Performance Evaluation of Cost-Based vs. Fuzzy-Logic-Based Prediction Approaches in PRIDE

Z. Kootbally, C. Schlenoff, R. Madhavan and S. Foufou

Intelligent Systems Division, National Institute of Standards and Technology (NIST)
100 Bureau Dr. Stop 8230, Gaithersburg, MD 20899-8230, USA

ABSTRACT

PRIDE (PRediction In Dynamic Environments) is a hierarchical multi-resolutional framework for moving object prediction. PRIDE incorporates multiple prediction algorithms into a single, unifying framework. To date, we have applied this framework to predict the future location of autonomous vehicles during on-road driving. In this paper, we describe two different approaches to compute long-term predictions (on the order of seconds into the future) within PRIDE. The first is a cost-based approach that uses a discretized set of vehicle motions and costs associated with states and actions to compute probabilities of vehicle motion. The cost-based approach is the first prediction approach we have been using within PRIDE. The second is a fuzzy-logic-based approach that deals with the pervasive presence of uncertainty in the environment to negotiate complex traffic situations.

Using the high-fidelity physics-based framework for the Unified System for Automation and Robot Simulation (USARSim), we compare the performance of the two approaches in different driving situations at traffic intersections. Consequently, we show how the two approaches complement each other and how their combination performs better than the cost-based approach only.

Keywords: Autonomous vehicles, cost-based approach, fuzzy-logic-based approach, fuzzy control, moving object prediction, PRIDE

1. INTRODUCTION

Fully autonomous vehicles have been the subject of international intensive research for many years. To better understand the challenges and problems of autonomous mobility from a number of different perspectives, researchers have launched many worldwide projects. Self-driving vehicles are not on the market yet, and many difficulties still need to be solved before the implementation of systems that fully control the behavior of a truly autonomous vehicle. Even so, many of today's cars feature technologies that are building blocks to a world populated by autonomous vehicles. Researchers have performed many experiments with autonomous vehicles to enhance the development of road safety, vehicle control during on-road driving situations, and driving-aids for handicapped people. Moreover, manufacturers have saved many lives by integrating automatic cruise controls, transmissions, parking systems, and collision detection in commercial cars.

In the past decade, the first prototypes of vehicles outfitted with automatic driving aids have been presented to the public. In 1995, a German team reached a milestone when their robot van (VaMP)[1] performed a trip over 1600 km from Munich (Germany) to Odense (Denmark) and back at a maximum speed of 180 km/h in traffic.

Commercial equipment and materials are identified in this paper in order to adequately specify certain procedures. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

Further author information: (Send correspondence to Z.K.)
Z.K.: E-mail: zeid.kootbally@nist.gov, Telephone: +1 301-975-3428
C.S.: E-mail: craig.schlenoff@nist.gov, Telephone: +1 301-975-3456
R.M.: E-mail: raj.madhavan@ieee.org, Telephone: +1 301-975-2865
R & D Staff Member, Computational Sciences and Engineering Division,
Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA.
S.F.: E-mail: sfoufou@u-bourgogne.fr, Telephone: +33 380-39-38-05
Laboratoire LE2i, University of Burgundy, B.P. 47870, Dijon, France.

In July 1995, the No Hands Across America project by the Carnegie Mellon University's Robotics Institute drove the Navlab 5 autonomously over 2797 miles out of the 2849 miles (98.2 %) from Pittsburgh, Pennsylvania, to San Diego, California. In November 2007, the third competition of the Defense Advanced Research Projects Agency (DARPA) Grand Challenge, known as the "Urban Challenge",[2] took place in Victorville, California. The competition features autonomous ground vehicles maneuvering in a mock city environment, interacting with each other and making intelligent decisions in real time based on the actions of other vehicles.

In an attempt to explain drivers' behaviors and to identify the reason for certain dysfunctions, psychologists have performed many experiments on drivers' actions and predicted behavior.[3] Our research interest bears upon a level of situation awareness of how other vehicles in the environment are expected to behave considering their situation. When humans drive, they need to understand how each object in the environment moves according to the situation it finds itself in. To address this need, we have developed a multi-resolution hierarchical framework, called PRIDE (PRediction in Dynamic Environments). This framework provides an autonomous vehicle planning system with information that it needs to perform path planning in the presence of moving objects.[4] 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.

Despite tremendous advances made in the field of autonomous vehicles in the last decade, a number of problems remain. Most of the difficulties originate in the interactions between vehicles and the real world, unstructured environment, and in the large uncertainties that are inherent in these environments.[5] While performing a driving task, autonomous vehicles face different levels of uncertainty in their vicinity due to noises and dynamic changes of the environment. When drivers approach an uncontrolled traffic junction for example, they negotiate their path based on the information they collect from the environment and from the other agents in the environment. That is, drivers rarely use crisp data, but they rather approximate this information to make the right decision. Such information could be the speed, the distance and the aggressivity of other drivers.

This paper proposes a new approach that couples the PRIDE cost-based algorithm with a fuzzy logic controller. The implemented system copes with vagueness and computes uncertainty tolerating predictions. We detail the integration of the fuzzy controller in PRIDE and describe the specific role of each approach and how they interact with each other. Through different tasks involving up to four vehicles using the Unified System for Automation and Robot Simulation (USARSim), we discuss the performance of the new framework in different traffic scenarios at traffic intersections.

