FUZZY CONTROLLER FOR ANTITANK WIRE GUIDED MISSILE SIMULATOR WITH DIRECT X SDK

Valentine PENEV

1. Introduction

The conditions of modern high technology engagement demand the best from operators. Some of the warriors ordered to active duty with the guided weapons had little or no launch training, making the use of simulation system crucial in gaining guidance experience. The missile launcher to operational deployment and combat is turned into simulation training system. The operator simultaneously can train under realistic war fighting conditions in the same unit. He actually will operate to maintain or enhance proficiency, especially in anti-armored engagements. Creation of the similar simulator is complex a task, where different skill have to be used.

Many studies have been performed on the regulator synthesis of an antitank wire guided missile, taking in mind possible countermeasures of mobile targets. Flexible features of a fuzzy regulator give a possibility to design pitch and yaw loops of missile with (line-of-sight) LOS guidance. In semi-automatic systems the operator only tracks the target, his task is to keep the optical center line of his sight pointing at the required point of missile impact.

In the first part of this paper the notations of the missile models are listed. In part 2 the missile model is yielded in the respect of three-dimensional pursuit problem. The effect of fin servo saturation is discussed in part three. The design of the fuzzy controller is described in the part 4, and effects of changes of membership functions and fuzzy inference rules related to the time-to-go, miss distance and other parameters are discussed in part 5. Part six is devoted to software issues. In the final section of the paper the main features of the simulator for antitank wire guided missile are discussed.

Notations

In this paper the following notations are used:

P, Q, R	Roll, pitch and yaw rate to the axes b_x, b_y and b_z
Φ	Body-axes roll angle
Θ	Body-axes pitch angle
Ψ	Body-axes yaw angle
$(X,Y,Z)^T$	Position vector of the missile center of mass with respect to the inertial frame
$(U,V,W)^T$	Velocity vector of the missile with respect to the body frame (m/s)
α, β	Attack angle and sideslip angle (rad)
δ_q, δ_r	Elevator deflation angle and rudder deflation angle
T	Thrust
M	Missile mass (kg)
S	missile wing reference area (m^2)
V_m	Magnitude of the missile velocity
M_x, M_y, M_z	External torques to the directions b_x, b_y and b_z
F_x, F_y, F_z	External forces to the directions b_x, b_y and b_z
C_{Fx}, C_{Fy}, C_{Fz}	Total aerodynamics force coefficients to the directions b_x, b_y and b_z
C_{Mx}, C_{My}, C_{Mz}	Total moment coefficients to the directions b_x, b_y and b_z
A_x, A_y, A_z	Acceleration along the directions b_x, b_y and b_z at center of mass
G_x, G_y, G_z	Forces of gravity to the directions b_x, b_y and b_z
I	Moment of the inertial tensor of the missile
M	Mach number
D	Reference length (m)
QS	Dynamic pressure
σ_z, σ_y	Misalignment to the central line
K_{sl}, k	Seeker stabilization gain
K_1, k_2, k_3	Guidance loop gains
T_1, T_N	Seeker and noise filter time constant, respectively

2. Missile Model

A conceptual medium range wire guided missile is considered where missile dynamics having six degrees of freedom are provided. Figure 1 shows the pitch guidance loop, which is typical for "twist and steer" and has rate and acceleration feedback type autopilot.[1,2,3] The yaw guidance loop is almost the same as the pitch channel[2,3] except that the pitch and the yaw change their places.
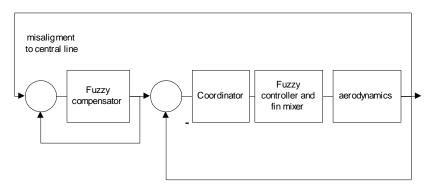

