

# Avoidance of Moving Obstacles Through Behavior Fusion and Motion Prediction

N. H. C. Yung & C. Ye

Department of Electrical & Electronic Engineering, The University of Hong Kong  
Chow Yei Ching Building, Pokfulam Road, HONG KONG SAR

## ABSTRACT

In this paper, we propose a novel approach of fusing the fuzzy control actions of the obstacle avoidance, goal seeking and steering behaviors, in which the steering behavior is derived from motion prediction. As such, the navigator is more capable to steer clear of the zone of high collision probability. Through simulation, it has been confirmed that the navigator having this steering behavior can tackle multiple moving obstacles successfully at much higher speed compared with those without. Furthermore, it does not require any a priori knowledge of the obstacles' motion.

## 1. INTRODUCTION

Navigation of an Autonomous Mobile Vehicle/Robot (AMV) in a dynamic environment may be viewed as an issue of motion planning in the presence of moving obstacles and/or in which the topological properties of the environment is changing. Published results indicated that the problem of avoiding moving obstacles is substantially harder than the problem of avoiding static obstacles as both time and space need to be considered simultaneously [1].

Of the many methods developed such as path planning in Time-Space configuration, trajectory prediction and motion prediction, complexity and the knowledge of the obstacles' trajectories have always been the major issues. In [2-5], although the complexity of the planning was dealt with to some extent, a priori knowledge of the obstacles' motion was also assumed. In terms of on-line navigation in a dynamic environment with unknown obstacle motions, such assumption is no longer true and therefore most research effort has been focused almost entirely on the problem of estimating or predicting the obstacles' trajectories. From these estimated obstacle trajectories, a motion planner is often used to modify the AMV trajectory to avoid collisions. As it is often assumed that the planner has no knowledge of the obstacle trajectories before the motion begins, it is necessary for both the analysis of obstacle trajectories and the revision of the motion plan to be carried out in real time.

In motion prediction, Wang and Tsai [6] used a least-mean-square-error classification scheme to calculate the trajectories of moving obstacles. The environment they studied was a hallway with moving obstacles constrained to linear paths. Based on the

obstacle trajectories, they applied the Tcyhonievich's method [7] to adjust the AMV velocity to create a collision-free path, and modify it as new information became available. Along this line of thoughts, similar statistical methods using statistical features to estimate obstacle locations were proposed by other researchers, such as the Hidden Markov stochastic model to predict the motion of obstacles [8], Poisson distribution to describe the probability of collision with obstacles [9], or the occupancy probability of each element in a grid map [10]. In general, these estimation techniques are complex in computation, and so far, have only been implemented for 2-D polygonal environments.

As observed, many of these methods have restrictions imposed on the obstacles' motion. For instance, they must have constant velocity, or must have known motion, or have a priority order. When the obstacle motion is assumed unknown, statistical or probability models are often used to predict the obstacle motion. The success of these approaches hinges on the accuracy of the prediction as well as how rapid the prediction can be achieved. Computational complexity is often a problem associated with these approaches. In our research, we challenge this problem by defining a steering behavior via motion prediction. The fuzzy control action of this behavior is then fused with the obstacle avoidance and goal seeking behaviors before the final crisp values are defuzzified. In our approach, obstacle motion is predicted through estimating the AR coefficients by fitting a 1<sup>st</sup> order AR model to the sequence relating the current acceleration and position of the moving obstacle in a least square sense, where its future position is predicted based on the coefficients. From the predicted motion, a new steering behavior is formed based on the collision cone concept. The result of fusing the fuzzy control actions of the three behaviors is that the navigator can now steer clear of high collision probability zones, and yet still maintain its course towards the goal. It has been demonstrated in simulation that the avoidance ability of the navigator is much enhanced when navigating through a dynamic obstacle course, without requiring any a priori knowledge of the obstacle motions.

This paper is organized as follows: Section 2 presents the motion prediction of moving obstacles. Section 3 illustrates how the steering behavior is being determined from the predicted obstacle motion, and how it is fused with the obstacle avoidance and goal seeking behaviors. Section 4 assesses the performance of the new navigator having the steering behavior through a number of simulation trials. A comparison is made

between the performance of the original and the new navigators. This paper is concluded in Section 5.

