Vision-Based Fuzzy 2D Motion Control of a Model Helicopter

S. A. Tadayoni, B. Gholami, K. Demirli

Mechanical Engineering Department,
Concordia University, Montreal, Canada

***Abstract*— In this paper the 2D motion of a model helicopter is studied. The position control of a 2D model helicopter falls into complex nonlinear problems domain which makes it rather hard. In this paper a fuzzy controller is proposed to stabilize the helicopter on a designated target. The 3D model of the helicopter is simplified to derive a 2D model and all the states are assumed to be measurable. Based on the model, it was possible to decouple the position and orientation control of the helicopter. However the two parts are related through a fuzzy rule base. To verify the proposed method a simulation program is written in C++ which takes advantages of OpenGL to enable having 3D features. The response graphs are then presented.**

***Index Terms*— Fuzzy Control, Model Helicopter, Position Control.**

I. Introduction

The interest in Unmanned Aerial Vehicles (UAVs) has growth enormously during the past two decades. The main reason is that small autonomous helicopters can perform tasks comparable to man-controlled and can replace the use of humans in hazardous or time-consuming tasks such as fire fighting in forests and search and rescue missions. Recently, numerous researchers have investigated the idea of cooperative control of multiple UAVs (see e.g. [1]).

On the other hand, fuzzy control has proved to be a useful design technique to design controllers for nonlinear systems, where classical nonlinear controllers are hard to design. The idea of fuzzy control of helicopters has been investigated by a number of authors [2][3].

In this paper we apply a fuzzy-logic-based controller to a model helicopter. Model helicopters have higher maneuverability compared to the actual size helicopters and therefore their control we would be more demanding. Obviously, with higher maneuverability it comes a higher potential for use of such helicopters in more demanding missions. We use a Mamdani type reasoning and use a subjective modelling to control the helicopter. As a first step for controlling a model helicopter in 3 dimensions, we confine ourselves to the plane motion of the helicopter. We also consider a number of simplifying assumption to model the helicopter.

In section II we describe the basic equations of motion of the helicopter, where the basic equations of motion of the helicopter in a plane is derived from the full model. We propose our fuzzy controller in section III followed by computer simulation in section II-C. Conclusions and direction for future works are stated in section IV.

II. Helicopter Dynamics

A. Deriving the Equations of Motion of the Helicopter in a Plane

In this section the equations of motion of a helicopter will be briefly described based on [4]. Then the derivation of the 2D model from the full model will be described. We assume that all the states are measurable and the position measurement is done by a vision system observing the helicopter moving on the plane. This is due to the setup being developed at the Control and Information Systems Lab at Concordia University. This is not a confining assumption since if we use GPS for measuring the global position we would end up with the same situation discussed in this paper.

We consider the helicopter as a rigid body and we define a coordinate system fixed to its center of mass as shown in the Figure 1.

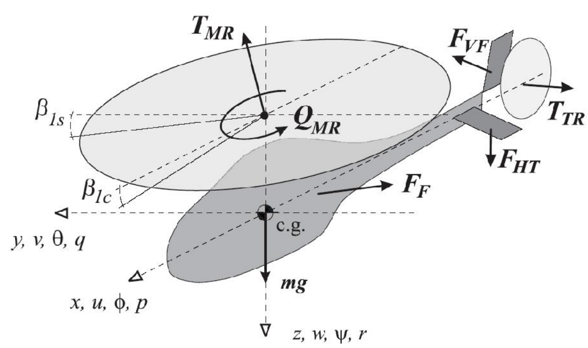

Fig. 1. Moments and forces acting on a helicopter [4]

