Autopilot Design Based on Fuzzy Logic and Genetic Algorithm

Xionghui Lu, Jianqiang Yi, Member, IEEE, and Dongbin Zhao, Member, IEEE
Institute of Automation, Chinese Academy of Sciences
Xionghui.lu@ia.ac.cn, {jianqiang.yi, dongbin.zhao}@mail.ia.ac.cn

Abstract

Compared with human beings, autopilots have advantages in controlling ships in higher precision. However, due to disturbances caused by sea wind, wave, flow and some other uncertainties such as variance of ship physical parameters, how to control autopilots properly is a very difficult problem. An autopilot based on fuzzy logic and genetic algorithm is designed to make a ship follow "virtual course angle", which is calculated from real course angle and track deflection. Simulation results show that the ship follows the desired track satisfactorily.

1. Introduction

The first autopilot was implemented by Sperry using proportional control in 1922 [1]. Those early PID (proportional plus integral plus derivative) autopilots are widely known with their theoretical simplicity, reliability and easy-to-construct.

In 1970s [2], adaptive autopilots were designed and their control parameters are adjusted automatically in accordance with conditions and therefore are able to serve well under different circumstances. However, they are of great complexity in computation.

Since 1980s, intelligent autopilots began to attract wide attention. Three intelligent theories, Expert System[3], Fuzzy Logic Control[4][5] and Neural Network[6], are successfully used in many applications and result in more sophisticated and reliable control systems.

In this paper, an intelligent controller involving several intelligent methods is presented. Firstly, instead of controlling both course angle and trajectory, "virtual course angle" is proposed to be the only variable that the controller concerns. Secondly, a fuzzy logic controller is designed. After that, scaling parameters of the controller are adjusted with genetic algorithm. At last, simulation results verify the effectiveness of the method.

The paper is organized as follows. Firstly, in section Ⅱ, ship system is described and its mathematical model is introduced. Secondly, in section Ⅲ, the fuzzy controller is designed, and its fuzzy rules are carefully established. Besides, in this section, particularly, we will propose a new concept of "virtual course angle", which helps to simplify construction of the control system. In section Ⅳ, scaling gains of the fuzzy controller is tuned with genetic algorithm. Simulation results will be shown in section Ⅴ. Finally, conclusion is given in section Ⅵ.

2. System description

2.1. Mathematical model

A two-order model is introduced in this paper[7]:

$$\ddot{\psi}(t) + Kd(\dot{\psi}(t)) = K[\delta_r(t) + disturb(t)] \tag{1}$$

where ψ is course angle of the ship, δ_r is real rudder angle, K is ship hydrodynamic parameter depending on some ship geometric parameters such as ship speed, $disturb(t)$ denotes disturbances, and $d(\dot{\psi})$ is damping term of the equation:

$$d(\dot{\psi}) = d_3\dot{\psi}^3 + d_2\dot{\psi}^2 + d_1\dot{\psi} + d_0 \tag{2}$$

Since most ships are symmetry in shape, we have $d_2 = d_0 = 0$.

Besides, since the real rudder angle does not follow the desired rudder angle exactly, changing rate of the real rudder angle is limited in about 3° per second in general and the real rudder angle does not exceed about 35° in either direction for most ships. Therefore, we have following equation:

$$\dot{\delta}_r(t) = 3\text{sat}\left[\frac{35\text{sat}\left(\frac{\delta_c(t)}{35}\right) - \delta_r(t)}{3}\right] \tag{3}$$

Where δ_c is the desired rudder angle.

Proceedings of the First International Conference on Innovative Computing, Information and Control (ICICIC'06)
0-7695-2616-0/06 $20.00 © 2006 IEEE

2.2. Virtual course angle

In early years, the destination of autopilots was only to keep ships following desired course angle. Nowadays, many autopilots are capable of keeping both ship course angle and trajectory meantime. However, these autopilots are usually very complex in control construction and control algorithm.