## 2. MOTION PREDICTION

Suppose an AMV is to navigate from a start configuration  $s_1$  to a destination  $g_1$  in an operating space consisting of  $m$  moving obstacles  $B_1, B_2, \dots,$  and  $B_m$  as depicted in Fig. 1. Let  $r_k(t)$  defines the actual position of  $B_k$  at time  $t$ , where  $r_k(t) = (X_k(t), Y_k(t))^T$ , and  $\hat{r}_k(t)$  defines the predicted position of  $B_k$  at time  $t$ . From its previous positions  $r_k(t-1), r_k(t-2), r_k(t-3)$  and etc., in theory  $r_k(t)$  can be predicted by using an  $n$ th order AR model [11]:

$$r_k(t) = \sum_{i=1}^n \alpha_i r_k(t-i) + \omega(t) \quad (1)$$

where  $\omega(t)$  denotes the prediction error according to [11],  $n (\leq t)$  is the number of the previous locations considered, and the coefficient  $\alpha_i$  is a  $2 \times 2$  matrix, for  $1 \leq i \leq n$ . Note that  $r_k(0)$  is the initial position of the AMV and  $\omega(0)=0$  initially.

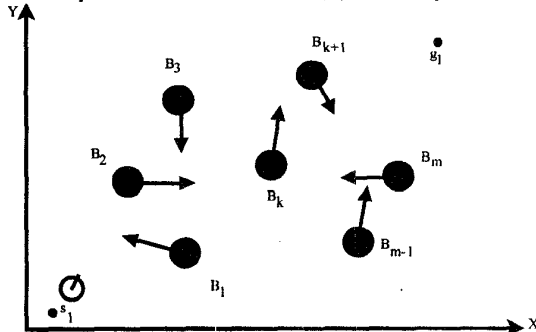


Fig. 1: Navigation through a dynamic obstacle course

Let  $\Delta t$  denote the time interval between two time units. In this time interval, assuming the acceleration of the obstacle  $B_k$  is slowly changing, therefore, it is reasonable to model its acceleration,  $a_k(t)$  with a 1<sup>st</sup> order AR model as follows:

$$a_k(t) = \beta_{k,t} a_k(t-1) + e_k(t) \quad (2)$$

where  $e_k(t)$  is the prediction error. The coefficient  $\beta_{k,t}$  can be a scalar value, or a diagonal or general matrix. For instance, if the motions along X and Y are correlated,  $\beta_{k,t}$  is represented by a general matrix. In all cases,  $\beta_{k,t}$  can be either dependent or independent of time  $t$ . If  $\beta_{k,t}$  is independent of time, it needs to be calculated only once. In general,  $\beta_{k,t}$  is dependent of time and is estimated adaptively as new distance measurements are made available through the distance sensors. In terms of the obstacle's velocity and position, the AMV's acceleration,  $a_k(t)$ , can be calculated by:

$$a_k(t) = \frac{1}{\Delta t} [v_k(t) - v_k(t-1)] = \frac{1}{\Delta t^2} [r_k(t) - 2r_k(t-1) + r_k(t-2)] \quad (3)$$

where  $v_k(t)$  stands for the velocity at time  $t$ . Substituting (3) into (2), we have:

$$r_k(t) - (2 + \beta_{k,t})r_k(t-1) + (2\beta_{k,t} + 1)r_k(t-2) - \beta_{k,t}r_k(t-3) = \Delta t^2 e_k(t) \quad (4)$$

Assume  $\beta_{k,t}$  changes with time  $t$ , an adaptive algorithm is desired to update  $\beta_{k,t}$  each time when a new set of measurement is available. The algorithm in [12] is adopted to determine  $\beta_{k,t}$  as follows:

$$\hat{\beta}_{k,t} = \arg \min_{\beta_{k,t}} \sum_{i=4}^t \lambda^{t-i} [a_k(i) - \beta_{k,t} a_k(i-1)]^T [a_k(i) - \beta_{k,t} a_k(i-1)] \quad (5)$$

where  $\lambda$  is a weighting factor,  $0 < \lambda \leq 1$ . For a slowly changing acceleration,  $\lambda$  is close to 1. The solution for this problem is thus

$$\hat{\beta}_{k,t} = \frac{\lambda \Delta_{k,t-1} + a_k^T(t) a_k(t-1)}{\lambda R_{k,t-1} + a_k^T(t-1) a_k(t-1)} \quad (6)$$

