Communications and Decisions of Autonomous Vehicles in Virtual Environments

Ronald R. Mourant, Zhishuai Yin, Yingzi Lin and Sagar Kamarthi

Northeastern University, Boston, MA

Abstract--In many virtual environments, autonomous objects, such as people and vehicles, are essential to increase the feeling of presence. A goal is to have autonomous objects behave as humans, or in the case of vehicles, as if humans controlled them. Such objects are known as intelligent autonomous objects. We present a combination of a communication model and a decision-making model to achieve the goal of modeling autonomous objects that behave intelligently. Both models are attached to autonomous objects that represent people and vehicles in a virtual environment. This enables such an autonomous object to be an independent entity that is self-motivated and self-controlled. These intelligent autonomous objects are able to communicate with other autonomous objects via their communication model according to decisions reached by their decision-making model. The decision-making model relies on the communication model to investigate possible outcomes before making decisions. The communication model defines senders, receivers, contents, and channels (media through which content is transferred) in real-time to gather desired information from specified objects. The decision-making model is divided into two levels, the global level and the local level. These work, respectively, with global information perceived by a perception model and the local information received by the communications model. A group of logic rules are formulated as decision trees to model the process of making decisions on the basis of real-time activities. We used traffic and people, in a virtual environments based driving simulator, as examples of intelligent communicating autonomous objects.

Index Terms--autonomous objects, communication, decision-making, logic rules

I. INTRODUCTION

Modeling urban environments as three-dimensional virtual environments may require the reproduction of real city traffic. This would entail aconsiderable amount of autonomous people and vehicles traveling on different paths in the virtual environment. In order to have these autonomous objects (AOs) behave somewhat like real people and real vehicles, past researchers [3, 6, 13] have used databases to store properties above the environment. Examples include 1) providing road speed limit information to autonomous vehicles, and 2) letting autonomous people know the state of traffic signals. Supplying such information gives AOs a low level of intelligence i.e., an autonomous vehicle knows how fast it should be going, and autonomous people know when to cross the street. This concept has been expanded to include storing topological and semantic information of roadways in databases [4, 14]. This has enabled autonomous vehicles to follow global routes while conforming to traffic rules (e.g. stop at stop signs and red traffic signals).

However, interactions between autonomous vehicles and roadways are not sufficient in terms of having a complete understanding of the driving environment. On real roadways, even if every single traffic rule is followed, there is still no guarantee of safe travel and efficiency. Drivers in the real world pay attention to other drivers on the roadway and react in a timely manner to avoid unsafe situations. Drivers act/react by observing others' behaviors in the real world. Thus, inter-vehicle communications are highly desired for AOs in virtual environments.

Some virtual environments include live human participants as well as AOs [5]. Thus there is also a need for communication between humans and AOs. Humans can be modeled as a special type of very intelligent AO [7]. This would enable them to be included in a communications model for all objects in virtual environments.

With information obtained from an in-memory array of roadway properties, and communications with other vehicles, an AO must react properly and timely, and its behaviors, in certain situations, could resemble those of humans. Note that AOs purposely have non-human behaviors in many virtual environments such as games. We use a perception model for describing the process of AOs receiving information from in-memory arrays.

Based on the knowledge of the environment that an AO has, and its communication with nearby objects (autonomous or human), the AO makes decisions regarding its direction and speed of travel. We use a decision model for this.

To react properly and timely, the AO must process information received and make correct decisions. Hence, a decision-making process is required to be executed after attaining in-memory array information and communicating with other AOs [6]. This paper proposes a decision-making model to address this problem.

Our paper is organized as follows: first, a brief introduction of the perception model that perceives the driving environment is given. Then, the communications model is described. This is followed by the details of the decision-making model. We then present a virtual environment that implements these models in order to verify their performance. Lastly, conclusions are presented and future research is discussed.

II.   THE PERCEPTION MODEL

A perception model is a first step towards having intelligent AOs. To move in virtual environments, AOs need to have considerable knowledge of the environment. We use in-memory arrays and a system of triggers to provide such information to AOs as they move through a virtual environment. We define "perception" as the process of AOs receiving information from in-memory arrays on a when needed basis.   Upon program initialization, and when a moving AO fires a trigger, the AO is provided such information as 1) its location in world coordinates, 2) its direction of travel in terms of heading information, 3) rules of movement (such as speed limits, and placement and status of traffic signals), and 4) location of obstacles. The in-memory arrays, which contains information about possible network paths and their characteristics, was populated when the virtual environment was initialized. Not only geometric information such as length, width, elevation, etc. of paths, but also topologic information such as connectivity among paths, and possible directions of path travel, etc. are stored in the in-memory arrays.   By means of perception (acquisition of information from in-memory arrays) AOs have state knowledge of their near-by surroundings. In order to move safely, they also need information about other moving AOs. For this we use the communications model.

III.   THE COMMUNICATIONS MODEL

The communications model simulates communicating behaviors between AOs [2]. This model is attached to every autonomous object in the environment without introducing any overall supervisory control system. Each AO is an independent entity that possesses the ability to communicate.