In order to simplify controllers design and improve their performance, "virtual course angle" is proposed. The "virtual course angle" is calculated from the course angle error and the trajectory error. The virtual course angle can be expressed as:

$$\psi_v = \psi_c + f(d_r) \tag{4}$$

Where ψ_c is the desired course angle, ψ_v is the virtual course angle, d_r is the trajectory error, and $f(d_r)$ is a function of d_r. As a result, if a ship can keep virtual course angle successfully, it can also keep the course angle and the trajectory in the same time.

Definition of the function f should meet following requirements: when d_r is very big, the ship approaches the desired trajectory as soon as possible regardless of course angle error; on the other hand, when the ship sails near the desired trajectory, the ship should switch to mainly maintain course angle. Keeping all these in mind, f is defined as follows. The threshold of the saturation function can be assigned different values to meet different requirements:

$$f(d_r) = \begin{cases} d_r, & if\ d_r \le 2 \\ 0, & if\ d_r > 2 \end{cases} \tag{5}$$

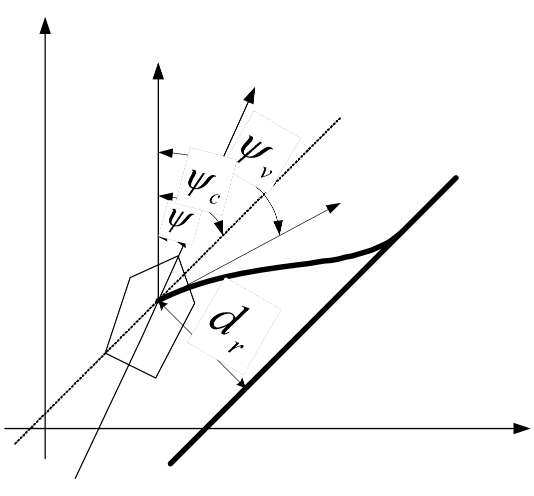

Figure 1: d_r, ψ_c and ψ_v in ship steering

The three variables, ψ_c, ψ_v and d_r, are depicted in Figure 1.

In the end, complete construction of the whole system is displayed in Figure 2.(in the last page)

3. Fuzzy controller

First of all, after and ahead of the fuzzy controller, we have three scaling gains: g_0, g_1 and h, which are displayed in figure 3. Genetic algorithm will be introduced to tune these scaling gains later on.

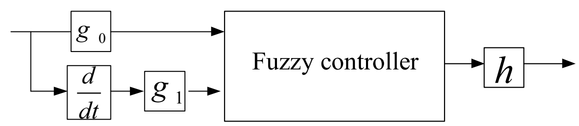

Figure 3: The scaling gains

The fuzzy controller has two inputs and one output. One input is the error between the real course angle and the virtual course angle, and the other is changing rate of the error:

$$e(kT) = \psi_v(kT) - \psi(kT)$$
$$c(kT) = \frac{\psi_v(kT) - \psi(kT)}{T} \tag{6}$$

Where ψ_v is the virtual course angle, ψ is the real course angle, and T is the sampling period.

The rudder angle δ_c is the output.

Membership functions are shown in Figure 4.Where E^i ($i = -1, 0, 1$) corresponds to the virtual course angle error, C^j ($j = -1, 0, 1$) corresponds to changing rate of the virtual course angle error, and U^k ($k = -1, 0, 1$) corresponds to the desired rudder angle.

Fuzzy rules are described in following form:

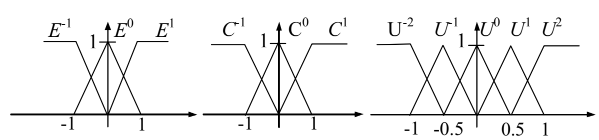

Figure 4: Membership functions

If E is E^i, and C is C^j, then U is U^k

Based on human knowledge, we have 9 rules that are listed in table 1.

0-7695-2616-0/06 $20.00 © 2006 IEEE

Table 1:Fuzzy rules

U		E^{-1}	E^{0}	E^{1}
C	C^{-1}	U^{-2}	U^{-1}	U^{0}
	C^{0}	U^{-1}	U^{0}	U^{1}
	C^{1}	U^{0}	U^{1}	U^{2}

4. Tuning of scaling gains with genetic algorithm

In this section, the three scaling gains, g_0, g_1 and h, are tuned with genetic algorithm(GA).