The equations of motion of a helicopter, which is essentially the equations of motion of a rigid body, can be described by Newton-Euler equations shown below [4]

$$
\begin{aligned}
\dot{u} &= vr - wq - g\sin\theta + (X_{mr} + X_{fus})m \qquad (1)\\
\dot{v} &= wp - ur + g\sin\phi\cos\theta + (Y_{mr} + Y_{fus} + Y_{tr} + Y_{vf})/m\\
\dot{w} &= uq - vp + g\cos\phi\cos\theta + (Z_{mr} + Z_{fus} + Z_{ht})/m\\
\dot{p} &= qr(I_{yy} - I_{zz})/I_{xx} + (L_{mr} + L_{vf} + L_{tr})/I_{xx}\\
\dot{q} &= pr(I_{zz} - I_{xx})/I_{yy} + (M_{mr} + M_{ht})/I_{yy}\\
\dot{r} &= pq(I_{xx} - I_{yy})/I_{zz} + (-Q_e + N_{vf} + N_{tr})/I_{zz}
\end{aligned}
$$

In the above formula $()_{mr}$ stands for the main rotor; $()_{tr}$ for the tail rotor, $()_{fus}$ for the fuselage, $()_{vf}$ for the vertical fin and

1-4244-0363-4/06/$20.00 ©2006 IEEE

$()_{ht}$ for the horizontal stabilizer. Q_e is the torque produced by the engine to counteract the aerodynamic torque on the main rotor blades. X , Y and Z stand for the components of the corresponding forces and L, M and N the components of the corresponding torques on the axis x, y and z of the fixed-body-axis, respectively. Components of the linear and angular velocities and accelerations is shown in the figure.

Since we are considering the 2D motion of the helicopter the only remaining degrees of freedom would be forward and lateral motion, corresponding to x and y and change in the yaw angle of the helicopter, *i.e.* θ. Furthermore, we assume that the helicopter is moving in low speeds and therefore the effects of the drag force and moment can be neglected. Also, the effect of the forces and moments produced by the vertical fin and the horizontal stabilizer can also be neglected.

The force components of the main rotor is described by the following equations

$$X_{mr} = T_{mr} \sin a_1 \tag{2}$$

$$Y_{mr} = T_{mr} \sin b_1 \tag{3}$$

where T_{mr} is the main rotor trust given by $T_{mr} = f(\Omega)$, Ω being the main rotor's angular velocity and a_1 and b_1 are the longitudinal and lateral tilts of the tip path plane of the main rotor with respect to shaft, respectively. On the other hand, the y component of the tail rotor's force is equal to the tail rotor trust

$$Y_{tr} = T_{tr} = f(\omega) \tag{4}$$

where ω is the tail's rotor angular velocity and the torque produced by the tail rotor is given by

$$N_{tr} = Y_{tr} l \tag{5}$$

l being the distance from the tail rotor to the center of mass of the helicopter. If we assume that Ω is constant we can take X_{mr}, Y_{mr} and Y_{tr} as the inputs of the system. The changes in the aforementioned parameters are provided by changes in a_1, b_1 and ω respectively, which can be considered as the true inputs to the helicopter. Following a simple substitution in the original equations of motion of the helicopter in the 2D plane we end up with the dynamic model of the helicopter

$$\dot{u} = vr + u_1 \tag{6}$$

$$\dot{v} = -ur + u_2 + u_3 + \alpha \tag{7}$$

$$\dot{r} = u_3 \tag{8}$$

where u_1, u_2 and u_3 are the control inputs of the system and α is a constant. It would be straightforward to find the relationship between those and the actual physical inputs to the helicopter. The interested reader should refer to [Gavrilets] for a detailed description.

B. Relationship between the Global Position of the Helicopter and its Velocity in the Fixed-Body Axis

As described earlier, the velocities described by 6 are described in the fixed-body axis of the helicopter. Since the measurements are available in the global coordinates (either by a camera observing the helicopter from above or the position measurement using GPS) we should describe the relationship between them. Using simple trigonometrical calculations, this relationship is given by