Usually, communication is described as a process of transferring information from one entity to another. Four indispensible elements of communication are defined as: sender, receiver, channel, and information to transfer. To have communication between AOs, the communications model, with these four elements, is attached (made a component) to each AO. As an AO travels through the virtual environment, the communications model updates these four elements in real-time with regards to the AO's instantaneous circumstances. Consider an AO named AO1. The way AO1's communication model works is as follows:

a) By default AO1 is a receiver.
b) Based on AO1's current heading and velocity, a group of autonomous vehicles that may possibly send messages to AO1 is determined. The determination of this group of possible senders (PSs) is based on the current position, heading, and velocity of the vehicle. The rational for this is that a driver in the real world would only want to pay attention to other drivers whose behaviors may affect his/her travel safety and efficiency. Similarly, an autonomous object should only communicate with objects that are worth communicating with. Communicating with all other autonomous objects in the virtual environment is neither necessary nor human-like.

c) Determination of the contents to be received from the senders. We use the fact that AO1 is a receiver by default. Also, every autonomous object only sends information to others upon request. First, AO1 determines what information it wants from each PS in its group of possible senders. AO1 then sends an information request to each PS.   Senders then respond by sending the requested information to AO1.
d) Choose the channel to send and receive information.
e) If AO1 receives a request from other autonomous objects, it sends the requested information.

Below, we consider the Communication Model in terms of autonomous vehicles in a driving simulation.

3.1 Determination of Possible Senders

Before an AO starts to communicate, the members of its possible senders group (PSG) has to be determined. Since an AO considers itself a receiver by default, it has to define some other AOs as senders at initialization time. A driver on real roads experiences communication with different vehicles as he/she travels along a route.   Our communications model mirrors this process by adding and subtracting from an AO's group of possible senders in real time.

In order to refer to an autonomous vehicle clearly, each roadway in the environment is transformed into a 2D array. In the 2D array, R_i stands for the i_{th} roadway. The m_{th} lane of the roadways is represented as $R_i[m]$, and the n_{th} autonomous vehicle on that lane is denoted as $R_i[m][n]$. Elements (autonomous vehicles) in $R_i[m]$ are sorted by their relative distances to the end of the lane. Operations enabled on the one-dimensional array $R_i[m]$ are: 1) insert an element into the array, 2) remove an element from the array and 3) sort elements in the array. Therefore, when the n_{th} autonomous vehicle in the m_{th} lane of roadway i, represented as $R_i[m][n]$, shifts to the adjacent $(m-1)_{th}$ lane, it will be removed from the array $R_i[m]$ and inserted into the array $R_i[m-1]$. Each autonomous vehicle has a unique ID, which is used for communication purposes. When an AO becomes a member of the senders group of another AO, its ID is added to the "senders array".   The array supports dynamic allocation, insertion and deletion.

To find senders, an AO perceives the driving environment around it. We will consider three driving scenarios: 1) on a straight roadway 2) at an intersection and 3) at a merge point. The model filters out non-relevant AOs by using its perception of the driving environment. We consider a specific A0 below that we have named, AO1.

3.1.1   On a Straight Roadway

When on a roadway and still far from an oncoming intersection, AO1 would mostly be affected by the autonomous vehicle ahead if it remains in the same lane. AO1 has to keep a safe distance from the autonomous vehicle ahead as well to maintain the desired speed. This is described as the following behavior, one of the most universal situations in daily driving. Let's say AO1 is the K_{th} autonomous vehicle on the j_{th} lane of roadway i, mathematically written as $R_i[j][k]$, it would only have to access the autonomous vehicle ahead, mathematically written as $R_i[j][k-1]$. The ID of autonomous

vehicle $R_i[j][k-1]$ is then added to the "senders array" of AO1. An autonomous vehicle is considered as an element in a lane even if only part of its 3D visual model is in that lane, e.g. an autonomous vehicle that is currently shifting from the adjacent lane.

However, AO1 might be unsatisfied with its speed in the current lane. As a result, it may intend to make a lane change instead of following the autonomous vehicle ahead of it. The decision of making a lane change is made by the decision-making model, which is described later in this paper. AO1 doesn't initialize a lane change right after it makes the decision. It has to communicate with other vehicles in order to find the appropriate time and place. In the case of a lane-changing behavior, the autonomous vehicle ahead in the same lane is no longer important. Instead, autonomous vehicles in the target lane require more attention. The following vehicle and the leading vehicle in the target lane are two vehicles that determine whether it's safe to start a lane change. Let's say AO1 is still the k_{th} autonomous vehicle on the j_{th} lane of roadway i and it intends to shift to the $(j-1)_{th}$ lane. The IDs of autonomous vehicles, which are closest to AO1 in the $(j-1)_{th}$ lane are added to the senders array of AO1.

3.1.2   At an Intersection

Another frequently encountered circumstance in driving simulations is when autonomous vehicles come to an intersection. Autonomous vehicles' intended paths often intersect at intersections. Other than conforming to traffic signs and signals, autonomous vehicles have to communicate with each other to avoid collisions and to continue on their planned routes. For example, at a stop sign before a "T" intersection, a driver should stop until the way ahead is clear. To realize the way ahead is clear, the driver has to check for other vehicles and/or pedestrians.