Figure 1: Missile pitch guidance loop

The 6-DOF missile dynamic equations[1,2,3,6] are written in the following state space form

$$\dot{P} = \frac{I_{yy} - I_{zz}}{I_{xx}} QR + \frac{M_x}{I_{xx}}$$

$$\dot{Q} = \frac{I_{zz} - I_{xx}}{I_{yy}} RP + \frac{M_y}{I_{yy}}$$

$$\dot{R} = \frac{I_{xx} - I_{yy}}{I_{zz}} PQ + \frac{M_z}{I_{zz}}$$

$$\dot{\Phi} = P + \left(Q\sin\Phi + R\cos\Phi\right)\tan\Theta$$

$$\dot{\Theta} = Q\cos\Phi - R\sin\Phi$$

$$\dot{\Psi} = \frac{Q\sin\Phi + R\cos\Phi}{\cos\Theta}$$

$$\dot{U} = RV - QW + \frac{1}{m}F_x$$

$$\dot{V} = -RU + PW + \frac{1}{m}F_y$$

$$\dot{W} = QU - PV + \frac{1}{m}F_z$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = R_1(\Phi)R_2(\Theta)R_3(\Psi) \begin{bmatrix} U \\ V \\ W \end{bmatrix}$$ - where R_1, R_2 and R_3 are the rotational matrices

The external forces and torques caused by the control surfaces, aerodynamics and gravity forces are defined as follows:

$$F_x = C_{Fx} \cdot QS \cdot S + T + G_x$$
$$F_y = C_{Fy} \cdot QS \cdot S + T + G_y$$
$$F_z = C_{Fz} \cdot QS \cdot S + T + G_z$$

$$M_x = C_{Mx} \cdot QS \cdot S_d$$
$$M_y = C_{My} \cdot QS \cdot S_d$$
$$M_z = C_{Mz} \cdot QS \cdot S_d$$

Part of normal values are listed as follows: m = 11.5 kg;   Vm = 186.0 m/s; Tm = 580 N; ht0 = 0.50 m.

3. Fin Mixer

This section explores regions of instability and general problems of fin mixer.[2,3] Semi-automatic missiles often utilize a set of four fins to control pitch and yaw. Missiles of this type are typically guided by some form of LOS guidance (see figures 2 and 3). As a result the guidance loop gain is proportional to the range to the missile. The resulting fin commands sent from a classically developed autopilot often saturate the fin actuators, the rate, acceleration or position. When an actuator is driven into saturation, the control loop does not perform in the manner in which the control law has been designed. This can lead to instability or slow dynamical response. The internal square represents the area, where pitch and yaw commands may be implemented to decrease misalignment to center line.
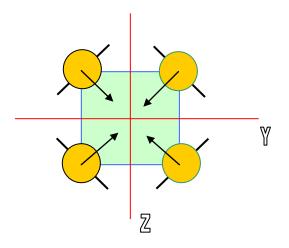
Figure 2: Rear view of representative twist and steer missile with 2 fins

This phenomenon is particularly troublesome for twist and steer missile which rely on the same fins for roll control and pitch control. During intercept the actuators are often driven into and/or acceleration limits due to large pitch commands caused by the LOS guidance and diminishing range. However, the fins must also be utilized to perform roll control to align the plane of maximum normal acceleration capability in the proper direction to pursue the sight. As a result of the saturation, roll control is lost momentarily and in some cases permanently, and the overall miss distance is increased.

This phenomenon is illustrated in figure 3, where the vector A represents the desired command in terms of combined pitch and yaw commands. The internal square represents the limits of capability of the fin actuators. For clarification purposes only position limitations are represented; however, rate and acceleration limits can be view in similar manner. Because the pitch command exceeds the actuator capability, the actuators are driven into saturation. As a result, not only is the pitch command not achieved, but the yaw command is also not accomplished as seen in vector B. For unstable missiles, this gain reduction can results in a temporary loss of stability. This instability is present as long as the fin actuators remain in saturation and often leads to large transient in roll error and increased miss distance.

It is desirable for a fin mixer to accommodate the limitations of the fins and perform the desired roll command represented by vector C. One way to minimize this effect is to reduce the guidance loop gain in the pitch axis. However, this results in performance loss for nominal intercepts.

