

Genetic Fuzzy System in Weapon Control Module of Unmanned Tanks

Xinzhu Liu, Di Ding, Zihuan Yu, Zhongshuai Wang, Yuhan Dai and Jing Huang*

School of Computer Science and Technology, Jilin University, Changchun, China

Key Laboratory of Symbolic Computation and Knowledge Engineer (Jilin University), Ministry of Education, China

*Corresponding author

Abstract—The genetic fuzzy system is applied to weapon control module of unmanned tanks. In this paper, the simulation system is built to train the fuzzy rule base of genetic fuzzy system in order to get the optimal fuzzy rule base with training tasks. Testing tasks are used to test the success rate of different missions under the optimal rule base. The fuzzy inference system obtained by the algorithm achieves high success rate in various tasks.

Keywords—unmanned tanks; genetic fuzzy system; fuzzy rule base; weapon control module

I. INTRODUCTION

A. Problem Description

In this paper, we solve the problem of weapon selection in unmanned tanks combats. In the missions of unmanned tanks, decisions on how to use weapons need to be made by tanks. We have built an unmanned tanks simulation system to simulate the missions. The genetic fuzzy system is used as the weapon control module of unmanned tanks. The fuzzy rule base of the system is trained with no prior knowledge by different types of training tasks and the optimal fuzzy rule base obtained is tested with testing missions.

B. Related Work

Decisions of the unmanned system can be made by genetic algorithm, reinforcement learning algorithm and other intelligent methods, which are applied in unmanned aircrafts and unmanned tanks.

Nicholas Ernest et al. propose Genetic Cascading Fuzzy System [1, 2] and Genetic Fuzzy Tree [3, 4, 5, 6, 7], which divide the intelligent control system into independent sub-modules and are respectively optimized by genetic algorithm. However, the fitness function of the genetic algorithm they use is kind of inappropriate. The values of coefficients in that function are arbitrary and variables considered in that function are not normalized. Zhixiong Xu [8] et al. apply the improved reinforcement learning algorithm to unmanned tank battle decisions, introducing prior knowledge for learning, and further contrasting the convergence speed of different learning algorithms. However, they do not focus on the success rate of their algorithm.

Based on the research of Nicholas Ernest, we use the genetic fuzzy system in the weapon control module of unmanned tanks. The algorithm does not use any prior knowledge to complete the training process, focusing on the

success rate of different tasks with the optimized system. Our algorithm solves problems of other research mentioned above.

C. Main Contributions

The main contributions of this paper are as follows:

First, an unmanned tanks simulation system is built to simulate different missions. Second, the genetic fuzzy system is applied to the weapon control module of unmanned tanks, and the structure of the genetic fuzzy system is designed. Third, we have designed different kinds of training missions to train the fuzzy rule base to obtain the optimal solution. Besides, we have designed testing missions to test the optimal fuzzy rule base.

II. THE UNMANNED TANKS SIMULATION SYSTEM

The unmanned tanks simulation system is used to simulate the missions of unmanned tanks. The simulation system includes our unmanned tanks and enemy units. Enemy units consist of patrol tanks, infantry units with anti-tank missiles and targets. Our tanks have defense missiles and offensive weapons. Offensive weapons contain attack missiles and machine guns. Attack missiles can attack arbitrary enemy units, and machine guns can only attack infantry units. Defense missiles can intercept enemy attack missiles.

In the missions, the goal of our tanks is to destroy all enemy targets and avoid being destroyed by other enemy units. The following assumptions are made for the simulation system to simplify the simulation process. The route of our tanks is defined previously and enemy units cannot move. If our tanks enter the attack range of enemy tanks or anti-tank missiles, corresponding enemy units will launch two missiles to attack our tanks immediately. Targets cannot attack our tanks. If our tanks detect the enemy missiles, our tanks will choose appropriate defensive strategies to defend themselves. Defense missiles have the success rate limit. After the defense process, our tanks can attack enemy units in the attack range. In each attack round, our tanks can only attack one enemy unit. If the number of enemy units in the attack range of our tanks is more than one, our tanks will choose the attack object according to the following priority: target, patrol tanks and infantry units. Enemy units have no defense system, and our offensive weapons have the success rate limit. All weapons in the simulation system have the maximum capacity.

III. WEAPON CONTROL MODULE BASED ON GENETIC FUZZY SYSTEM

A. Basic Concepts

Genetic fuzzy system [9] combines genetic algorithm with fuzzy inference system. In this system, the genetic algorithm is used to optimize the rule base and membership function of fuzzy inference system, instead of using expert knowledge to build a knowledge base. The rule base and membership function of fuzzy inference system are encoded as the chromosome of genetic algorithm. The optimal knowledge base of the inference system will be obtained.

B. Weapon Control Module

In the simulation system, the fuzzy inference system is used as the weapon control module, which is divided into two fuzzy inference sub-modules: the offensive weapon selection sub-module and the defensive strategy selection sub-module.