We require all autonomous vehicles coming to an intersection to strictly follow traffic signals and signs. Then, we only need to consider the autonomous vehicles whose intended courses of travel interfere with the route for our autonomous vehicle. Fig 1 shows such a scenario at an intersection.

Autonomous vehicle A intends to turn left while autonomous vehicle B intends to go straight. Once the traffic light for them turns green, there is an available time period for them to cross the intersection. Autonomous vehicles A and B share the same time period. Their intended routes physically intersect at point I. If communication is missing between them, they may come into conflict and cause a collision. At the same time, autonomous vehicle C on the left roadway plans to drive straight. Its intended route physically intersects with A's planned route at point J as well. However, because the traffic light for C is still on red, its time period to cross the intersection does not overlap with that of A. Therefore, even though A doesn't communicate with C, there would be no expected collision. As a conclusion, autonomous vehicle B is a sender of autonomous A, and also the other way around.

3.1.3   At a Merge Point

An autonomous vehicle that intends to merge onto a highway has to negotiate with vehicles in the target lane. The autonomous vehicle needs to recognize a gap between vehicles traveling in the lane, into which it wants to merge. This again emphasizes the necessity of defining senders dynamically in real-time.

3.2 Determine the Contents to be Received from Senders and Choose the Channel

Given a group of senders, the model has to determine the information that a receiver expects from its senders. The model starts the process of communication by sending requests to senders. Senders only reply to receivers upon
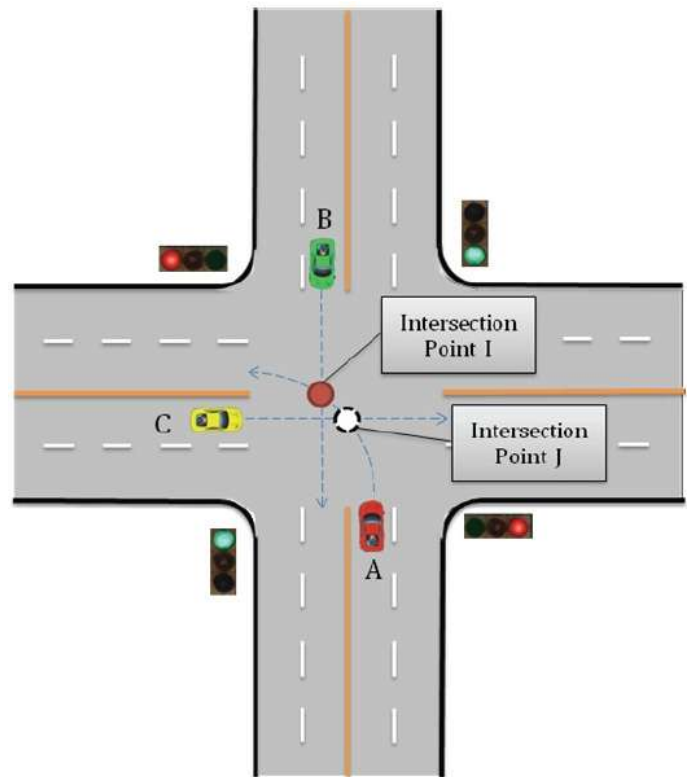

Fig. 1. Vehicle A intends to turn left while vehicle B intends to go straight at an intersection.

requests. By sending a request, the model clearly specifies what information is needed from whom and which receiver should the sender respond to. The request is expressed as: {Receiver ID, Sender IDs, Contents: xx, xx, xx}. Under different circumstances, an AO may require different information from senders. Typically, the speed and position of a sender is requested. A dynamic buffer is utilized as the channel to transfer information. The receiver sends the request to the dynamic buffer and corresponding senders read the request. Upon request by the receiver, senders send contents needed by the receiver to the buffer for the receiver to read it.

IV.   THE DECISION-MAKING MODEL

The decision-making model is responsible for making the best decision based on perceptions of the virtual environment and information received from other AOs. Decisions made by an

AO's decision-making model are carried out by a dynamics model associated with the AO.

The decision-making model does not only make decisions but also decides when and where to carry out these decisions. With information received from other AOs, the decision-model is capable of evaluating local conditions to see whether it's ok to safely implement these decisions.

To achieve the goal of making proper decisions at the proper time and location, the decision-making model is divided into two levels: 1) the global level and 2) the local level. At the global level, the model reaches a decision D. At the local level,

the model has to reach a decision whether to carry out the decision D or not. The global level is named global, because the model makes a decision based on perceptions of the global environment via the perception model. At the local level, the model reaches a decision on the basis of local information received from the communication model.