This paper is organized as follows: Section 2 gives an overview of the PRIDE framework. Section 3 briefly describes the basics of fuzzy sets and details the integration of a fuzzy controller within PRIDE. Section 4 discusses the results of the simulation of different scenarios at a traffic intersection. Section 5 concludes the paper and gives an overview of future work.

2. OVERVIEW OF 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. The PRIDE framework is based on the 4D/RCS architecture,[6] which provides a reference model for unmanned vehicles on how their software components should be identified and organized.

We developed the PRIDE framework to provide 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 we use the long-term prediction approach to predict 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 autonomous vehicle. In PRIDE, we use a road network that we have fully developed, called the Road Network Database.[7] At the lower levels, we use Extended Kalman Filter (EKF) estimation theoretic short-term predictions to predict the future location of moving objects with an associated confidence measure. Complete details on the aforementioned prediction algorithms can be found in previous work.[8]

In recent efforts, the PRIDE framework has used the Mobility Open Architecture Simulation and Tools (MOAST) framework along with the USARSim system.[9] The integration of PRIDE with the MOAST and USARSim frameworks provides predictions incorporating the physics, kinematics and dynamics of vehicles involved in traffic scenarios. MOAST is a framework that provides a baseline infrastructure for the development, testing, and analysis of autonomous systems. 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 the Karma Physics engine and high-quality 3D rendering facilities of the Unreal game engine to create a realistic simulation environment that provides the embodiment of a robotic system. The integration and the system architecture on the integration of PRIDE with the MOAST and USARSim frameworks is described in previous work.[10]

Handling drivers' aggressivity is the latest enhancement of the PRIDE framework. In this context, the aggressivity represents the style and driving preferences of a driver. For example, we would likely assume that a conservative driver will remain in his lane whenever possible and stay at a safe distance behind the vehicle in front of him. Conversely, an aggressive driver would have a higher probability of changing lanes and would be more apt to tailgate the driver in front of him. We 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 these driver types and all situations mentioned above. Experiments and corresponding results performed on aggressivity can be found in previous work.[11]

3. INTEGRATION OF THE FUZZY CONTROLLER

Research in autonomous vehicles has widely adopted fuzzy controllers to design reactive robot behaviors.[12,13] We have used a fuzzy controller in PRIDE due to the uncertainty in both sensory information and motor execution. We also employ fuzzy logic to facilitate the driving task for an autonomous vehicle. Fuzzy systems have been used to design controllers that handle the behavior of autonomous vehicles and robotic systems for navigation and obstacle avoidance.[14,15] Such systems offer the advantage of an easy incorporation of the knowledge for a human expert in the prototype of the controller, and in terms of the robustness of the results. Driankov and Saffiotti[16] present a large variety of fuzzy techniques which address the challenges posed by autonomous robot navigation.

A fuzzy-logic-based system has the advantage that it allows the intuitive nature of intersection negotiation to be easily modeled using linguistic terminology. In addition, with the proposed methodology we can relax some of the constraints associated with the cost-based approach. The complex phase of designing and defining costs can turn tedious for complex traffic situations. Moreover, fuzzy logic is flexible, conceptually easy to understand, and has the particularity to be blended with conventional control techniques.

3.1. Fuzzy Set Theory

We shall start with some basic principles of the fuzzy set theory to better understand the mechanics of a fuzzy controller. Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth. Fuzzy sets and fuzzy logic are used to heuristically quantify the meaning of linguistic variables, linguistic values, and linguistic rules that are specified by the expert. In 1965, Zadeh introduced fuzzy sets as an extension of the classical notion of a set to represent concepts that do not have sharp boundaries. A fuzzy set is a class of objects with a continuum of grades of membership. Such a set is characterized by a membership function (MF) which assigns to each object a grade of membership in the interval $[0, 1]$. Mathematically, a fuzzy set A over a universal set X is defined by its membership function

$$\mu_A : X \rightarrow [0, 1]$$

such that, for any $x \in X$, the value of $\mu_A(x)$ represents the degree of membership by which x belongs to the set A.

There is a variety of shapes for membership functions. According to fuzzy set theory, the choice of the shape and width is subjective, but may follow a few rules: (1) A MF should be sufficiently wide to allow for noise in the measurement, (2) A certain amount of overlap is desirable; otherwise the controller may run into poorly defined states, where it does not return a well defined output. Commonly used function shapes for MF are shown in Figure 1.

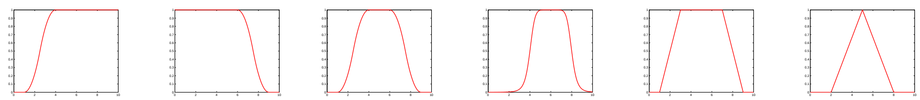

(a) (b) (c) (d) (e) (f)

Figure 1. Examples of membership functions: (a) S-shape, (b) Z-shape, (c) Pi-shape, (d) Generalized bell-shape, (e) Trapezoidal-shape, (f) Triangular-shape.