Since  $a_k(t)$ ,  $\Delta_{k,t}$ ,  $R_{k,t}$  and  $\lambda$  are all known, Eq.(6) can be solved. Therefore, the next position of  $B_k$  at time  $(t+1)$  is predicted by the following equation, which is derived from Eq.(2) & (3):

$$\hat{r}_k(t+1) = r_k(t) + v_k(t) \Delta t + \hat{\beta}_{k,t} a_k(t) \Delta t^2 \quad (7)$$

## 3. BEHAVIOR FUSION

### Moving Obstacle Model

To ensure collision-free navigation, the path planned by the AMV must not intersect the Collision Zone (CZ) of the obstacles as defined below: Let the AMV radius be  $R_v$ , and  $D_s$  be a safety margin representing the minimum measurable distance of the AMV sensors. The CZ of an obstacle is calculated as the area after the expansion of the obstacle's perimeter by  $R_v + D_s$ , as depicted in Fig. 2. Using the CZ concept, the AMV can be reduced to a single point as shown in Fig. 3.

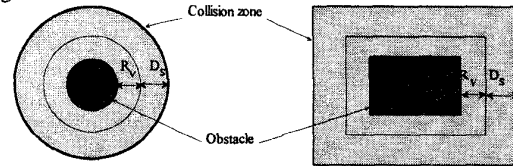


Fig. 2: Collision zone around an obstacle

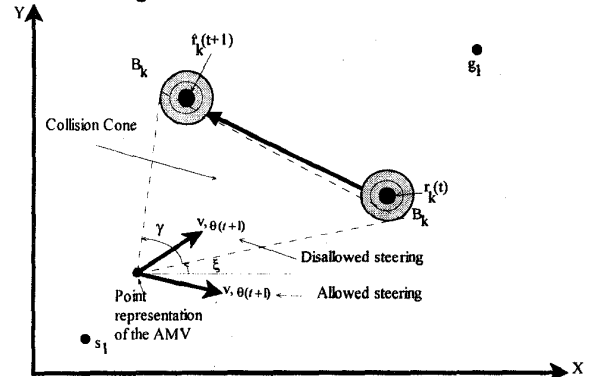


Fig. 3: Definition of the collision cone

Considering an obstacle  $B_k$ , assume its next position can be predicted using Eq.(7), the Collision Cone (CC)  $C_k$  with respect to  $B_k$  is defined by drawing tangents to the CZ of the

obstacle at  $r_k(t)$  and  $\hat{r}_k(t+1)$ , as depicted by the broken lines in Fig. 3. If the next heading angle of the vehicle,  $\theta(t+1)$ , where  $\theta(t+1) = \theta(t) + \Delta\theta(t)$  and  $\Delta\theta(t)$  is the steering angle, falls within  $C_k$  then the possibility of a collision is considerably higher than if it falls outside  $C_k$ . From this argument, the concepts of allowed steering angle and disallowed steering angle are defined below:

**Disallowed Steering Angle:**  $\xi + \gamma - \theta(t) \geq \Delta\theta(t) \geq \xi - \theta(t)$

**Allowed Steering Angle:**  $\pi/2 > \Delta\theta(t) > \Delta\theta_r$ ,

where  $\Delta\theta_l = \max(\xi - \theta(t), -\pi/2)$  and  $\Delta\theta_r = \min(\gamma + \xi - \theta(t), \pi/2)$ .

In essence, without considering the AMV and obstacle's velocities, the AMV heading angle can be further governed by the allowed/disallowed steering behavior. In a multiple moving obstacle environment, the AMV only considers the CC of the obstacle that has  $\hat{r}_k(t+1)$  nearest to it at any time instance.

### Fusing the Fuzzy Control Actions

The navigation architecture described in this section is developed from the one discussed in [13]. In the new navigator, fuzzy control actions from the obstacle avoider, goal seeker, navigation supervisor and motion predictor are fused. Using this approach, the strength of the original navigator is retained and yet its ability in handling moving obstacle is being enhanced because of the additional steering behavior. As depicted in Fig.4, the predicted location is used to define the collision cone and subsequently the fuzzy sets and eventually the steering behavior.