Firstly, 50 random strings composed of "1" and "0" are generated, and each of these strings has 27 bits. Then every string is split into three small strings(genes), which correspond to the three scaling gains respectively. The real scaling gain values, which can be calculated through mapping function, distribute randomly in desired range.

Under each group of the scaling gains, real track trajectory is generated and then is evaluated with fitness function, which is sum of absolute values of trajectory errors in every sampling instant:

$$J = N - \int(|\, y_r(t) - y_c(t)\,|)dt \qquad (7)$$

Where N is a positive value, which guarantees that the trajectory that is the closest to desired trajectory has the biggest fitness value and all fitness values are positive. $y_r(t)$ and $y_c(t)$ are vertical coordinates of real track trajectory and desired trajectory in time t respectively. Integrating range is the total simulation time. Genes of the next generation are chosen according to their fitness values. The ones that own bigger fitness values are more likely to be selected and replicated to the next generation. The method of Roulette is used to find out these "good" genes.

When it comes to the operation of crossover, one-point crossover is applied. During every crossover cycle, after we select two "good" genes(parent genes), a crossover position is set randomly, and each gene is split into two parts by the crossover position, then the four fragments are exchanged and form two new genes(child genes). Again and again, a new population of genes is generated. Crossover happens in crossover rate, which should be set carefully since it dramatically affects algorithm performance.

Another indispensable operation in GA is mutation, which guarantees gene diversity in every generation. In the case of bit representation, the genes will be flipped in a mutation operation, that is, a "1" is changed to a "0" or a "0" is changed to a "1". Mutation happens in mutation rates, which is usually very small. Too big mutation rates will result in algorithm divergence.

5. Simulation results

In the mathematical model, $k = 0.0107, d_1 = 9.42, d_3 = 2.24$ [8]. The ship speed V_e is supposed to be 16 knots (one unit distance in our plot corresponds to one sea mile in reality). Besides, the disturbances are introduced ahead of the block of real rudder angle and are represented with white noise whose amplitude is 1.5°.

In the fuzzy controller, the search range is defined as follows. Suppose $g_0 = \frac{1}{20}$ =0.05, $g_1 = \frac{1}{6}$ =0.1667, h=100. Upper and lower boundary of the search range is threefold and third of these values respectively.

Crossover and mutation rate of the GA selected in this paper is 0.3 and 0.001 respectively. The N in (12) is 160.

Simulation shows that the algorithm tends to be stable after generation 15 and a relatively optimal set of scaling gains are obtained, that is, g_0 =0.0276, g_1=0.1086 and h=77.6908.

When initial position of the ship is [-3, 1], and the initial course angle is 0°, the ship tracking and desired trajectory are displayed in Figure 5. Apparently, the ship follows desired trajectory satisfactorily.

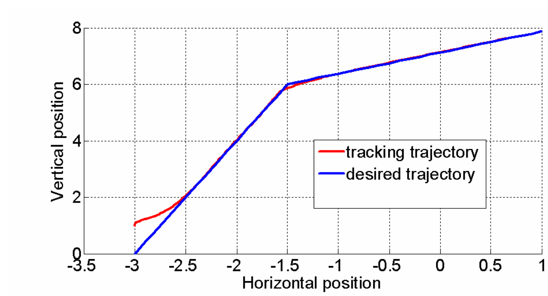

Figure 5: The tracking and desired trajectory

With other initial states, similar results are obtained with the same fuzzy controller.

6. Conclusion

In this paper, we proposed an autopilot consisting of a fuzzy logic controller whose fuzzy rules are established based on human knowledge and scaling gains are tuned with genetic algorithm. Besides, instead of keeping course angle and trajectory simultaneously, our ship needs only to follow the "virtual course angle". As a result, construction of the whole system is simplified a lot. Simulation results show the effectiveness of the method.

7. Acknowledgement

This work was supported in part by NSFC Projects (No. 60334020, 60440420130, 60575047) and the Outstanding Overseas Chinese Scholars Fund of Chinese Academy of Sciences (No. 2005-1-11), China

8. References

[1] E. Sperry. Directional Stability of Automatically Steered Bodies. Journal of the American Society of Naval Engineers. vol. 42, no.1, pp.2-13, 1922.