$$\dot{x} = u \cos\theta + v \sin\theta \tag{9}$$

$$\dot{y} = u \sin\theta + v \cos\theta \tag{10}$$

$$\dot{\theta} = r \tag{11}$$

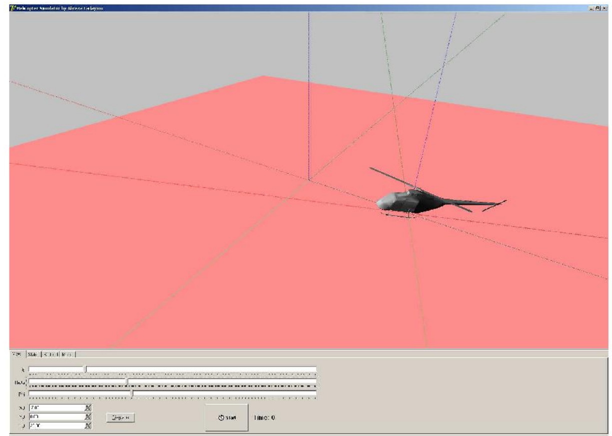

Fig. 2. The 3D Screen shot of the simulation software.

Combining (6) and (9) the dynamic equation of motion of our system is described by

$$\begin{aligned} \dot{x} &= u \cos\theta + v \sin\theta \\ \dot{y} &= u \sin\theta + v \cos\theta \\ \dot{\theta} &= r \\ \dot{u} &= vr + u_1 \\ \dot{v} &= -ur + u_2 + u_3 + \alpha \\ \dot{r} &= u_3 \end{aligned} \tag{12}$$

Our strategy is to generate the inputs u_1 and u_3 corresponding to forward motion and change in the orientation of the helicopter by fuzzy reasoning and use a simple feedback law to calculate for u_2. This is due to the fact that forward motion and change in the orientation are sufficient inputs for the system to steer the helicopter to any point on the plane. This simplifies the control design process considerably. Therefore, our approach is a combination of a simple classical feedback control law for u_2 and fuzzy control for u_1 and u_3.

C. Computer Simulation

In order to simulate the proposed method on the dynamic system (12) a simulation program is written with C++. Using OpenGL enables us to add 3D features to the program. The initial value problem to be solved, *i.e.* differential equations (12) with the given initial condition, was solved using first order Euler method with time steps of 1_{e-3} time units. Figure 2 shows a snapshot from the 3D model resulting from the program.

III. Fuzzy Controller

In this section the fuzzy controller is described and simulation results are presented. First a definition of the problem

1) If the distance is positive accelerate to the origin.
2) If the distance is negative decelerate from the origin.
3) If the velocity is positive stop (negative break).
4) If the velocity is negative stop (positive break).

TABLE I
SIMPLEST FUZZY RULES

is given and in then the rules of the fuzzy controller are presented.

The problem is defined as followed:

> *"Given the model (12) and assuming that all the states are available for measurements, design a controller to regulate the helicopter around origin, starting with a non-zero initial condition."*

The fuzzy controller used in this paper is based on subjective modelling. The rules are based on human reasoning and the Mamdani type reasoning is used. The decoupled dynamics of the helicopter provides us with the possibility to design decoupled rules for the orientation and the position. So the rules are separated to two main parts which the first part is aimed to regulate the position and the second part is supposed to regulate the orientation. Although the rules are somehow decoupled but there is one rule that relates the position and the orientation to each other. It says that

> *"if the direction of the helicopter is pointing at the origin move to the origin, otherwise turn towards the origin".*

In the subsection III-A the position regulation rules are described, while the orientation regulation rules are described in subsection III-B. Then the relating rules are described in subsection III-C.

A. Position Regulation Rules

In this section it is considered that the helicopter is targeting to the origin and the only goal is to regulate the position around origin. The global idea behind the rules is shown in Table I.

If the same magnitude is used for the negative and positive break, we will have some neutral regions in the state space where the negative and positive break command will cancel out and the velocity remains constant. The membership functions are defined in Figures 10 to 13.

These four rules simply results in the stability of the system around the origin. To describe that, one should note that the role of these fuzzy rules are two-folded: Go to the origin and at the same time stop. Therefore, the control output will be zero only in the case that these two are satisfied (the helicopter is at the origin or the helicopter is moving towards the origin and the two commands cancel out each other). The system response to the *Simplest Fuzzy Rules* is shown in Figure 3 and the relationship between the input u_1 and the distance and velocity is shown in Figure 4.