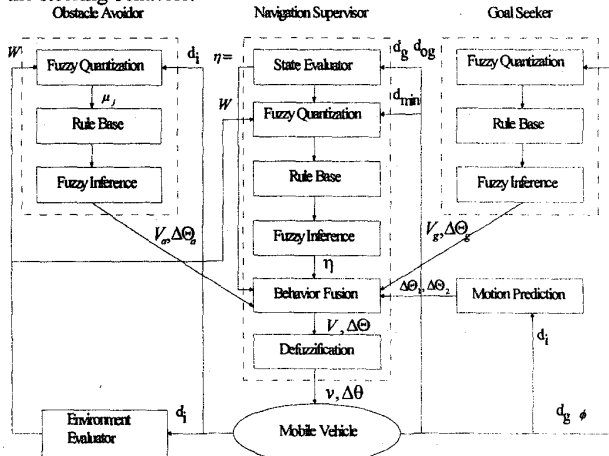


Fig. 4: New navigator with motion prediction

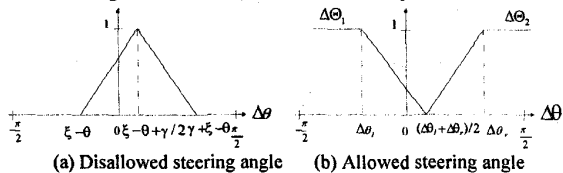


Fig. 5: Fuzzy representations of allowed and disallowed angles

At navigation step  $t$ , from the predicted next locations, the obstacle nearest to the AMV is used to construct the collision

cone. Supposed the current heading of the vehicle is  $\theta(t)$ , its next heading is given by  $\theta(t) + \Delta\theta(t)$ . The disallowed and allowed steering angles can be represented by the fuzzy set as depicted in Fig. 5(a) & (b) respectively, where  $\Delta\Theta_1$  and  $\Delta\Theta_2$  represent the left and right allowed steering angles.

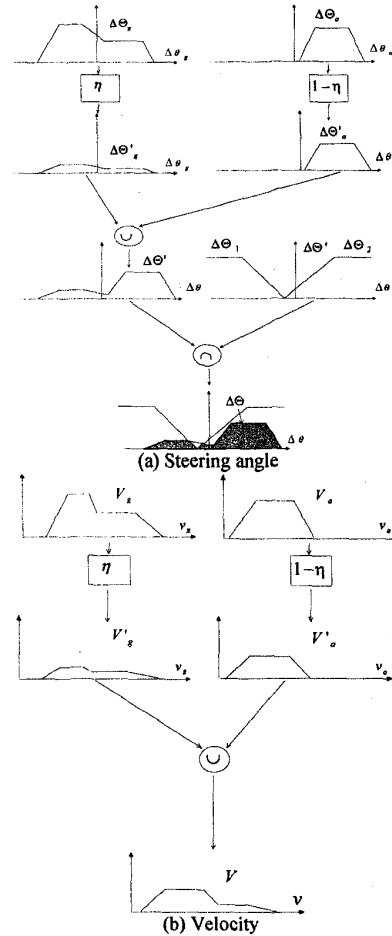


Fig. 6: Behavior fusion of the fuzzy control actions

The implementation of the behavior fusion is depicted in Fig. 6. As shown in Fig 6(a), the fuzzy sets of the steering angle inferred by the Obstacle Avoider and the Goal Seeker,  $\Delta\Theta_a$  and  $\Delta\Theta_g$ , are weighted by  $(1-\eta)$  and  $\eta$  respectively, and yield  $\Delta\Theta'_a$  and  $\Delta\Theta'_g$ . Both  $\Delta\Theta'_a$  and  $\Delta\Theta'_g$  are aggregated using an appropriate t-conorm operator [14] (the 'max' operator is used) to yield the fuzzy set  $\Delta\Theta'$ . Considering the allowed steering determined by the Motion Predictor,  $\Delta\Theta'$  is aggregated with  $\Delta\Theta_1$  and  $\Delta\Theta_2$  using the intersection operation and resulted in two fuzzy sets, of which the one with the larger area,  $\Delta\Theta$ , is taken as the final fuzzy set of the steering angle,  $\Delta\theta(t)$ .