Once we have defined fuzzy sets, we need to define some basic relations and operations on them. The original set-theoretic operators of complement, intersection, and union given by Zadeh[17] are:

$$\mu_{A'}(x) = 1 - \mu_A(x)$$
$$\mu_{A \cap B}(x) = min(\mu_A(x), \mu_B(x))$$
$$\mu_{A \cup B}(x) = max(\mu_A(x), \mu_B(x))$$

for any x in the universal set X. More generally, many alternative fuzzy operators have been defined and studied. One can find more details on fuzzy operators and their use in Zadeh's work.[17]

3.2. Mechanism of a Fuzzy Controller

It is a difficult task to model and simulate complex real-world systems for control systems development, specially when implementation issues are considered. Fuzzy control provides a formal methodology for representing, manipulating, and implementing a human's heuristic knowledge about how to control a system that reasons in the presence of uncertain or vague data.

Fuzzy controllers are rule based controllers where the inference mechanism is grounded on fuzzy logic. The general architecture of a fuzzy controller is depicted in Figure 2. The fuzzy controller uses input data gathered from sensors and outputs information to control the vehicle. We can distinguish four main components in a fuzzy controller:

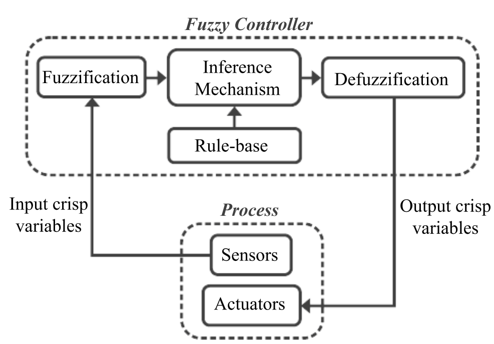


Figure 2. Architecture of a fuzzy controller.

1. The *fuzzification module* converts real input values to degrees of membership using the membership functions. The fuzzification block thus matches the input data with the conditions of the rules to determine how well the condition of each rule matches that particular input instance. There is a degree of membership for each linguistic term that applies to that input variable.

2. The *rule base* holds the knowledge, in the form of a set of rules, of how best to control the system. In general, fuzzy controllers are based on control rules of the type "IF *condition* THEN *control*". *condition* is always a fuzzy proposition (formula of fuzzy logic) of the type "x is A", where x is a linguistic variable and A is a linguistic term. *condition* tells when the rules should be applied; and *control* is a formula that describes the action to apply. In most practical cases, rules have a purely conjunctive form, that is, *condition* is a conjunction of atomic predicates of the input variables, and *control* is a single atomic predicate of the output variable. For example the following rules can be part of a control system that outputs the deceleration of a vehicle approaching a stop sign:

   IF "Distance" is Small AND "Speed" is Small THEN "Deceleration" is Small
   IF "Distance" is Small AND "Speed" is Medium THEN "Deceleration" is Big

3. The *inference mechanism* is the kernel of the fuzzy controller. It evaluates which control rules are relevant at the current time and then decides the fuzzy commands to apply to the process. The inference mechanism is performed in three steps: (1) If the *condition* of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. The input to the fuzzy operator is two or more membership values from fuzzified input variables. (2) We use the implication process that returns a fuzzy set. The input for the implication process is a single number given by the antecedent using (1), and the output is a fuzzy set. The fuzzy set is obtained by fuzzifying the membership function of the output, using the value obtained from the fuzzy operator from (1). (3) Aggregation is the process by which fuzzy sets that represent the outputs of each rule from (2) are combined into a single fuzzy set. Aggregation only occurs once for each output variable, just prior to the defuzzification process.

4. The *defuzzification module* converts the conclusions (fuzzy sets) reached by the inference mechanism into actual inputs for the actuators of the process. The basic idea of the defuzzification method is to compute a point in the resulting fuzzy set and to take its value as the output. Several defuzzification methods can be found in the literature.[18] Existing defuzzification methods differ mostly in the way they pick the point in the final fuzzy set. PRIDE applies the most popular and widely used method, the "Center of Gravity" (CoG), which is defined below:

   Let b_i denote the center of the membership function of the consequent rule (i). Let $\int \mu_i$ denote the area under the membership function μ_i. The CoG method computes the crisp output u to be

$$u = \frac{\sum_i b_i \int \mu_i}{\sum_i \int \mu_i}$$

3.3. Design of a Fuzzy Controller in PRIDE

We have developed a system that incorporates both the fuzzy-logic-based and the cost-based approaches. The new system implements an intelligent control strategy where it senses the information in the environment and reacts according to the nature of this information. In PRIDE, the combined-approach system is used for traffic intersection management. The system first computes the acceleration and deceleration of vehicles driving toward intersections, and then calculates the predicted estimates to control these vehicles. Although this paper focuses on the application of fuzzy controller at traffic intersections, we have used the fuzzy-logic-based approach in different situations, and for different purposes that we will present in future work.

3.3.1. Application of the Fuzzy Controller

PRIDE uses the fuzzy controller for vehicles at uncontrolled and controlled intersections. An uncontrolled intersection is a road intersection where no traffic lights or signs are used to indicate the right-of-way. In Figure 3, vehicles A and B are at uncontrolled intersections, whereas vehicles C and D are at controlled intersections defined by stop signs. The Road Network Database holds information for navigation at intersections. Such information is the type of intersection a vehicle is driving toward (controlled or uncontrolled), the type of traffic sign at the intersection, the number of paths at the intersection, ...