One can note that although the controller accomplishes the task for which it is designed for, it still needs improvements

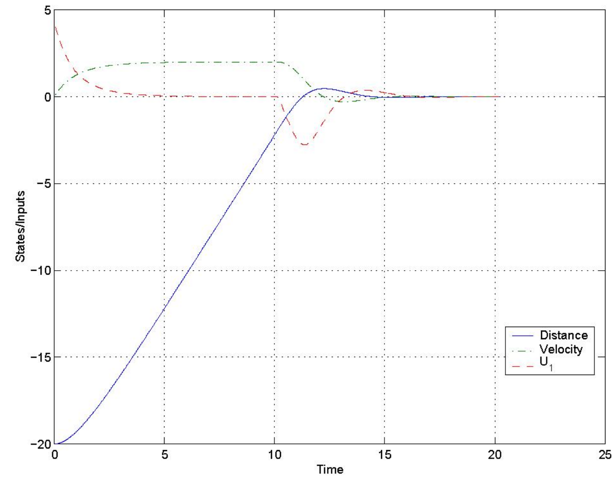

Fig. 3. System Response - Equal Acceleration and Break.

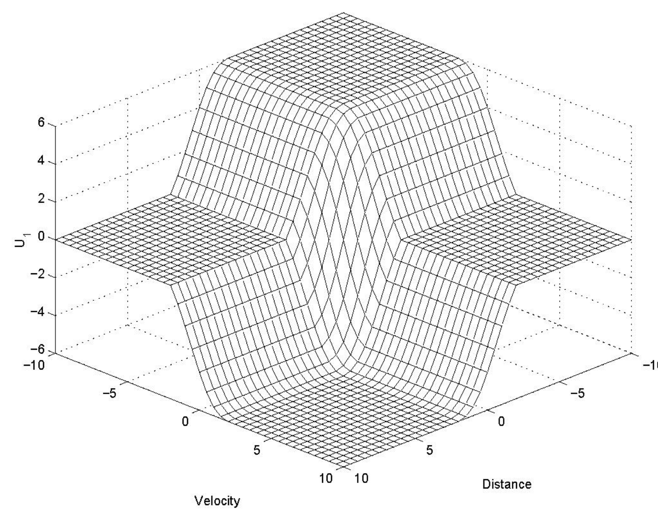

Fig. 4. Position Rules Surface - Equal Acceleration and Break.

in its performance. More specifically, since the controller is trying to satisfy both of the objectives at the same time, *i.e.* move towards the origin and stop, we are not using the full potential of our actuator capabilities. This results in the situation where the helicopter should accelerate to reach the origin in a shorter time, but the other objective doesn't allow for it.

This unnecessary break can be solved by defining a *Close Condition* and change the rules accordingly:

> *"If Velocity is Positive and Distance is Close, then Break."*

This results in faster convergence to the origin and improves the transient response. A consequence of this new improvement is that the system has a bigger overshoot in the response. We could expect this since now when the helicopter enters the close region, it has a greater velocity than what it had before and therefore it needs a break with a higher strength. The response of the system for the case where we have added a close condition is shown in Figure 5 and the control surface in the Distance-Velocity-u_1 space is shown in Figure 6.

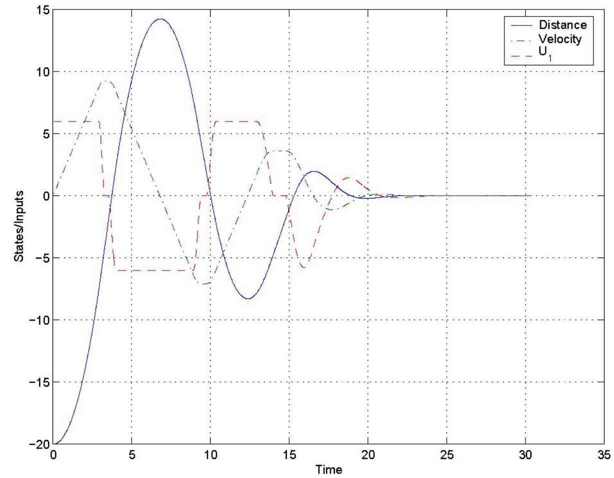

Fig. 5. System Response - Equal magnitude of acceleration and break with the close condition.