The key to solving this problem is to develop a fin mixer which can anticipate actuator limitations and reduce the pitch gain prior to issuing fin commands, thus maintaining tight yaw control while providing maximum achievable pitch capability.
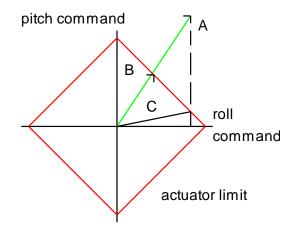

Figure 3: The effects of actuator limits

Let's introduce the simple linear fin mixer common in most missile designs. Linear fin mixers translate the pitch and yaw commands from the control law to individual fin deflection commands in a linear fashion. This may be thought of as a simple matrix multiplication of the commands through a mixer matrix:

$$\begin{bmatrix} fin1 \\ fin2 \\ fin3 \\ fin4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ -1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} pitchcommand \\ yawcommand \end{bmatrix}$$

$$u = 0.25\, M^T\, \tau_d\ (1/4MM^T = 2I)$$

This technique is effective for a range of fin commands well within the actuator limitations, and it is with this mixer in mind that the classical control loops are designed. It is appropriate to use again fuzzy rules[4] to design a nonlinear fin mixers, which have identical mixing capabilities as the linear mixer within the limitation envelope of the fin actuators.

4. Fuzzy Control

LOS system can be called "three-point guidance" because there are target, missile and tracker or sight unit. The objective of the guidance loop is to move the missile as closely as possible on the line joining the tracker and the target. The concept is very simple, but there are some fundamental features, which limit the accuracy of LOS systems. The general characteristics which limit the short range semi-automatic systems are: missile "g" requirements in the respect of target motion, the limited beam width of tracker, uncertainty of the missile range.

The velocity control loop is suitable for systems designed to hit the stationary or slow moving targets, where small adjustments only are required by the operator. The basic idea is to incorporate the fuzzy IF-THEN rules into the control loop and to investigate how expert experience will guide highly nonlinear coupled object like missile.

The five input variables are used as follow: LOS pitch rate - *pr*; LOS yaw rate - *yr*; LOS pitch - *pp*; LOS yaw -*yy*; time-to-go - *t*. The LOS pitch rate and pitch, LOS yaw rate and yaw have six fuzzy allegiances - NB, NM, NS, PS, PM, PB. The membership functions have shape that is defined in the next equation and Table 1, where *u* may be *pr*, *yr*, *yy* and *pp*.

$$\mu(u) = \begin{cases} 0, & \alpha \leq \alpha_1 \\ a_1\alpha + b_1, & \alpha_1 \leq \alpha \leq \alpha_2 \\ a_2\alpha + b_2, & \alpha_2 \leq \alpha \leq \alpha_3 \\ 0, & \alpha \geq \alpha_3 \end{cases}$$

Table 1. Maximal grades for MF (without magnifier).

	a_1	b_1	a_2	b_2	α_1	α_2	α_3
NB	-	-	$-3/\pi$	-2	-	$-\pi$	$-2\pi/3$
NM	$3/\pi$	3	$-3/\pi$	-1	$-\pi$	$-2\pi/3$	$-\pi/3$
NS	$3/\pi$	2	$-3/2\pi$	$\frac{1}{2}$	$-2\pi/3$	$-\pi/3$	$\pi/3$
PS	$3/2\pi$	$\frac{1}{2}$	$-3/\pi$	2	$-\pi/3$	$\pi/3$	$2\pi/3$
PM	$3/\pi$	-1	$-3/\pi$	3	$\pi/3$	$2\pi/3$	π
PB	$3/\pi$	-2	-	-	$2\pi/3$	π	-

The variable time-to-go has two fuzzy allegiances ZO and PS. The elevator deflation angle and rudder deflation angle are used as fuzzy output variables. For the commands δ_q and δ_r six fuzzy allegiances (NB, NM, NS, PS, PM, PB) are considered and the membership functions have the form of singleton.

The fuzzy engines for pitch and yaw loops[4,5] are shown on tables 2 through 5.