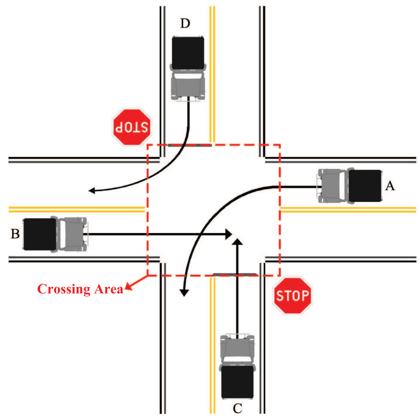


Figure 3. A scenario at an intersection.

The fuzzy controller takes into account two pieces of information to compute the acceleration and deceleration of a vehicle. The first one pertains to the rules developed for driver education courses.[19] These rules affect the strategies of intersection management such as the waiting time at stop signs, the rights-of-way, etc. The second point considers the aggressivity of the drivers. The aggressivity plays an important role in the system, where it greatly impacts the decision-making process, and thus the behaviors of the drivers.

Figure 3 sketches a scenario where the decision made by each vehicle depends on the objects of interest in the environment. In this context, an object of interest is a moving or stationary object in the environment that has a reasonable probability of intersecting the path of the controlled vehicle within a predetermined time frame. The reasonable probability is a function of the aggressivity of the controlled vehicle and the risk tolerance it is willing to assume. Note that it is impossible to know the exact path that a stationary or moving object in the environment will take at points in the future, so all possible paths must be considered. In the following, the crossing area represents the part of the intersection where the paths of the vehicles intersect and where collisions are possible.

In Figure 3, vehicle A is planning to turn left and vehicle B is planning to go straight. According to the driver education manual, vehicle A does not have the right-of-way and has to yield to vehicle B. However, if the driver of vehicle A is aggressive, he may be able to turn before vehicle B reaches the crossing area. Vehicle C has to consider vehicles A and B and thus makes a decision based on the paths that these vehicles plan to take. In the same way, the path of vehicle A can interfere with the path of vehicle D if vehicle A decides to go straight.

In these different cases, we clearly see that the decisions the drivers make depend on different parameters that impact the action of the vehicles. The drivers have the choice to decelerate or accelerate to perform one action over the other. The fuzzy controller used in PRIDE is in charge of combining different parameters and outputs the acceleration and deceleration of the vehicles to perform the appropriate action. The elaboration of the parameters constitutes the first part of the design process of a fuzzy controller and corresponds to the design of the fuzzy sets. The second part consists of developing a set of rules able to model the intuitive way we negotiate intersections using linguistic terminology, this is the build-up of the rule base. The two parts of the design process are described below.

Membership Functions We first need to develop the membership functions for the fuzzification and defuzzification modules. Figures 4(a) and 4(b) respectively represent the input and output fuzzy variables along with their corresponding fuzzy sets. As we can see, the fuzzy controller uses three different input parameters (distance, speed, aggressivity) in order to compute the acceleration/deceleration to control the vehicle. The membership functions depicted in Figure 5 are used to translate real conditions into fuzzy logic values.

The values used for the boundaries of the fuzzy variables are relevant to the information we have on the vehicles and the road network in PRIDE. "Distance" represents the distance between a vehicle and the beginning of the crossing area. The maximum distance of a road to an intersection is 10 m. "Speed" represents the speed of a vehicle and does not exceed 20 m/s. We have defined different levels for the "Aggressivity" of a vehicle by values ranging from 0 to 10. "Acceleration/Deceleration" represents the acceleration and deceleration of a vehicle and is used both as an input variable and an output variable. The values for the acceleration and the deceleration are in the interval $[-4.5, 3.5]$ m/s^2.

Input fuzzy variables	Distance	Speed	Aggressivity	Acceleration / Deceleration
Fuzzy sets	Zero (Z) Small (S) Medium (M) Big (B)	Zero (Z) Small (S) Medium (M) Big (B)	Passive Normal Aggressive	Negative Big(NB) Negative Medium (NM) Negative Small (NS) Zero (Z) Positive Small (PS) Positive Medium (PM) Positive Big (PB)

(a)

Output fuzzy variable	Acceleration / Deceleration
Fuzzy sets	Negative Big(NB) Negative Medium (NM) Negative Small (NS) Zero (Z) Positive Small (PS) Positive Medium (PM) Positive Big (PB)

(b)

Figure 4. Input and output fuzzy variables.

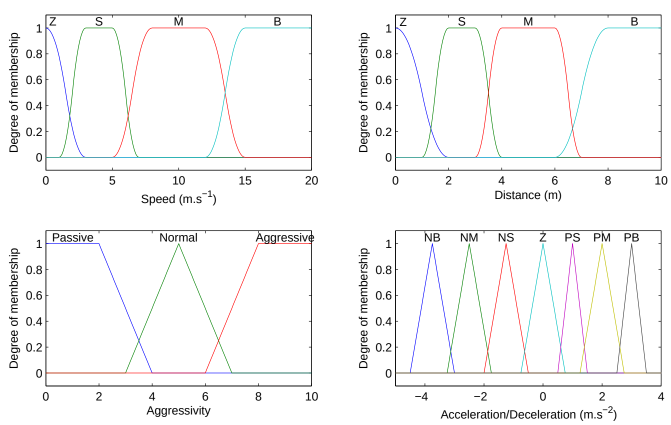


Figure 5. Membership functions for input and output variables.