A "tree" data structure was used to model the process of decision-making [1]. With regards to different environments, different trees were utilized.
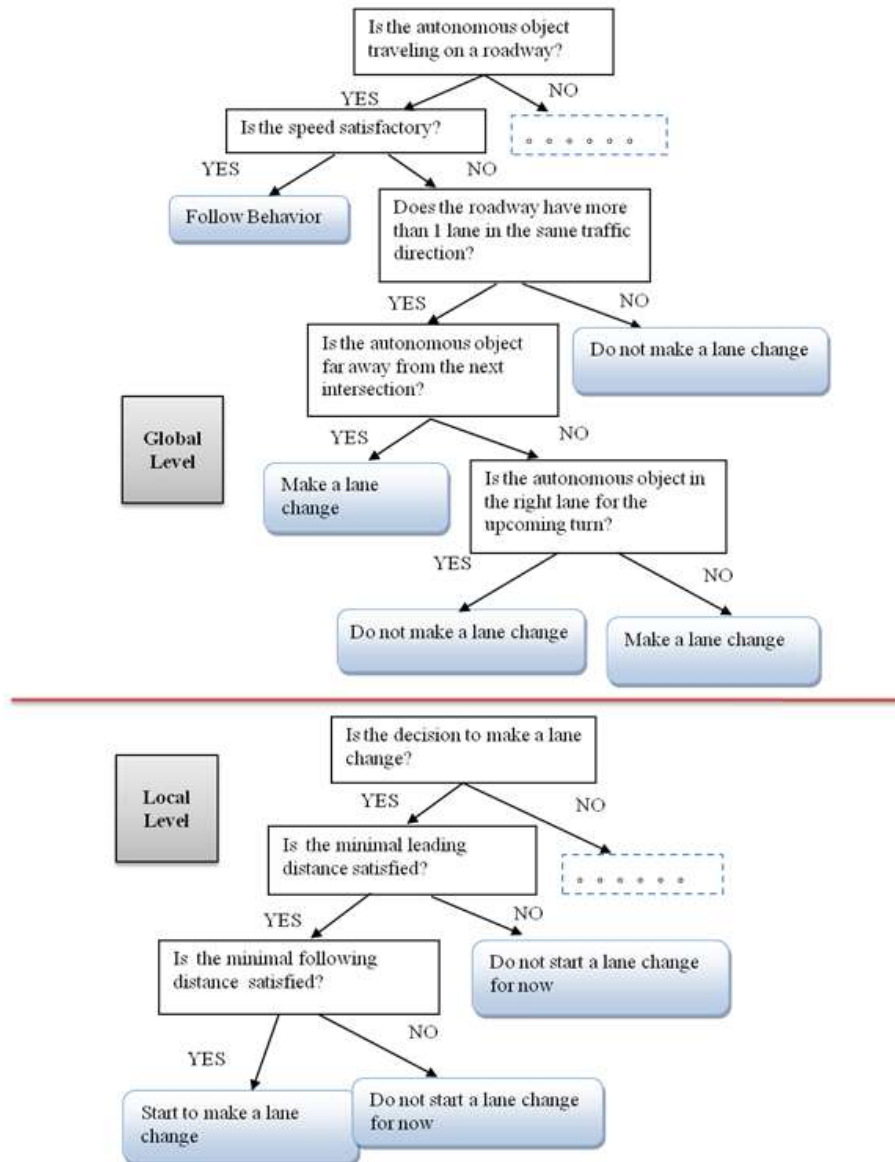
Fig. 2. A sample of the decision-making tree at the global level and the local level

Fig. 2 shows a sample tree that indicates how the decision-making model works at the global level and the local level. The tree is composed of linked nodes. Each node in the tree is either a parent node or a leaf node. Each parent node

represents a question that has a few possible answers. Each answer leads to one child node of that parent node. By stepping through the nodes of the tree, a leaf node is reached. A leaf node, which has no child node, represents a decision. Therefore, by reaching a leaf node, a decision is made. Once a

decision is made at the global level, the decision-making model goes to the local level and relies on the communication model to gather the needed information. An AO that intends to make a decision communicates with its senders defined by the communication model. With information sent from senders, the decision model steps through the tree at the local level, and finally reaches a leaf node that either permits the decision or provides an alternative decision.

For examples, we consider virtual traffic. Below, the logic rules constructed into the decision-making trees that handle lane changing behavior by an autonomous vehicle is presented.

4.1 Lane Changing Behavior

Lane changing is a common driving behavior. In virtual environments driving simulators, lane changes by autonomous vehicles help to produce more realistic and immersive scenarios. Here are logic rules considered at the global level by our decision model:

   a)   If an autonomous vehicle is traveling on a roadway with more than one lane in the same traffic direction and is far from the next intersection, and its speed is unsatisfactory, then the decision is to make a lane change.
   b)   If the autonomous vehicle is traveling on a roadway with more than one lane in the same traffic direction and is close to the next intersection, and is in the right lane for the upcoming turn, then the decision is not to make a lane change.
   c)   If the autonomous vehicle is traveling on a roadway with more than one lane in the same traffic direction and is close to the next intersection, and is not in the right lane for the upcoming turn, then the decision is to make a lane change.
   d)   If the autonomous vehicle is not traveling on a roadway with more than one lane in the same traffic direction, then the decision is to not make a lane change.