The fusion of the fuzzy velocity is shown Fig. 6(b), the fuzzy sets of the velocity inferred by the Obstacle Avoidor and the Goal Seeker,  $V_a$  and  $V_g$ , are weighted by  $(1-\eta)$  and  $\eta$  respectively, which yield  $V'_a$  and  $V'_g$ . Both  $V'_a$  and  $V'_g$  are aggregated using the 'max' operator and yield the fuzzy set  $V$ . Finally, the fuzzy sets  $V$  and  $\Delta\Theta$  are defuzzified using the method of height defuzzification.

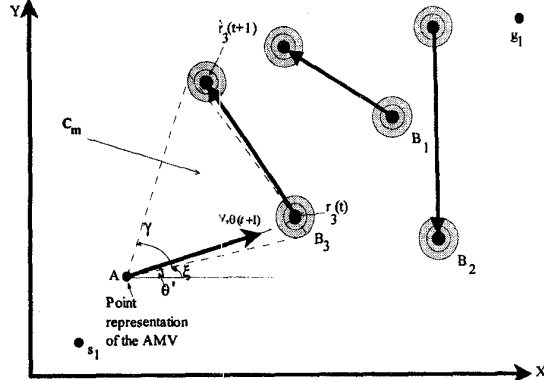


Fig. 7: Obstacle course with three moving obstacles

#### Four Navigation Scenarios

Consider the obstacle course as depicted in Fig. 7. Assume obstacle  $B_3$  has  $\hat{r}_3(t+1)$  nearest to the AMV at location A and its current heading is  $\theta(t)$ , there are four possible fusion scenarios to be considered.

**Scenario 1 ( $\theta(t)=\theta'$ ):** This is the case when the AMV heading towards the center of  $B_3$  at  $\hat{r}_3(t+1)$ . For convenience sake, a trapezoid fuzzy set  $\Delta\Theta'$  is used to represent the fuzzy value of the steering angle  $\Delta\theta(t)$  without considering the obstacle motion for the time being. Assume that the obstacle is 'far', then  $\Delta\Theta'$  is symmetrical about the axis of the membership value. This means that the crisp value of the steering angle defuzzified from the fuzzy set  $\Delta\Theta'$  is  $\Delta\theta(t)=\Delta\theta_1(t)=0$ . When the obstacle motion is taken into account by the fuzzy set aggregation as depicted in Fig. 8(a), it yields the shaded fuzzy set  $\Delta\Theta$  for  $\Delta\theta(t)$ . The final crisp value of  $\Delta\theta(t)$  defuzzified from  $\Delta\Theta$  is  $\Delta\theta_2(t)$  for  $\Delta\theta_2(t)\leq 0$ . This gives the AMV a new heading direction towards the right, moving away from  $C_m$ .

**Scenario 2 ( $\theta(t)\leq\xi$ ):** In this case, the crisp value of  $\Delta\theta(t)$  defuzzified from the fuzzy set  $\Delta\Theta'$  is  $\Delta\theta_1(t)$  for  $\Delta\theta_1(t)\leq 0$ , i.e. without considering the obstacle's motion, the AMV will go ahead if the obstacle distance is 'far' or turn right otherwise. When the motion of the obstacle is taken into account and the fuzzy set aggregation is implemented as depicted in Fig. 8(b), the shaded fuzzy set  $\Delta\Theta$  is taken as the fuzzy set for  $\Delta\theta(t)$ . The final crisp value of  $\Delta\theta(t)$  defuzzified from the fuzzy set  $\Delta\Theta$  is  $\Delta\theta_2(t)$  for  $\Delta\theta_2(t)\leq\Delta\theta_1(t)$ , meaning the vehicle will turn at least

$|\Delta\theta_1(t)|$  right resulting in a new heading outside  $C_m$ . The value of  $\Delta\theta_2(t)-\Delta\theta_1(t)$  increases or decreases when  $\theta(t)$  decreases or increases.