Rule Base Once the membership functions are defined for the fuzzy sets, we need to build a rule base for the system. The rule base consists of a set of IF-THEN rules that map fuzzy range variables into fuzzy speed, distance, aggressivity and acceleration/deceleration commands. The rule base defined in PRIDE contains more than two hundred fifty rules.

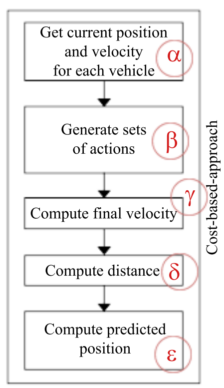
The rules need to reflect the behaviors we take when approaching an intersection. One rule from the rule base in PRIDE is given below. "Speed A" and "Speed B" represent the fuzzy variable "Speed" for the main vehicle and for the object of interest, respectively. Likewise, "Dist A" and "Dist B" represent the fuzzy variable "Distance" for the main vehicle and for the object of interest, respectively. "Aggressivity A" is the aggressivity of the main vehicle.

IF “Speed A” is M AND “Dist A” is M and “Speed B” is S AND “Dist B” is M AND “Aggressivity A” is Aggressive AND “Deceleration B” is NM THEN “Acceleration A” is PM

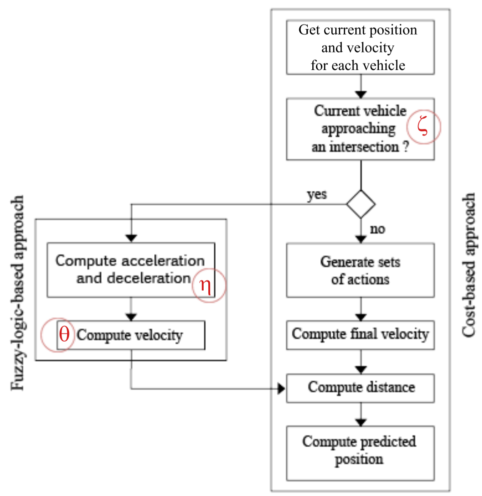
This rule can be applied to vehicle A in the scenario described in Figure 3 with vehicle B as object of interest. In this rule, vehicle A accelerates to turn before vehicle B reaches the crossing area.

3.3.2. Combinations of the Two Approaches

The cost-based approach uses the long-term prediction algorithm to compute the future location of moving objects. The overall process flow depicted in Figure 6(a) shows the mechanics of the cost-based-approach. Figure 6(b) shows the combination of the fuzzy-logic-based approach with the cost-based approach. To better illustrate the interaction between the two approaches, we show here a simplification of the long-term prediction algorithm. For a more complete description of the algorithm, one may want to refer to previous work performed on the PRIDE framework.[8]



(a)



(b)

Figure 6. Process flows for the cost-based and fuzzy-logic-based approaches.

The cost-based approach first retrieves the current position and the velocity of the corresponding vehicle (α) and generates different action sequences associated with a cost (β). The process of predicting several time steps into the future consists of a series of continuous actions (action sequences) which constitute a driving procedure. Each action is accomplished in one time step, thus, for a time of prediction n, n actions will be completed. The long-term prediction algorithms use two types of actions: A set of speed profiles and changing of lanes. From the action sequences, the algorithm then computes the final velocity v_f (γ). v_f is calculated by summing the velocity of each action within a set of actions. The next step is to get the distance d (δ) the vehicle will drive considering the final velocity v_f using $d = v_f \times \textit{time of prediction}$. The computation of the predicted position (ϵ) is the last step of the cost-based approach. Using the road network database and the distance d, the algorithm computes the predicted position the vehicle will be at n seconds in the future. The cost of each set of actions is inversely proportionally converted to probability. The set of actions with the highest probability is then selected and performed by the vehicle.

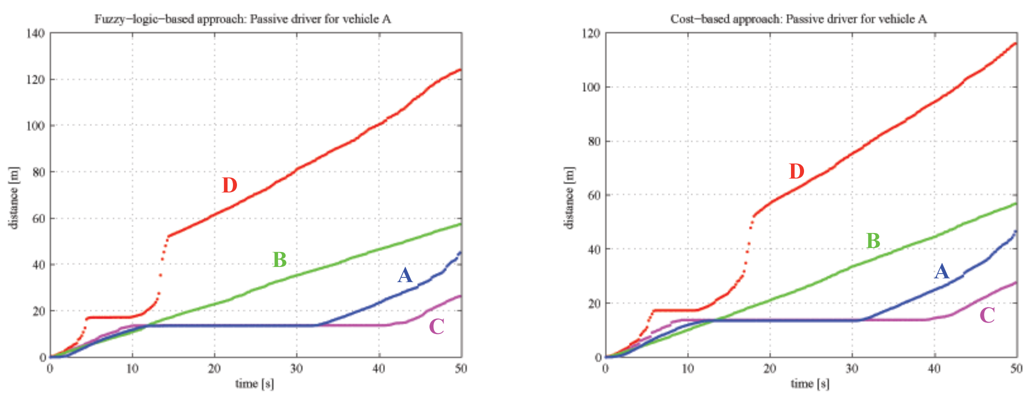
The fuzzy-logic-based approach is triggered in specific situations. The long-term prediction algorithm checks the structure of the road where the autonomous vehicle is driving (ζ) and then hands it over to the fuzzy controller when needed. In this paper, the fuzzy-logic-based approach is applied for vehicles approaching intersections. We previously saw that the cost-based approach computes different sets of actions and then drives the vehicle by performing the set with the lowest cost (highest probability). The fuzzy-logic-based approach acts differently by computing the acceleration/deceleration of the vehicle to negotiate the traffic intersection (η). Since the prediction algorithm requires the velocity to control the vehicle, the fuzzy-logic-based approach computes this new velocity from the acceleration/deceleration (θ). The remaining processes used to compute the predictions are the same as the cost-based approach as described previously.