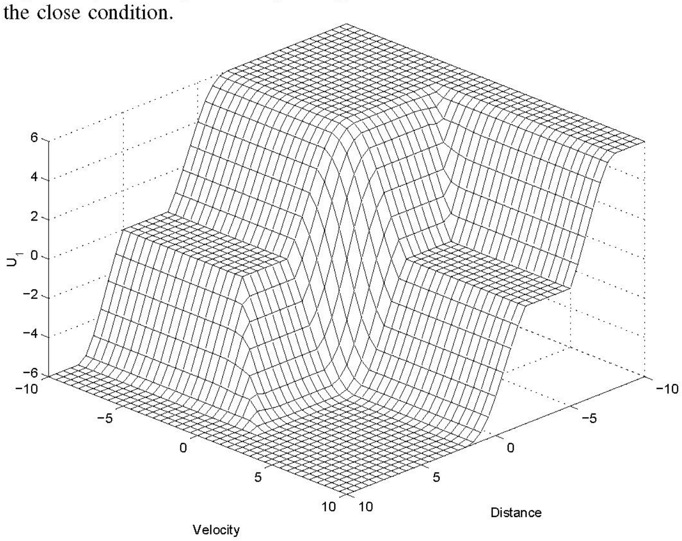

Fig. 6. Position Rules Surface - Equal magnitude of acceleration and break with the *Close* condition.

Now to solve the overshoot, we need a greater magnitude of breaking compared to the acceleration strength. Another improvement would be not to let the velocity of the helicopter to increase too much, while we are relatively far from the origin and there is no break in action. The latter rule makes more sense when we realize that the speed is increasing beyond the desired level, so by the time helicopter reaches near the origin the velocity is too high that we can stop the helicopter with acceptable performance.

Now to solve the overshoot, we need a greater magnitude of breaking compared to the acceleration strength. Another improvement would be not to let the velocity of the helicopter to increase too much, while we are relatively far from the origin and there is no break in action. The latter rule makes more sense when we realize that the speed is increasing beyond the desired level, so by the time helicopter reaches near the origin the velocity is too high that we can stop the helicopter with acceptable performance.

"If the distance is negative/positive & the velocity is small, then accelerate forward/backward to the origin."

The control surface for the new rules and the system response is shown in Figures 7 and 8, respectively. The response graph show improvement in the performance compared to the previous cases. The new rules surface and the system response are shown in Figures 7 and 8.

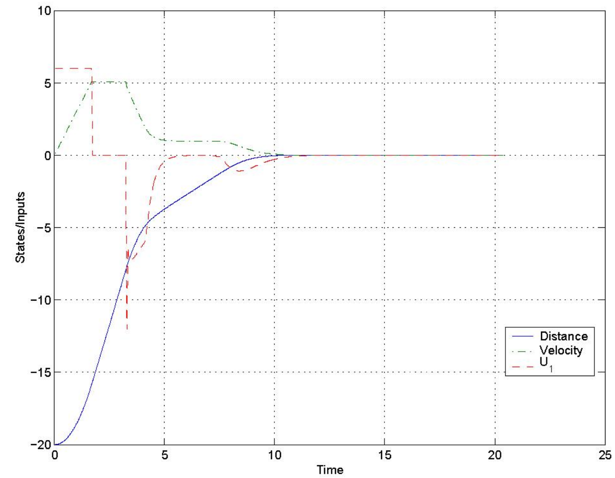

Fig. 7. System Response - Powerful break, Close condition, Velocity saturation.