The meaning of "far" and "close" to the next intersection varies on each roadway for each autonomous vehicle. A method was proposed to divide the roadway into three zones [9]. In the zone that is "close" to the intersection, the autonomous vehicle has to get into the right lane to follow its global route. In the zone that is "far" from the intersection, the autonomous vehicle has the freedom to change its lane to seek a better driving condition. This paper adopts this method proposed in [9] and defines the term "close" based on the time to complete a lane change. For example, if an autonomous vehicle has to make *n* discrete lane changes to get into the correct lane, to get prepared for the upcoming turn, then the term "close" can be expressed mathematically as:

$$v * n * T * c \qquad (1)$$

is the speed of the autonomous vehicle, *n* is the number of single lane changes to make, *T* is the time needed to finish a single lane change and *c* is a constant factor. The constant factor c is always greater than 1. This would allow a vehicle to be in the proper lane for the turn before reaching the intersection.

With the four logic rules introduced above, the decision-making model has decided to make a change or not. It's not yet finished because the model has to decide which lane is the target lane [9]. If the autonomous vehicle only tries to get into the correct lane to get prepared for the upcoming turn, it would be easy to decide the target lane because it's already defined by the global route. For example, if the autonomous vehicle wants to turn right at the intersection, it has no other choice but to get into the most right lane. However, if the autonomous vehicle is seeking to travel at a faster speed, then, only using the perception of the driving environment, it's impossible to make such a decision. Assistance from the communication model is needed. By communicating with other autonomous vehicles in adjacent lanes, the autonomous vehicle gets knowledge about the traveling speed in other lanes. The lane with the highest traveling speed is then chosen as the target lane. Logic rules to be implemented are:

   e)   If the lane has the highest traveling speed among all other lanes in the same direction, select this lane as the target lane.
   f)   If none of the lanes has a higher traveling speed, then abandon the decision to make a lane change.

Until now, the decision to make a lane change or not has been reached at the global level. If the decision is to not make a lane change, then the decision-making model stops at this level. Otherwise, the model goes deeper into the local level. With the decision to make a lane change to the target lane (the I_{th} lane) in mind, the autonomous vehicle starts to communicate via the communication model with its senders in the adjacent lane (the I_{th} lane) towards the target lane. Notice that the target lane may not be one of the adjacent lanes. As described in the communication model, both position and speed of the leading vehicle and following vehicle in the j_{th} lane are sent to the autonomous vehicle. The minimum acceptable distances to the leading vehicle and the following vehicle are defined as follows:

$$D_l = (v_p - v_{lp}) * T + D_s$$
$$D_f = (v_{fp} - v_p) * T + D_s \qquad (2)$$

v_p is the vehicle's speed projected onto the traffic direction vector, v_{lp} is the leading vehicle's speed projected onto the traffic direction vector, v_{fp} is the following vehicle's speed projected onto the traffic direction vector, *T* is the time to finish a lane change and D_s is the safe distance between vehicles. If both D_l and D_f are satisfied, then it's safe to start changing the lane. Logic rules to be implemented at the local level are:

   g)   If both the distances D_l and D_f are satisfied, then starts to make a lane change, else, do not start

The final decision is reached after applying this logic rule.

4.2 Other Typical Driving Behaviors Handled in the Decision-Making Model

4.2.1 Following Behavior

Following behavior is a common situation in driving. To ensure the safety of travel, one should decelerate if the vehicle ahead is within a certain distance D_l. To ensure the efficiency of travel, one should also accelerate if the vehicle ahead is without a certain distance D_h. The logic rule implemented at the global level is:

h) If the autonomous vehicle is traveling on a roadway and its traveling speed is satisfactory, then the decision is to follow. Once the decision to follow is made at the global level, the decision-making model goes deeper into the local level. The logic rule to implement at the local level is:

i) If the distance to the vehicle ahead is smaller than D_l, then decelerate, else if the distance is larger than D_h, then accelerate, else, maintain the current speed. The communication model returns the speed and position of the autonomous vehicle ahead and hence the distance to it is easily computed. Apply the logic rule and a final decision will be achieved.

4.2.2 Intersection Behavior

At intersections, the decision-making model together with the communication model, plays an important role in resolving conflicts among autonomous vehicles. The final decision made by the decision-making model determines whether it's legal and safe to cross the intersection.

First, the autonomous vehicle perceives the driving environment to check the status of the traffic light.

j) If the autonomous vehicle is coming to an intersection and the traffic light is on red, then stop to wait

k) If the autonomous vehicle is coming to an intersection and the traffic light is on green, then cross the intersection

l) If the autonomous vehicle is coming to an intersection and the traffic light is on yellow and the autonomous vehicle is close enough to the intersection entry point, then cross the intersection.