Offensive weapon selection sub-module is used to select the attack weapons of our tanks. If our tank is going to attack an enemy infantry unit, our tank needs to choose attack missiles or machine guns as the offensive weapon. The inputs of the subsystem are the number of remaining attack missiles of our tank, remaining machine guns and remaining enemy combat units including patrol tanks and infantry units. The output is the use of attack missiles or machine guns. The fuzzy inference subsystem adopts triangular membership function, and the input variables correspond to three fuzzy subsets -- "small quantity", "medium quantity" and "large quantity".

Defensive strategy selection sub-module is used to select the defensive strategy. If our tank is attacked, our tank can choose an adventurous, moderate or conservative defensive strategy which uses two, three or four missiles respectively to intercept the two enemy missiles launched by one enemy unit. The inputs are the number of remaining defense missiles and remaining enemy combat units. The output is the adventurous, moderate or conservative defensive strategy. The membership function is the same as the offensive weapon selection subsystem.

C. Fuzzy Rule Base Code

The outputs of the two subsystems are encoded as shown in TABLE I and TABLE II.

We encode the fuzzy rules of the two subsystems. Offensive weapon selection module has 3 input variables, and each variable corresponds to 3 fuzzy subsets, so the fuzzy rule base has 27 if-then rules. Defensive strategy selection module has 2 input variables, and each variable corresponds to 3 fuzzy subsets, so the fuzzy rule base has 9 if-then rules. The length of the string is 36 bits, and each bit representatives an if-then rule. The string is considered as the chromosome of genetic algorithm. The optimization process of the two subsystems is independent.

TABLE I. OFFENSIVE WEAPON SELECTION SUBSYSTEM OUTPUT CODE

Output(offensive weapon)	The output code
Attack missiles	0
Machine guns	1

TABLE II. DEFENSIVE STRATEGY SELECTION SUBSYSTEM OUTPUT CODE

Output(defensive strategy)	The output code
Adventurous strategy	0
Moderate strategy	1
Conservative strategy	2

D. Fitness Function

In this problem, the fitness function is set as follows.

$$f = a + p_1 \cdot eg + p_2 \cdot ek + p_3 \cdot ef - p_4 \cdot am - p_5 \cdot dm - p_6 \cdot en \quad (1)$$

We normalize the variables in the function in order that the values of all variables are in the range of [0, 1]. The variables are normalized as follows.

$$eg = \frac{\text{The number of destroyed enemy targets}}{\text{The total number of enemy targets}} \quad (2)$$

$$ek = \frac{\text{The number of destroyed enemy tanks}}{\text{The total number of enemy tanks}} \quad (3)$$

$$ef = \frac{\text{The number of destroyed infantry units}}{\text{The total number of infantry units}} \quad (4)$$

$$am = \frac{\text{The number of used attack missiles}}{\text{Maximum capacity of our attack missiles}} \quad (5)$$

$$dm = \frac{\text{The number of used defense missiles}}{\text{Maximum capacity of our defense missiles}} \quad (6)$$

$$en = \frac{\text{The number of used machine guns}}{\text{Maximum capacity of our machine guns}} \quad (7)$$

p_1, p_2, p_3, p_4, p_5 and p_6 are all non-negative values, which are the weight coefficients of each variable in the function. The weight of each variable determines the direction of the optimization process of fuzzy rule base. The goal of our tanks is to destroy all enemy targets without being destroyed by enemy units. Therefore, p_1 is the largest coefficient. If our tank is destroyed by enemy units during the mission, the value of the fitness function will be 0. a is the offset parameter to ensure that the function has a non-negative value. We assume that $a = p_4 + p_5 + p_6 + 1$ to ensure that the value is smallest when our tank is destroyed in the missions.

IV. IMPLEMENT OF THE ALGORITHM

In the algorithm, we use Roulette method in natural selection process and set the probability of crossover and mutation. The chromosome in the algorithm is the encoded rule base. The process of the algorithm is shown in FIGURE I.

The algorithm performs as follows:

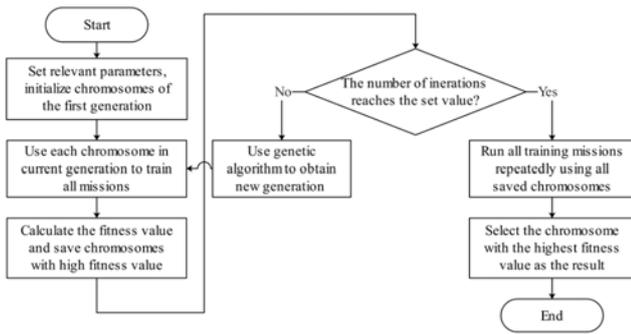


FIGURE I. THE PROCESS OF THE ALGORITHM.

Step 1: Set the relevant parameters of genetic algorithm and initialize the chromosomes of the first generation randomly.