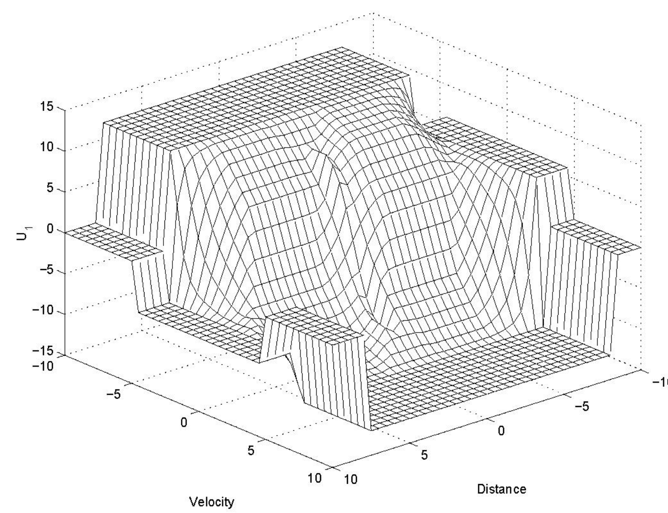

Fig. 8. Position Rules Surface - Powerful break, Close condition, Velocity saturation

B. Orientation Regulation Rules

In III-A we assumed that the helicopter is on a line passing through origin. This serves for the first part of our total strategy

"if the direction of the helicopter is pointing at the origin move to the origin, otherwise turn towards the origin."

In order to change the orientation of the helicopter so that it points towards the origin, we should regulate the yaw angle

θ so that the angle between θ and the line from the origin to the helicopter be some multiplier of π. Furthermore, we not only want to align it but also stop the change in the yaw angle when the helicopter is point exactly towards the origin. This is the same case that we had for regulating the position around origin with the same membership functions. Here θ and distance have the same role, so as r and velocity. The response of the system to rotate to the origin with no position change is shown in Figure 9.

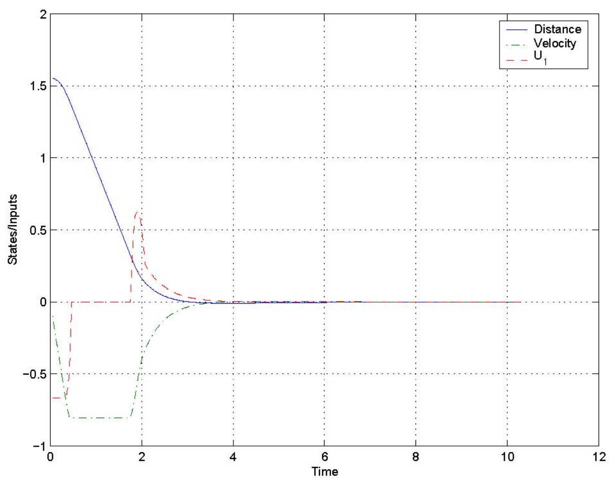

Fig. 9. Orientational Motion

C. Position Orientation Relating Rules

The relation between the position and the orientation and going to the origin is somehow geometric. The helicopter should point to the origin before starting to go. So a membership function called *Front* is defined based on the following geometric angles.

θ	The angle of the helicopter with the X-Axis.
ϕ	The angle between the line that connects the center of the helicopter to the origin with the X-Axis.
e_θ	$\theta - \tan^{-1}(\tan(\theta - \phi))$

The goal of the coupling can be simply said to be

> *"Accelerate to the origin if the origin is in front of the helicopter, otherwise break."*

This rule can be then improved by defining the *Far* condition and say that

> *"If the origin is in front of the helicopter or the helicopter is far away from the origin, accelerate to the origin, otherwise break."*

The *Front* condition is described in Figure 12.

IV. Conclusion

In this paper we proposed a fuzzy controller to stabilize a helicopter moving on a plane around origin given non-zero initial conditions. A subjective control design technique for the controller was used and the fuzzy reasoning is of Mamdani type. We show that the problem can be solved using a number of simple rules. In order to improve performance, other rules was added to the rule-base fuzzy controller where the improvement in the response of the system was shown using computer simulation. The future works would be to apply the proposed method to an experimental setup, which brings together other issues such as sensor fusion in the vision system. The next step would be to deal with the full model moving in 3 dimensions.