m) If the autonomous vehicle is coming to an intersection and the traffic light is on yellow and the autonomous vehicle is not close enough to the intersection entry point, then stop to wait. By saying "the autonomous vehicle is close enough to the intersection entry point", we mean there is enough time for the autonomous vehicle to reach the entry point before the traffic light turns from yellow to red. On basis of its speed and current distance to the entry point, the decision-making model is capable of determining whether it's close enough. "Stop" is a driving behavior that slows the autonomous vehicle down to a complete stop. By reaching the leaf node of the decision-making tree at the global level, the model already has a conclusion about whether it's legal to cross the intersection. If it's not legal, the decision-making model stops at the global level and lets the decision of "stop to wait" be carried out. If the decision is to cross the intersection, the decision-making model starts to step through the decision-making tree at the local level to see whether the

n) autonomous vehicle has the priority to cross. As described in the communication model, an autonomous vehicle communicates with other objects whose intended paths interfere with its planned routes. Here, these objects are named as "conflicted objects" for convenience. Logic rules presented at the local level are:

o) If the autonomous vehicle has a priority over conflicted objects, then it has the priority

a) If the autonomous vehicle does not have a priority over conflicted objects and cannot reach the intersection point before the next conflicted object in line, then it does not have the priority

b) If the autonomous vehicle does not have a priority over conflicted objects and can reach the intersection point before the next conflicted object in line, then it has the priority

c) If conflicted objects all passed the intersection point, then the autonomous vehicle has the priority

d) If the autonomous vehicle has the priority, then cross the intersection, else, stop to wait. Priority, as introduced in the communication model, is the right to use the conflicting part of an intersection.

V.   IMPLEMENTATION

In order to verify the models proposed in this paper and compare their performance with driving in a simulator, a virtual urban environment with the above models was designed and programmed. Parameters and factors used in logic rules were given real values with reference to real driving experiences. A series of screenshots of the virtual urban environment (top view) is included in Fig 3 to show a lane-changing scenario initialized and governed by the combination of the perception model, the communication model and the decision-making model. As shown in Fig 3a, a number of autonomous vehicles are traveling on a six-lane roadway with three lanes in each traffic direction. The roadway is stored in the in-memory array as R_7. The right most lane is represented as $R_7[0]$ and the left most lane is represented as $R_7[5]$ in the communication model.

Table 1 lists the results perceived by the perception model with reference to the decision tree at the global level shown in Fig 2. As described in the Eq. 1, the term "close" is mathematically defined as: $v * n * T * c$ Here, c is assigned a value of 2, and T is set to 5 seconds. It's clearly illustrated in Table 1 that all autonomous vehicles shown in Fig 3a have already come into the "close" zone of the upcoming intersection. When judging whether the vehicle is in the correct lane for the upcoming turn, an exceptional vehicle was found. All other vehicles were already prepared for the next intersection with the exception of the vehicle referred as $R_7[1][3]$. The decision-making models of all vehicles shown here, came to the decision to not change lanes, except for vehicle $R_7[1][3]$. The decision tree at the global level indicates a lane change is required for vehicle $R_7[1][3]$.

The decision-making model of vehicle $R_7[1][3]$ goes deeper into the decision tree at the local level with a purpose to find a proper time and place to initialize the lane change. Steps involved are listed as follows:

1) The communication model of vehicle $R_7[1][3]$ chooses vehicle $R_7[2][1]$ and vehicle $R_7[2][2]$ as its senders. Both speeds and positions of these vehicles are sent to vehicle $R_7[1][3]$ via the communication model.

TABLE 1: DISTANCES OF VEHICLES SHOWN IN FIG 3A TO THE UPCOMING INTERSECTION

Vehicle	$R_7[1][0]$	$R_7[1][2]$	$R_7[1][3]$	$R_7[2][0]$	$R_7[2][1]$	$R_7[2][2]$	$R_7[2][3]$
On Roadway?	Yes	Yes	Yes	Yes	Yes	Yes	Yes
No. of lanes	3	3	3	3	3	3	3
Distance to the intersection (ft.)	144.4	170.6	187.0	219.8	164.0	193.6	249.3
Velocity (mph)	19.9	20.1	18.4	19.6	20.8	21.1	22.0
"close" zone (ft.)	293	295	270	288	305	310	322
Current lane	1_{th}	1_{th}	1_{th}	1_{th}	2_{th}	2_{th}	2_{th}
Global route	Straight	Straight	Straight	Turn left	Turn left	Turn left	Turn left

2) Distances of vehicle $R_7[1][3]$ to the leading vehicle $R_7[2][1]$ and the following vehicle $R_7[2][2]$ are computed, 5 feet and 24 feet respectively. Both D_l and D_f are also computed as defined in Eq. 2, 20 and 26 feet respectively. D_s is given a value of 16 feet, T is computed on basis of the vehicle's current speed, and v_p is the projection of the vehicle's speed onto the traffic direction. Results suggest that both D_l and D_f are satisfied. Refer to the decision tree at the local level shown in Fig 2, a decision of starting the lane change is therefore reached and carried out by the driving behavior model. Fig 3b shows a screenshot of the vehicle $R_7[1][3]$ beginning to make a lane change and Fig 3c shows a screenshot of the vehicle $R_7[1][3]$ finishing changing its lane from lane $R_7[1]$ to lane $R_7[2]$.

As reported in [10], the distribution of lane change times ranges from 3.41 to 13.62 seconds and a lane change time of 6 seconds was observed most frequently. Also, the distribution of inter-lane change distance ranged from 8.84 to 15.19 feet, and 12 feet was the average. In our proposed model, the time to finish a lane change was set to 5 seconds and the inter-lane change distance was set to 12 feet.