4. SIMULATION AND EXPERIMENTAL RESULTS

We have simulated two scenarios with four vehicles (A, B, C, and D) at a four-way traffic intersection. The vehicles follow the paths sketched in Figure 3. For each scenario we have performed one simulation using the cost-based approach and one simulation using the fuzzy-logic-based approach. The driver of vehicle A is fully passive in scenario 1 where the aggressivity is set to 2 and fully aggressive in scenario 2 where the aggressivity is set to 9. The minimum time at a stop sign is set to 5 s.

4.1. Behaviors of Four Vehicles at a Traffic Intersection

In the first simulation of scenario 1, we use the fuzzy-logic-based approach to control each vehicle. Figure 7(a) depicts the distance covered by the vehicles and the corresponding period of time (distance-time diagram) for this simulation. The gradient of the curves represents the actual velocity.

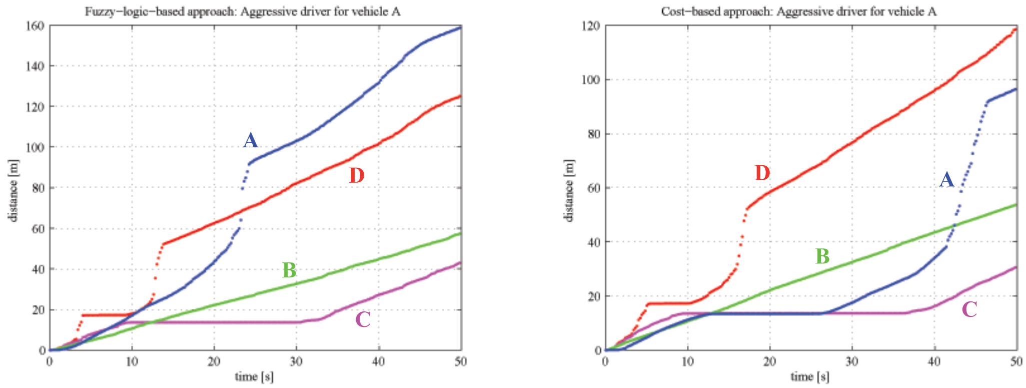


(a) (b)

Figure 7. Distance-time diagrams for scenario 1 using a passive driver for vehicle A.

At the beginning of the simulation (time $t = 0$ s), vehicle A speeds up first before reducing the velocity at $t = 11$ s and stopping before reaching the crossing area. At this point, the fuzzy-logic-based approach decelerates the speed of vehicle A to avoid any collision with vehicle B, which has the right-of-way. Once vehicle B has cleared the intersection at $t = 31$ s, vehicle A starts moving at $t = 32$ s and turns left at the intersection. Vehicle D reaches the stop sign at $t = 5$ s. In this simulation, vehicle A appears as an object of interest for vehicle D since vehicle A drives in the lane into which vehicle D is planning to go. At $t = 10$ s, vehicle D starts to move since it detects no collision with vehicle A, which is at a stop position. Vehicle C reaches the stop sign at $t = 7$ s. Since vehicle C plans to go straight at the intersection, it has to wait until the intersection is cleared. Vehicle A crosses the intersection and exits the crossing area at $t = 41$ s, followed by vehicle C which starts to move at $t = 42$ s.

In the second simulation of scenario 1 we use the cost-based approach to control each vehicle. The distance-time diagram of the simulation is depicted in Figure 7(b). We can clearly see that the vehicles behave the same way as in the first simulation for scenario 1. The fuzzy-logic-based and cost-based approaches come to the same behavior for vehicle A which corresponds to the behavior of a passive driver. A passive driver, very conservative yields to vehicle B and thus avoid any collision. We can say that both approaches can mimic the decision that a passive driver would make in such situation.



(a) (b)

Figure 8. Distance-time diagrams for scenario 2 using an aggressive driver for vehicle A.