References

[1] R. T. C. J. T. D. M. Stipanovic, G. Inalhan, "Decentralized overlapping control of a formation of unmanned aerial vehicles," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 20, Las Vegas, December 2002, pp. 569–571.

[2] P. D. D. B. Kadmiry, P. Bergsten, "Autonomous helicopter using fuzzy-gain scheduling," in *Proc. of the IEEE Int. Conf. on Robotic & Automation(ICRA)*, vol. 3, Seoul, Korea, May 2001, pp. 2980–2985.

[3] J. J. G. R. O. R. N. C. Cavalcante, J. Cardoso, "Design and tuning of a helicopter fuzzy controller," in *International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., Proceedings of 1995 IEEE International Conference on Fuzzy Systems.*

[4] E. F. V. Gavrilets, B. Mettler, "Dynamic model for a miniature aerobatic helicopter," in *Department of Aeronautics and Astronautics, MIT*, Cambridge, USA.

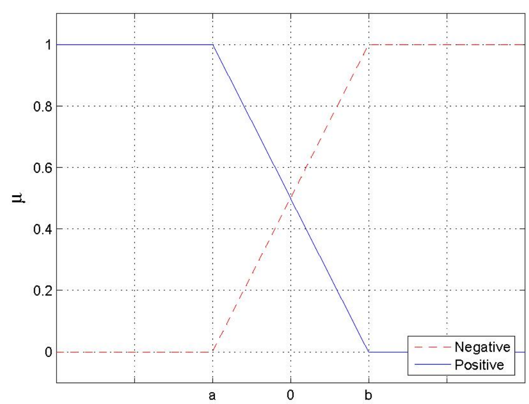

Fig. 10. Positive & Negative: Velicity, Distance, r, θ

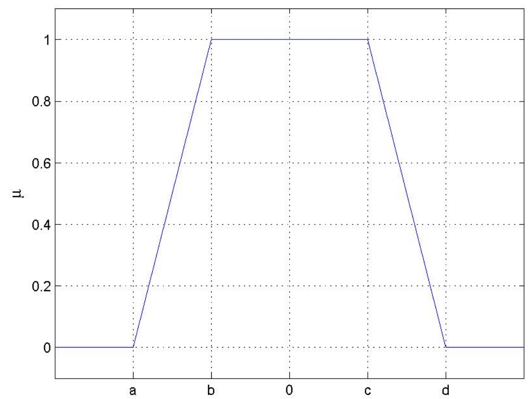

Fig. 11. Distance Close

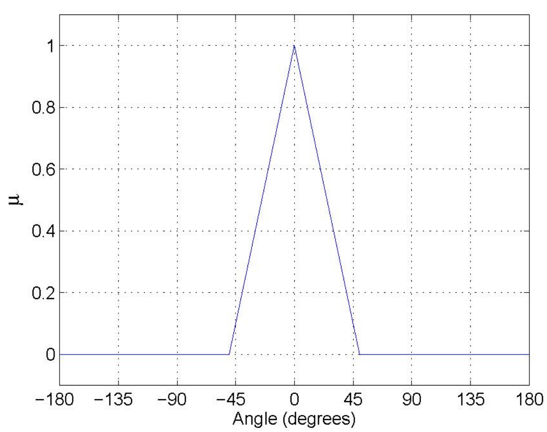

Fig. 12. Front

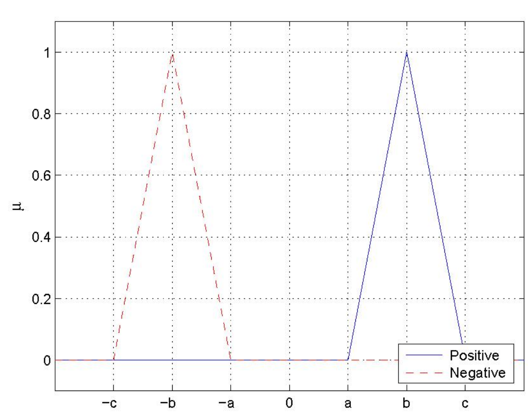

Fig. 13. u_1 Positive & Negative