Results of real driving experiments done previously [12] indicated that all drivers concluded that it's dangerous to make a lane change if the time to collision with the following vehicle in the target lane is less than 6 seconds. When the time to collision with the following vehicle is more than 10 seconds, it's considered by all drivers to be completely safe to make a lane change. As a conclusion, a driver only starts to make a lane change if the following vehicle is not going to catch up with him/her in at least 6 seconds.

In our implementation, the minimal acceptable following distance is 24 feet. As shown in Table 1, the relative velocity of the following vehicle to the lane changing vehicle is about 2 mph. Therefore, the minim the lane change in our model is about (24*3600)/(5280*2)=8.18 seconds.
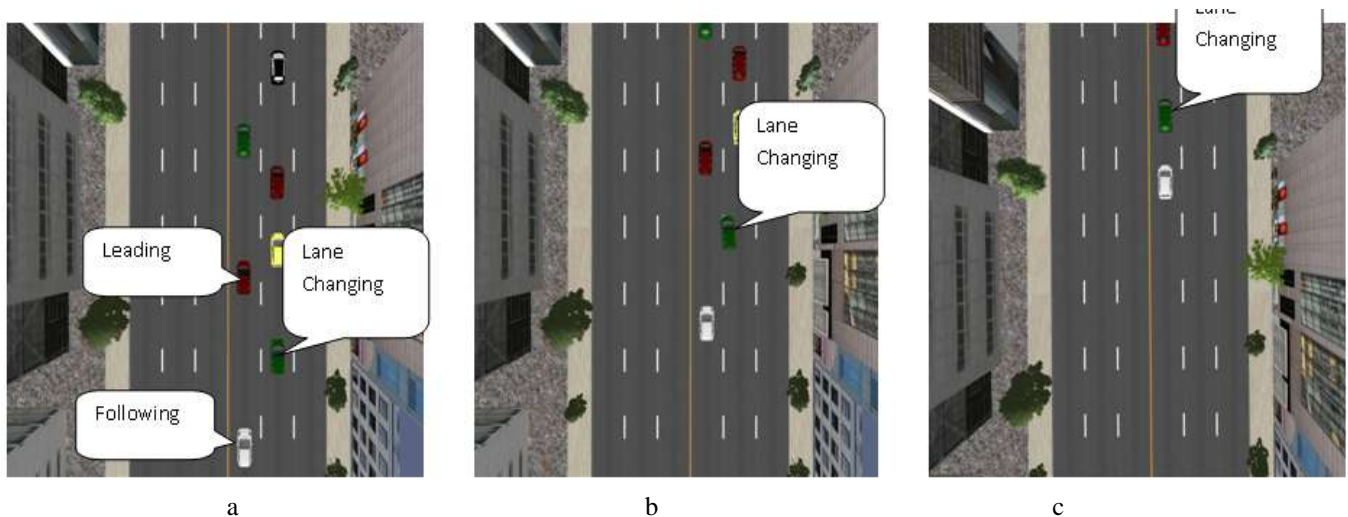

Fig. 3. A series of screenshots of a vehicle making a lane change in the virtual environment

VI. DISCUSSION AND CONCLUSION

The communication model and the decision-making model introduced in this paper is another step towards producing realistic traffic in driving simulators. The communication model improves the ability of autonomous vehicles to understand driving conditions by enabling effective inter-vehicle communications. It builds communications between autonomous vehicles by referring to the process of real communications. Autonomous vehicles know clearly about local traffic conditions as a result. Working in connection with the communication model, the decision-making model guarantees that autonomous vehicles act/react properly and quickly to changing circumstances. The decision-making trees at the global level and local level are a beginning to modeling the process of human decision making by incorporating a group of logic rules. The communication model and the decision-making model both can be revised easily to be adapted to new environments. By establishing new logic rules, the decision-making model is capable of handling new driving conditions.

With the implementation of these two models, the following can be achieved:

1) Simplified reproduction of real traffic in urban areas. Nowadays, traffic in cities is heavy. A great number
2) of vehicles/pedestrians are involved. To reproduce even part of these vehicles and pedestrians in a driving simulator could be quite difficult. Without inter-vehicle communications and self decision-making, designers may have to organize the traffic manually. Other than designing global routes for autonomous vehicles to follow, designers must also take local conditions into account. Workloads increase considerably as the complexity of traffic increases.
3) Enhanced immersion for the human operator in a driving simulator. Thanks to the improvements of intelligence and autonomy, autonomous vehicles can behave more naturalistic and flexible. Also, because each autonomous vehicle acts and reacts with regard to its instantaneous environment, it would not behave in the exactly same way (unless wanted by the designer) every time the simulator runs. Users who drive in the driving simulator for multiple times would be less likely to get bored in repeated scenarios.
4) More reliable testing results. Because of the enhanced immersion of the human operator and improved reality of the simulator, driving tests done with driving simulators would be expected to be more reliable and convincing.