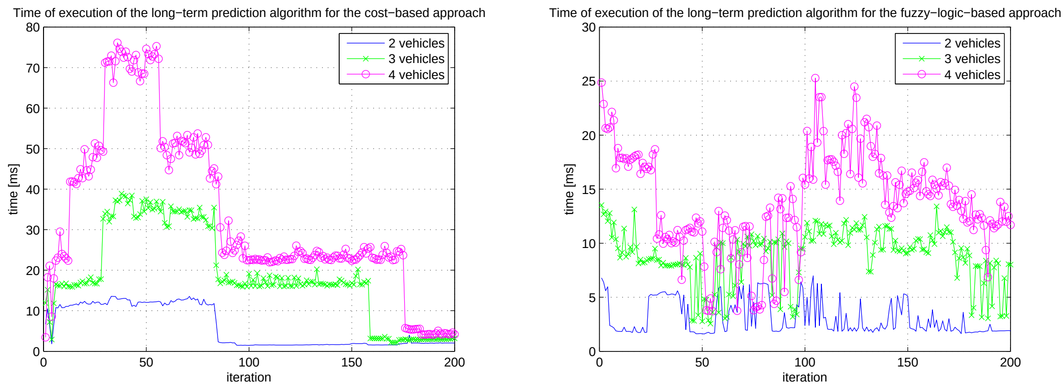
The first simulation of scenario 2 uses the fuzzy-logic-based approach. The distance-time diagram for this simulation is presented in Figure 8(a). At the beginning of the simulation, vehicle A accelerates to enter the crossing area at $t = 12$ s and exits the crossing area at $t = 18$ s. The fuzzy controller considers the rules that involve a vehicle with high aggressivity ("Aggressivity" is Aggressive) and then computes the final acceleration or deceleration. In this simulation, the fuzzy-logic-based approach speeds up vehicle A to cross the intersection before vehicle B reaches the crossing area. The acceleration of vehicle A is modified to perform a driving task while avoiding any collision with vehicle B. We note in Figure 8(a) that vehicle C leaves the stop position earlier ($t = 32$ s) than in scenario 1 ($t = 42$ s). Vehicle C waits for vehicle B to clear the intersection, which happens faster than the time waited for vehicle A to clear the intersection in scenario 1.

In the second simulation of scenario 2, we use the cost-based approach. The distance-time diagram for this simulation is shown in Figure 8(b). Although the driver of vehicle A is fully aggressive, the behavior for this vehicle is different from the one observed in the first simulation of scenario 2. In the cost-based approach, vehicle A detects vehicle B as an object of interest and considers a possible collision. Instead of accelerating as compared to the previous simulation, vehicle A decelerates and yields to vehicle B during the time period $[12 - 28]$ s. In this simulation, the cost-based approach does not speed up vehicle A to make it clear the intersection before vehicle B reaches the crossing area. We note that vehicle A enters the crossing area at $t = 29$ s where we can see the speed increasing. This step happens sooner than the ones we observed during scenario 1 ($t = 32$ s), due to the high aggressivity of the vehicle. The action of vehicle A to move earlier results in vehicle C moving earlier as well at $t = 38$ s compared to scenario 1 where C starts to move at $t = 42$ s.

The behavior of vehicle A is different in the two simulations of scenario 2. Vehicle A slows down and yields to vehicle B for the cost-based approach. Vehicle A speeds up to enter the crossing area and crosses the intersection before vehicle B reaches the crossing area for the fuzzy-logic-based approach. The way the cost-based approach uses the gathered information is different from the fuzzy-logic-based approach. In this scenario, vehicle A detects vehicle B as being close rather than being in a "degree of close" as conceived by the fuzzy-logic-based approach. The fuzzy-logic-based approach triggers different rules that involve an aggressive driver and then controls the velocity of vehicle A. The cost-based approach tests if the paths of vehicles A and B intersect and then computes when and where the collision is likely to happen. Since the driving procedure of the cost-based approach is defined by the action sequences, the actions performed in a driving task are limited. Consequently, each action sequence performed by vehicle A will lead to a future collision with vehicle B if vehicle A keeps on driving through the intersection. The fuzzy-logic-based approach catches the aggressivity of the vehicle and controls the acceleration to mimic a real collision-free behavior for an aggressive driver.

4.2. Time of Execution of the Long-term Prediction Algorithm

To test the performance of the cost-based and the fuzzy-logic-based approaches, we have computed the time of execution of the long-term prediction algorithm at an intersection. Figures 9(a) and 9(b) show the time of execution for the cost-based and the fuzzy-logic-based approaches, respectively. We have recorded the time of execution on two hundred iterations. One iteration represents one loop of the long-term prediction algorithm that computes one prediction for each vehicle. The time of execution has been computed for both approaches with two, three and four vehicles at the intersection. Figure 10 gives detailed information on the time maximum and the time average of execution.



(a) (b)

Figure 9. Time of execution of the long-term prediction algorithm for the cost-based and the fuzzy-logic-based approaches.

	Cost-based approach		Fuzzy-logic-based approach	
	Time maximum (ms)	Time average (ms)	Time maximum (ms)	Time average (ms)
4 vehicles	76.12	33.17	25.29	13.96
3 vehicles	38.91	18.68	13.52	8.87
2 vehicles	13.57	5.92	6.97	3.06

Figure 10. Time of execution for the cost-based and the fuzzy-logic-based approaches.

Figure 9 shows that the time of execution increases with the number of vehicles at the intersection. We can also distinguish higher peaks of time (time maximum) for the cost-based approach compared to the fuzzy-logic-based approach. Accurate values of the time maximum can be found in Figure 10. A closer look at the time average demonstrates that the cost-based approach is more computationally intensive than the fuzzy-logic-based approach during a driving task at an intersection.