Table 2. Fuzzy rules for pitch loop - t is ZO.   **Table 3.** Fuzzy rules for yaw loop - t is ZO

pr/pp	NB	NM	NS	PS	PM	PB
PB	PM	PM	PS	NS	NM	NM
PM	PM	PM	PS	NS	NM	NM
PS	PB	PB	PS	NS	NB	NB
NS	PB	PB	PS	NS	NM	NM
NM	PM	PM	PS	NS	NM	NM
NB	PM	PM	PS	NS	NM	NM

yr/yy	NB	NM	NS	PS	PM	PB
PB	PM	PS	PS	NS	NS	NM
PM	PB	PM	PS	NS	NM	NB
PS	PB	PB	PM	NM	NB	NB
NS	PB	PB	PM	NM	NB	NB
NM	PB	PM	PS	NS	NM	NB
NB	PM	PM	PS	NM	NM	NM

Table 4. Fuzzy rules for pitch loop - t is PS.   **Table 5.** Fuzzy rules for yaw loop - t is PS.

pr/pp	NB	NM	NB	PB	PM	PB
PB	PB	PB	PB	NB	NB	NB
PM	PB	PM	PM	NM	NM	NB
PS	PM	PM	PM	NM	NM	NM
NS	PM	PM	PM	NM	NM	NM
NM	PB	PB	PM	NM	NB	NB
NB	PB	PB	PB	NB	NB	NB

yr/yy	NB	NM	NS	PS	PM	PB
PB	PB	PB	PB	NB	NB	NB
PM	PM	PM	PM	NM	NM	NM
PS	PM	PM	PM	NM	NM	NM
NS	PM	PM	PM	NM	NM	NM
NM	PM	PM	PM	NM	NM	NM
NB	PB	PB	PB	NB	NB	NB

5. Discussion of Controller

The result show that fuzzy controlled missile against high mobile target generally produces appropriate miss distance. But the increasing down distance leads to the very big miss distance. This investigation is a first step in using fuzzy control in the field of guided weapons and the investigation. The simulator consists of model of launcher unit, computer, first monitor mounted ahead of the sight, second monitor for controlling officer, keyboard and mouse.

6. Description of Direct X SDK

Direct3D® Immediate Mode application programming interface (API) is part of the three-dimensional (3-D) graphics component of DirectX®. Direct3D is designed to enable world-class game and interactive three-dimensional (3-D) graphics on a computer running Microsoft® Windows®. It provides device-dependent access to 3-D video-display hardware in a device-independent manner. Direct3D is a drawing interface for 3-D hardware.

Direct3D has two modes: Immediate Mode and Retained Mode. Retained Mode is a high-level 3-D API for programmers who require rapid development or who want the

help of the Retained Mode built-in support for hierarchies and animation. Direct3D Immediate Mode is a low-level 3-D API that is ideal for developers who need to port games and other high-performance multimedia applications to the Windows operating system. Immediate Mode is a device-independent way for applications to communicate with accelerator hardware at a low level. Direct3D Retained Mode is built on top of Immediate Mode.

These are some of the advanced features of Direct3D:

- Switchable depth buffering (using z-buffers or w-buffers)
- Flat and Gouraud shading
- Multiple light sources and types
- Full material and texture support, including mipmapping
- Robust software emulation drivers
- Transformation and clipping
- Hardware independence
- Full support on Windows 95, Windows 98, and Windows 2000
- Support for the Intel MMX architecture

Developers who use Immediate Mode instead of Retained Mode are typically experienced in high-performance programming issues and may also be experienced in 3-D graphics. The best source of information about Immediate Mode is the sample code included with this SDK; it illustrates how to put Direct3D Immediate Mode to work in real-world applications. The world management of Immediate Mode is based on vertices, polygons, and commands that control them. It allows immediate access to the transformation, lighting, and rasterization 3-D graphics pipeline. If hardware is not present to accelerate rendering, Direct3D offers robust software emulation. Developers with existing 3-D applications and developers who need to achieve maximum performance by maintaining the thinnest possible layer between their application and the hardware should use Immediate Mode, instead of Retained Mode.

Direct3D Immediate Mode provides simple and straightforward methods to set up and render a 3-D scene. The key set of rendering methods are referred to as DrawPrimitive methods; they enable applications to render one or more objects in a scene with a single method call. For more information about these methods, see DrawPrimitive Methods. Immediate Mode allows a low-overhead connection to 3-D hardware. This low-overhead connection comes at a price; the designer must provide explicit calls for transformations and lighting, all the necessary matrices, and to determine what kind of hardware is present and what its capabilities are.