[2] Amerongen, Van J.; "Adaptive steering of ships - A model reference approach", Automatica, vol. 20, No. 1, 1984, 3-14.

[3] W. Brown, E. Pizzariello. Factors to Consider in Developing a Knowledge-based Autopilot Expert System for Ship Maneuvering Simulation. Proceedings of the 14th Ship Technology and Research Symposium. S5, pp.1-11, USA, 1989.

[4] J. V. Amerongen, H. R. van Nauta Lemke, and J.C.T. van der Veen. An Autopilot for Ships Designed with Fuzzy Sets. IFAC on Digital Computer Applications to Process Control. USA, 1997

[5] Jianqiang Yi, Naoyoshi Yubazaki, and Kaoru Hirota: Trajectory Tracking Control of Unconstrained Object by the SIRMs Dynamically Connected Fuzzy Inference Model, Journal of Advanced Computational Intelligence, Vol.4, No.4, pp.302-312 (2000).

[6] N.A. J.Witt, Miller KM. A Neural Network Autopilot for Ship Control. Proceedings of Maritime Communications and Control Conference. Marine Management (HoldingsLtd. London, UK, pp.13-19, 1993

[7] Minh-Duc Le; Thanh-Ha Nguyen; Thi-Thai Nguyen; Song-Phi Nguyen; Thi-Vi Le; Trung-Dong Ngyuen; Thi-Huong Hoang; Tien-Dung Hoang. Computational Intelligence in Robotics and Automation, 2003. Proceedings. 2003 IEEE International Symposium on Volume 3, 16-20 July 2003 Page(s):1411 - 1415 vol.3

[8] D.R.Yoerger and J.E Slotine. Nonlinear Trajectory Control of Autonomous Underwater Vehicles using Sliding Methodology[J]. Ocean ， 1984 ， 16 ： 588-593.

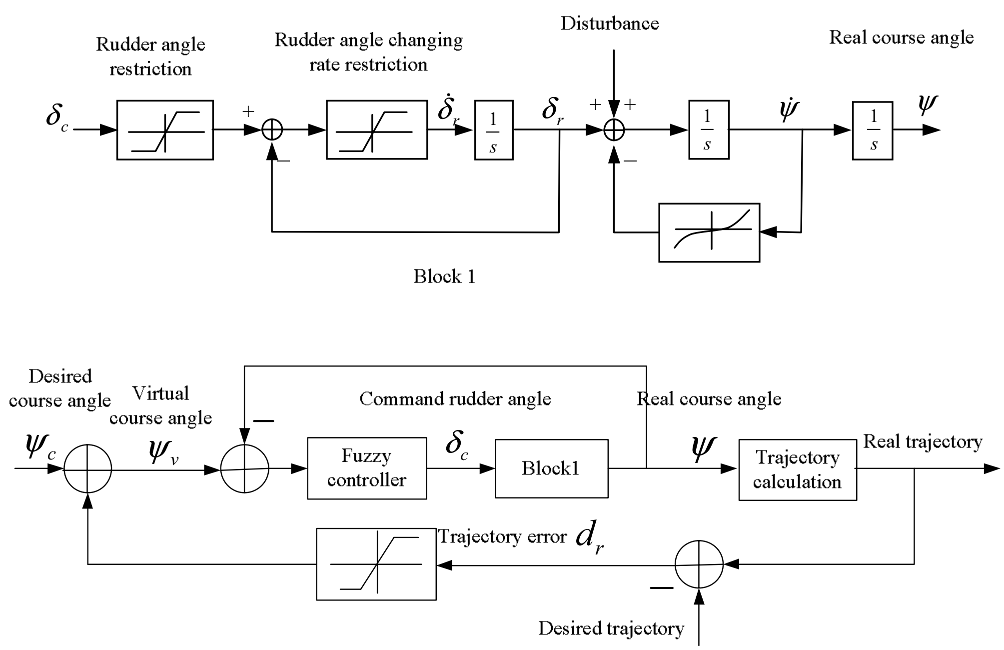

Figure 2: Construction of the ship system