**Scenario 3 ( $\xi+\gamma>\theta(t)>\xi$ ):** In this case, the heading angle of the AMV is within  $C_m$ . It can be divided into two cases:  $\theta>\theta(t)>\xi$  and  $\xi+\gamma>\theta(t)>\theta'$ . The first case is similar to Scenario 2, which is omitted here for conciseness. For the second case, when  $\xi+\gamma>\theta(t)>\theta'$ , the crisp value of  $\Delta\theta(t)$  defuzzified from the fuzzy set  $\Delta\Theta$  is  $\Delta\theta_1(t)$ ,  $\Delta\theta_1(t)\geq 0$ , i.e. without considering the obstacle's motion, the AMV will go ahead if the obstacle is 'far' or turn left otherwise. When the motion of the obstacle is taken into account and the fuzzy set aggregation is implemented as depicted in Fig. 8(c), the area of the fuzzy sets  $\Delta\Theta'_1$  and  $\Delta\Theta'_2$  is determined by  $\theta(t)$ . When  $\theta(t)$  increases/decreases, the area of  $\Delta\Theta'_2$  increases/decreases and the area of  $\Delta\Theta'_1$  decreases/increases. Fig. 8(c) only shows the situation that the area of  $\Delta\Theta'_2$  is larger than that of  $\Delta\Theta'_1$ . In this case,  $\Delta\Theta'_2$  is taken as the fuzzy set for  $\Delta\theta(t)$  and is defuzzified to produce the steering angle  $\Delta\theta_2(t)$ , where  $\Delta\theta_2(t)>\Delta\theta_1(t)$ , which causes the AMV to head outside  $C_m$ . When  $\theta(t)$  is small enough such that the area of  $\Delta\Theta'_1$  is larger than that of  $\Delta\Theta'_2$ ,  $\Delta\Theta'_1$  is chosen as the fuzzy set for  $\Delta\theta(t)$ . As the crisp value of  $\Delta\theta(t)$  defuzzified from  $\Delta\Theta'_1$  may be less than 0, the AMV may choose to turn right, i.e.,  $\Delta\theta_2(t)<0$ , to avoid the obstacle, as it seems easier to avoid the obstacle by turning right. In the case that these two areas are equal, one of the fuzzy sets is chosen such that  $\Delta\theta(t)$  is the same as  $\Delta\theta(t-1)$  to keep a continuous steering.

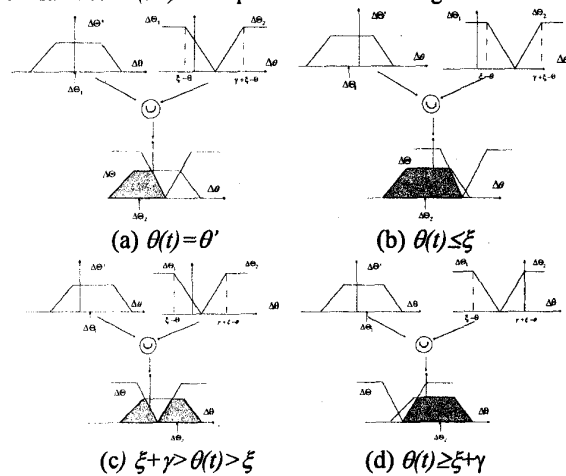


Fig. 8: Fuzzy sets aggregation of the four navigation scenarios

**Scenario 4 ( $\theta(t)\geq\xi+\gamma$ ):** In this case, the crisp steering angle defuzzified from the fuzzy set  $\Delta\Theta'$  is  $\Delta\theta_1(t)\geq 0$ , the vehicle will go ahead if the obstacle is 'far' or turn left otherwise. When the motion of the obstacle is taken into account and the fuzzy set aggregation is implemented as depicted in Fig. 8(d), the shaded

fuzzy set  $\Delta\Theta$  is produced as the fuzzy set for  $\Delta\theta(t)$ . The final crisp value of  $\Delta\theta(t)$  defuzzified from fuzzy set  $\Delta\Theta$  is  $\Delta\theta_2(t)$ . for  $\Delta\theta_2(t) \geq \Delta\theta_1(t)$ , meaning that the vehicle will turn at least  $\Delta\theta_1(t)$  left resulting in the AMV heading further away from  $C_m$ . The value of  $\Delta\theta_2(t) - \Delta\theta_1(t)$  increases/decreases when  $\theta(t)$  decreases/increases.

In summary, the fusion of the steering behavior with the two existing behaviors enables the AMV to always steer away from the Collision Cone. By doing so, the probability of a collision is reduced.