7. Building the Simulator

Wire guided antitank missile is a semi-automatic system. The operator only tracks the target, his task is to keep the optical center line of his sight pointing at the required point of missile impact. His sight is fitted with appropriate cross-wires and circles to assist him in aiming. Signals proportional to the angular misalignment of the missile from the center line of sight are processed by the ground unit (analog or digital computer) and are transmitted to the missile via wires. The processing of angular misalignment will include multiplication by term proportional to the missile range. The question is how accurately can an operator track the target.

To create the sensation of real combat conditions the computer simulation display mounted ahead of the sight is used. Because the system uses actual background terrain pictures, a variety of realistic environments from the desert to winter scene can be used. There are plenty of ground features such as buildings, rivers, threes and so on. The visual display simulation of weather, with particularly good representation of scud, fog is better than in much more expensive simulators. A personal computer controls the simulation system, automatically scores engagements and collects training management data. Audio equipment realistically replicates engine and weapon. Inputs to the computer come from instructor or operator.

The simulator for antitank wire guided simulator /ATGMS/ consists of launcher, computer, software package, sight control unit, computer, keyboard, mouse, second monitor (figure 4). This version is for indoor usage. Software package consists of simulator, scenario maker and viewer. ATGMS was designed into Direct3D Immediate Mode, which gives possibility to render the scene with 40000 vertices in real-time.

In the simulator of wire guided antitank missile the following objects can be defined:

- Terrain
- Stationary objects – buildings, trees, bushes, bridges
- Slow moving objects – tanks, armored vehicles, helicopters
- Fast moving objects – the missile
- Atmospheric effects – fog

The Environment3D window displays the view from the launch unit sight (figure 5). There are some objects in this view: scene with background terrain, moving and stationary targets, other graphical objects that are part of the current configuration. In the Environment3D window, we have to see the missile response to the commands from the vertical and horizontal controls. It is very easy to connect software

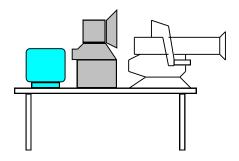application to the vertical and horizontal controls from the launching unit using the DirectX Input.
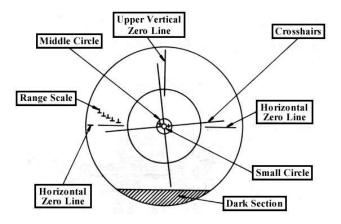

Figure 4: General view of ATGMS

Figure 5: Sighting reticle

References

1. Babister A. W., *Aircraft stability and control* (1961).
2. Bodner V.A., *Control systems for aircrafts* (Moscow: Mashinostroenie, 1973).

3.   Garnell, P., *Guided weapon control systems*, 2nd ed. (Oxford, England: Pergamon Press, 1980).

4.   Driankov D., H. Hellendoorn, and M. Reinfrank, *An introduction to fuzzy control* (SpringerVerlag, 1997).

5.   Valentine Penev, Petya Koprinkova, Georgi Georgiev, and Peter Petrov, "Fuzzy guidance controller for air to air missile," Presented during *HEMUS-96* (Plovdiv, Bulgaria, 1996).

6.   John M. Rolfe, *Flight simulation*, (Cambridge: Cambridge University Press, 1988).

VALENTINE PENEV: Born 1960. M.Sc. (1986, Automatics and Telemechanics) from the Technical University of Tula, Russia. Ph.D. (1992, Fuzzy sets in digital control systems) from the Robotics and Control Systems Research Center at the Bulgarian Academy of Sciences. Currently Dr. Penev is Deputy Director for R&D of the Institute of Control and Systems Research at the Bulgarian Academy of Sciences. Senior Research Fellow with over forty publications in the field of intelligent technology engineering - fuzzy control, fuzzy reasoning, signal processing, and neuro-like structures. Software realization of a flight simulator and the simulator "Fagot" for ATGMs under Windows95, Windows 3.1x and DOS for PC. Address for correspondence: Institute of Control and Systems Research, Bulgarian Academy of Sciences, Acad.G.Bonchev Str., Bl.2, Sofia 1113, Bulgaria. E-mail: valpenev@iusi.bas.bg