Vehicles”. Since 2005, he is a guest researcher in the Intelligent Systems Division at the National Institute of Standards and Technology. His research interests include path planning, autonomous navigation, primarily applied to on-road driving.

Craig Schlenoff received his Bachelors degree in mechanical engineering from the University of Maryland, College Park and his Masters degree in mechanical engineering from Rensselaer Polytechnic Institute. He is a researcher in the Intelligent Systems Division at the National Institute of Standards and Technology. His research interests include knowledge representation, ontologies, and process specification, primarily applied to autonomous systems and manufacturing. He has recently served as the program manager for the Process Engineering Program at NIST as well as the Director of Ontologies and Domain Knowledge at VerticalNet, Inc. He has served on the organizing and program committee of numerous knowledge representation and ontology related conferences and workshops, and has published over 70 papers in related areas.

Raj Madhavan is an R & D Staff with the Computational Sciences and Engineering Division of the Oak Ridge National Laboratory and a guest researcher with the Intelligent Systems Division of the National Institute of Standards and Technology. He received his Ph.D. in Field Robotics from the University of Sydney, Australia. Over the last decade, he has contributed to various topics in field robotics, systems and control theory. His current research interests include autonomous vehicle navigation in complex unstructured environments, performance metrics for intelligent systems, and urban search and rescue robotics. He has published over 125 papers in refereed journals, book chapters, conference proceedings, and technical reports.