#### 4. SIMULATION TRIALS

To verify the navigator's ability in tackling multiple moving obstacles, simulations with two moving obstacles were carried out. In the simulation, two obstacles were moving one behind the other from west to east at a constant velocity, while the AMV was moving from south to north. The two possible situations that can determine the trajectory of the AMV is that (1) the start configuration is towards the right of the 1<sup>st</sup> obstacle and (2) the start configuration is towards the left of the 1<sup>st</sup> obstacle. For the first situation, the 2<sup>nd</sup> scenario ( $\theta(t) \leq \xi$ ) was satisfied, and therefore, the AMV turned right as it moved closer to the 1<sup>st</sup> obstacle. This is depicted in Fig. 9(a). Fig. 9(b) renders the same scene after the AMV had successfully avoided the 1<sup>st</sup> obstacle and moved toward its goal.

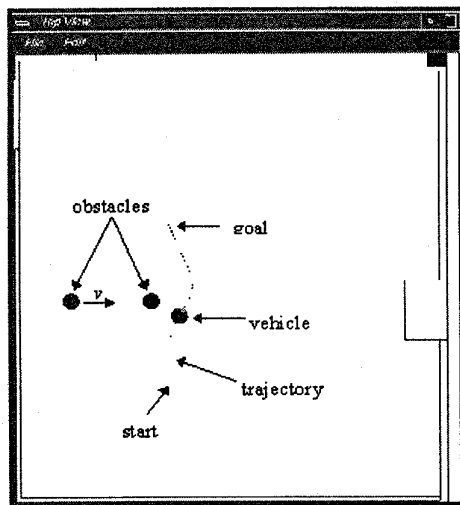


Fig. 9(a): Move toward right

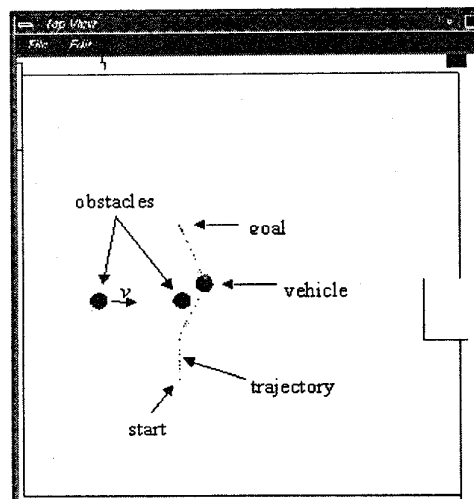


Fig. 9(b): Passed the first obstacle

For the second situation, the 4<sup>th</sup> scenario ( $\theta(t) \geq \xi + \gamma$ ) was initially satisfied, and therefore the AMV attempted to steer away from the collision cone of the 1<sup>st</sup> obstacle. It turned left as depicted in Fig. 10(a). As the 1<sup>st</sup> obstacle moved further to the right hand side, at some point, the 2<sup>nd</sup> obstacle had the next predicted location nearest to the AMV, which constituted to a new collision cone. With the new collision cone, scenario 2 ( $\theta(t) \leq \xi$ ) was satisfied which forced the AMV to turn right as depicted in Fig. 10(b).