Future research could consider the following. First, all autonomous vehicles in the above models behave at the same skill level. In other words, varieties of human drivers are not reflected in our models. Aggressive drivers should behave more aggressively as compared to moderate drivers. Skilled drivers surely have better controls over vehicle as compared to novice drivers. Second, autonomous vehicles perform ideally. For example, their deviations to the center of the lane do not vary. Also, the conformation to traffic rules is a hundred percent in the current model, which is not necessarily true in the real world. More unexpected scenarios would be generated if autonomous vehicles were allowed to violate traffic rules occasionall

REFERENCES

[1] Al-Shihabi, T., and R.R.Mourant. Toward More Realistic Driving Behavior Models for Autonomous Vehicles in Driving Simulators. *In Transportation Research Record: Journal of the Transportation Research Board,* Number 1843, 2003, pp. 41-49.

[2] Berder, O., Quemerais, P., Sentieys, O., Astier, J., Nguyen, T.D., Menard, J., Le Mestre, G., Le Roux, Y., Kokar, Y., Zaharia, G., Benzerga, R., Castel, X., Himdi, M., El Zein, G., Jegou, S., Cosquer, P., Bernard, M., Cooperative communications between vehicles and intelligent road signs**,** *ITS Telecommunications, 2008. ITST 2008. 8th International Conference*,24-24 Oct, 2008, pp.121 - 126

[3] Carles O., and Espie S., Database generation system for road applications, *Driving Simulation Conference*, 1999, pp. 87-103

[4] Donikian, S. (1997), VUEMS: a virtual urban environment modeling system，*In proceedings of Computer Graphics International,* 23-27, June, 1997, pp.84 – 92.

[5] Edward, L., Lourdeaux, D., Barthès, J.P., Lenne, D., Burkhardt, J.M. Modelling Autonomous Virtual Agent Behaviours in a Virtual Environment for Risk, *The International Journal of Virtual Reality, 2008, 7(3):13-22*

[6] Hongling Wang, Kearney, J.K., Cremer, J., Willemsen, P. Steering behaviors for autonomous vehicles in virtual environments, *In proceedings of Virtual Reality*, March, 2005, pp.155 – 162.

[7] Liu, Y., Zhong, S., Zhan, Y. Reasoning about action for behavioral animation of intelligent virtual agents, *The International Journal of Virtual Reality, 2008, 7(3):37-42*

[8] Michon, J.A. A critical view of driver behavior models: What do we know, what should we do? In: Evans, L., and Schwing, R. eds. *Human Behavior and Traffic Safety*, New York, Plenum Press, 1985, 485-520.

[9] P.G. Gipps. A model for the structure of lane changing decision. *Transportation Research Board*, 1986, pp. 403-414.

[10] Shannon H. Examination of driver lane change behavior and the potential effectiveness of warning onset rules for lane change or "side" crash avoidance system, Master's Thesis, Virginia Polytechnic Institute & State University, 1997.

[11] Tomer Toledo, Charisma F., Choudhury, Moshe E., Ben-Akiva. Lane-Changing Model with Explicit Target Lane Choice , *In Transportation Research Record: Journal of the Transportation Research Board,* Volume 1934, 2005, pp. 157-165

[12] Wakasugi T. A study on warning timing for lane change decision aid systems based on driver's lane change maneuver, Japan Automobile Research Institute, pp. 05-0290

[13] Willemsen, P., Kearney, J.K., Wang, H. Ribbon networks for modeling navigable paths of autonomous agents in virtual environments, *Visualization and Computer Graphics*, IEEE Transactions on Volume 12, Issue 3, May-June, 2006, pp.331 – 342

[14] Yan W., Weijian X., Design of virtual vision for a driving simulator, *In proceedings of the 2003 International Conference on Robotics, Intelligent Systems and Signal Processing*, Changsha, China, Oct, 2003, pp. 244-24

Ronald R. Mourant is a Professor in the Mechanical and Industrial Engineering Department at Northeastern University in Boston, Massachusetts. He is also the Director of the Virtual Environments Laboratory. His research interests include driving simulators, novice drivers, and games that enhance driving performance.

Zhishuai Yin is a Ph.D. Candidate in the Mechanical and Industrial Engineering Department at Northeastern University in Boston, Massachusetts. He is interested in building virtual driving environments that enhance the interactions between intelligent autonomous vehicles and human controlled vehicles.

Yingzi Lin is an Assistant Professor in the Mechanical and Industrial Engineering Department at Northeastern University in Boston, Massachusetts. She is also the Director of the Intelligent Human-Machine Systems Laboratory. Professor Lin's research interests include Human Machine System Integration and Evaluation, Adaptive Human Machine Interface Design, and Noninvasive Intelligent Sensors and Smart Sensing Systems.

Sagar Kamarthi is an Associate Professor in the Mechanical and Industrial Engineering Department at Northeastern University in Boston, Massachusetts. His research interests include neural networks and knowledge based systems in design and manufacturing, distributed and cooperative AI applied to systems integration monitoring, diagnosis and control of machining processes, and intelligent sensors/sensor integration.

.