5. CONCLUSION AND FUTURE WORK

In this paper we have described the evolution of the PRIDE architecture. To deal with problems characterized by the pervasive presence of uncertainty and vagueness in autonomous vehicles, we have combined a cost-based approach with a fuzzy-logic-based approach. We have pointed out in which situations the two approaches are triggered and we have showed how the approaches complement each other. We have tested the new system at traffic intersections with four vehicles. For passive drivers, the behaviors of the vehicles are the same, both approaches can simulate conservative driving. When we involved aggressive driver at the intersection, the cost-based approach did not catch the reality and mimics the behavior of passive drivers instead. The fuzzy-logic-based approach was able to use all the information to simulate the aggressive aspects of the drivers. The fuzzy-logic-based approach is able to approximate the input information and outputs the new acceleration or deceleration based on the aggressivity of the controlled vehicle. We also have computed the time of execution of the long-term prediction algorithm for each approach. We have showed that the cost-based approach is more computationally intensive than the fuzzy-logic-based approach for various number of vehicles at an intersection.

Although we have demonstrated the good performance of PRIDE with appropriate results, we still need to enhance this framework. Thus, we plan to introduce more vehicles into traffic situations, more complex road networks, and a better way to select objects of interest in highly populated environments. We also need to test our framework for situations of car following at intersections by avoiding traffic congestions. Before we release a version of PRIDE for public domain accessibility, we need to validate the results of the framework as compared to real world driving scenarios.

REFERENCES

1. E. D. Dickmanns, "The Development of Machine Vision for Road Vehicles in the Last Decade," in *Intelligent Vehicle Symposium*, **1**, pp. 268–281, (Versailles, France), June 17–21 2002.
2. "Darpa urban challenge." http://www.darpa.mil/GRANDCHALLENGE, 2007.
3. S. Hoseini, M. Vaziri, and Y. Shafahi, "Combination of Car Following and Lane Changing Models as a Drivers' Optimization Process," *Applications of Advanced Technologies in Transportation Engineering* **0-7844-0730-4**, pp. 601–605, 2004.
4. C. Schlenoff, J. Ajot, and R. Madhavan, "PRIDE: A Framework for Performance Evaluation of Intelligent Vehicles in Dynamic, On–road Environments," in *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, **12**, pp. 263–270, (Washington DC), August 2004.
5. A. Saffiotti, "The Uses of Fuzzy Logic in Autonomous Robot Navigation," *Soft Computing* **1**(4), pp. 180–197, 1997. Online at http://www.aass.oru.se/~asaffio/.
6. J. Albus et al., "4D/RCS Version 2.0: A Reference Model Architecture for Unmanned Vehicle Systems," Tech. Rep. NISTIR 6910, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2002.
7. C. Schlenoff, S. Balakirsky, A. Barbera, C. Scrapper, and J. Ajot, "The NIST Road Network Database: Version 1.0," Tech. Rep. NISTIR 7136, National Institute of Standards and Technology, July 2004.
8. C. Schlenoff, R. Madhavan, and Z. Kootbally, "PRIDE: A Hierarchical, Integrated Prediction Framework for Autonomous On–Road Driving," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2348–2353, (Orlando, FL, USA), May 15–19 2006.
9. C. Scrapper, S. Balakirsky, and E. Messina, "MOAST and USARSim: A Combined Framework for the Development and Testing of Autonomous Systems," in *Proceedings of the 2006 SPIE Defense and Security Symposium*, **6230**, (Orlando, FL, USA), April 17–21 2006.
10. Z. Kootbally, C. Schlenoff, and R. Madhavan, "A Brief History of PRIDE," in *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop, NIST SP 1073*, pp. 40–47, (Gaithersburg, MD), August 2007.
11. C. Schlenoff, Z. Kootbally, and R. Madhavan, "Driver Aggressivity Analysis within the Prediction In Dynamic Environments (PRIDE) Framework," in *Proceedings of the 2007 SPIE Defense and Security Symposium*, **6561**, p. 65611O, May 2007.
12. A. Saffiotti, K. Konolige, and E. H. Ruspini, "A Multivalued-Logic Approach to Integrating Planning and Control," *Artificial Intelligence* **76**(1-2), pp. 481–526, 1995. Online at http://www.aass.oru.se/~asaffio/.
13. H. Hu and D. Gu, "Reactive Behaviours and Agent Architecture for Sony Legged Robots to Play Football," *Industrial Robot: An International Journal* **28**(1), pp. 45–54, 2001.
14. C. Voudouris, P. Chernett, C. J. Wang, and V. Callaghan, "Hierarchical Behavioural Control for Autonomous Vehicles," in *Proceedings of the 2^{nd} IFAC International Conference on Intelligent Autonomous Vehicles*, pp. 267–272, Helsinki University of Technology, (Espoo, Finland), June 1995.
15. H. R. Beom and H. S. Cho, "A Sensor-based Obstacle Avoidance Controller for a Mobile Robot Using Fuzzy Logic and Neural Network," in *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, **2**, pp. 1470–1475, July 1992.
16. D. Driankov and A. Saffiotti, *Fuzzy Logic Techniques for Autonomous Vehicle Navigation*, Studies in Fuzziness and Soft Computing, Springer-Phisica Verlag, 2001.
17. L. A. Zadeh, "Fuzzy sets," *Information and Control* **8**, pp. 338–353, 1965.
18. C. C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Part I," Tech. Rep. UCB/ERL M89/90, EECS Department, University of California, Berkeley, 1989.
19. J. McKnight and B. B. Adams, "Driver Education Task Analysis Volume 1: Task Descriptions," Final Report 70-103, Human Resources Research Organization (HumRRO), 300 North Washington Street, Alexandria, Virginia 22314, November 1970.