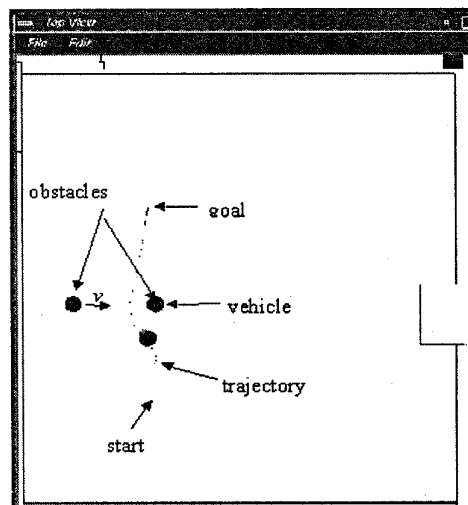


Fig. 10(a): Move toward left

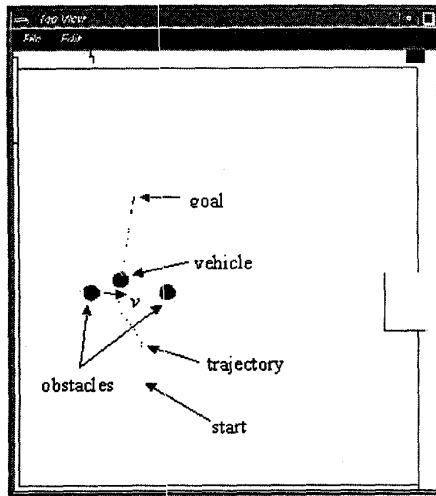


Fig. 10(b): Passed between two obstacles

Further simulation trials show that collision occurs if the obstacle's velocity is greater than 63% of the vehicle's velocity. Comparing this with the 38% of the original navigator [13], this illustrates the improved obstacle avoidance of the new approach. It also illustrates the point that based on the approach of the motion prediction and behavior fusion, the relative velocity between the obstacles and the vehicle plays an important part in determining how well the vehicle can avoid collision. Of course, the density of moving obstacles will have significant impact on the vehicle's avoidance ability too. Its exact effect is yet to be considered.

## 5. CONCLUSION

In conclusion, we have demonstrated two important issues in moving obstacle avoidance in this paper. Firstly, we have shown that obstacle motions can be predicted through an autoregressive model, from which a new steering behavior of the navigator can be established. Secondly, we have illustrated that by fusing the fuzzy control actions of this new behavior with the obstacle avoidance and goal seeking behaviors, the moving obstacle avoidance ability of the navigator can be substantially improved. Both these points have been verified through a number of simulation trials. Such improvement can be attributed to the ability of the navigator to predict the next position of the obstacle nearest to it, as well as the fusion of the fuzzy control actions of the three behaviors. Based on these results, two aspects will be investigated in the future. First, the effect of having varying obstacle velocities during the avoidance event should be studied. Second, the impact of obstacle density to the navigator's ability to navigate freely without collision is one important and yet practical issue that needs to be resolved.

## REFERENCES

- [1] J. C. Latombe, "Robot motion planning," Kluwer Academic Publishers, 1991.
- [2] J. Reif, "Complexity of the mover's problem and generalizations," in J. F. Canny and J. Reif, "New lower bound techniques for robot motion planning," in Proceedings of 20th IEEE Symposium on Foundations Computer Science, pp. 421-427, 1979.
- [3] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: the path-velocity decomposition," The International Journal of Robotics Research, vol. 5, no. 3, pp. 72-89, 1986.
- [4] P. Fiorini, and Z. Shiller, "Motion planning in dynamic environments using the relative velocity paradigm," in Proceedings of IEEE International Conference on Robotics and Automation, pp. 1610-1615, 1987.
- [5] K. Fujimura, "Time-minimum routes in time-dependent networks," IEEE Transactions on Robotics and Automation, vol. 11, no. 3, pp.343-351, June 1995.
- [6] L. Wang and W. Tsai, "Collision avoidance by a modified least-mean-square-error classification scheme for indoor autonomous land vehicle navigation," Journal of Robotics Systems, vol. 8, pp.677-698, Oct. 1991.
- [7] L. Tychonievich, "A maneuvering-board approach to path planning among moving obstacles," in Proceedings of 11th International Joint Conference on Artificial Intelligence Applications, 1989.
- [8] Qiuming Zhu, "Hidden Markov model for dynamic obstacle avoidance of mobile robot navigation," IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp.390-396, June 1991.
- [9] R. Sharma, "Locally efficient path planning in an uncertain, dynamic environment using a probability model," IEEE Transactions on Robotics and Automation, vol. 8, no. 1, pp.105-110, February 1992.
- [10] H. P. Moravec, "Sensor fusion in certainty grids for mobile robots," AI Magazine, vol. 9, no. 2, pp. 61-74, 1988.
- [11] N. Ketharnavaz and S. Li, "A collision-free navigation scheme in the presence of moving obstacles," in Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, pp. 808-813, 1988.
- [12] M. J. Shena, "Recursive least squares lattice algorithm - a geometrical approach," IEEE Transactions on Automatic Control, Vol. 26, pp. 695-702, June 1981.
- [13] Yung H. C. & Ye C., "An Intelligent Mobile Vehicle Navigator based on Fuzzy Logic and Reinforcement Learning", to appear in *IEEE Transactions on Systems, Man and Cybernetics*, 1998.
- [14] D. Driankov, H. Helledoom, and M. Reinfrank, "An introduction to fuzzy control," Springer-Verlag, ISBN 3-540-56362-8, 1993.