Step 2: Our tanks use every fuzzy rule base in the current generation to complete all training missions.

Step 3: For every fuzzy rule base, the simulation system calculates the fitness value of every mission and takes the sum of all those fitness values in every mission as the fitness value of the corresponding fuzzy rule base.

Step 4: Select and save several chromosomes, which represent corresponding fuzzy rule base, of current generation with high fitness value.

Step 5: If the number of iterations does not reach the set value, do selection, crossover and mutation process to get chromosomes of the new generation and go to Step 2. Otherwise, go to Step 6.

Step 6: All saved chromosomes repeat all missions for many times. The simulation system calculates the average fitness value and success rate of every chromosome. The chromosome with the highest average value and success rate will be selected as the best fuzzy rule base.

In the algorithm, the specific process of using fuzzy rule base in the simulation system to complete missions shows as follows:

Step 1: Set the route of our tanks previously.

Step 2: In every round of the mission, all enemy units determine whether our tank is in its attack range. Each unit, whose attack range covers our tank, fires two attack missiles at our tank.

Step 3: Our tank determines whether it is attacked by enemy units. If no enemy units attack our tank, go to Step 4. Otherwise, our tank uses defensive strategy selection subsystem to choose the defensive strategy and change the status after the defense process.

Step 4: Our tank determines whether there are enemy units in its attack range. If there is only one unit in the range, that unit is the attack object of our tank. Otherwise, our tank chooses one unit as the attack object according to the set priority order mentioned above. If the attack object is the enemy infantry unit, our tank will use offensive weapon selection subsystem to choose the attack weapon. The

simulation system updates the status of our tank and all enemy units after that attack round.

Step 5: If our tank arrives at the finishing point, the simulation mission is completed. Otherwise, go to Step 2 and repeat the process until our tank reaches the finishing point.

V. RESULTS AND ANALYSIS

A. Missions in the Simulation System

The system contains 5 training missions and 3 testing missions. Training missions are used to obtain the optimal fuzzy rule base in weapon control module. Testing missions are applied to test its success rate.

There are five training missions. In FIGURE II, Mission 1 is used to train general missions with short route. In FIGURE III, Mission 2 is similar to Mission 1. The number of infantry units is 2 less than that in Mission 1. Mission 2 is used to train easier missions. As shown in FIGURE IV, in Mission 3, the number of enemy units is small. The enemy units are distributed separately, and there is no patrol tank. In FIGURE V, Mission 4 has the same number of enemy units as that in Mission 1, but the route is longer. This mission is used to train missions with long path. In FIGURE VI, Mission 5 resembles Mission 4, with 6 more infantry units and one more patrol tank. It is used to train missions in which the route is long and the number of enemy units is large.

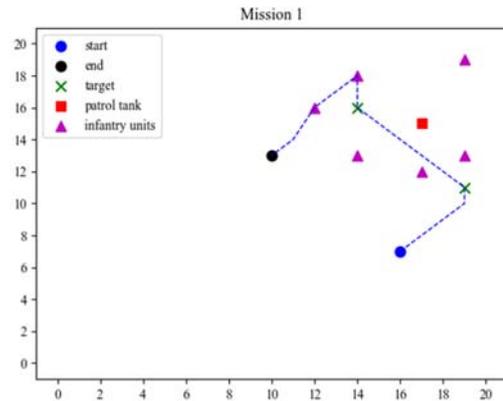


FIGURE II. THE MAP OF MISSION 1.

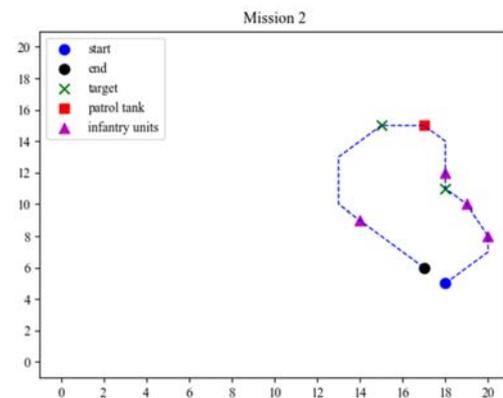


FIGURE III. THE MAP OF MISSION 2.

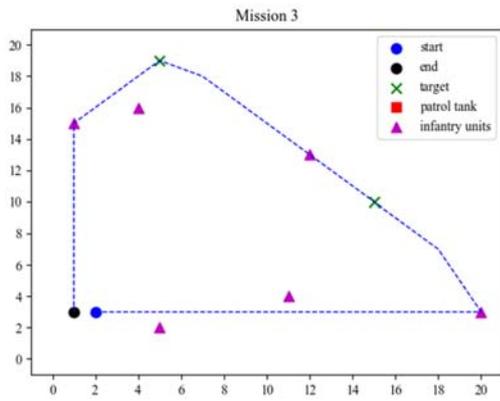


FIGURE IV. THE MAP OF MISSION 3.

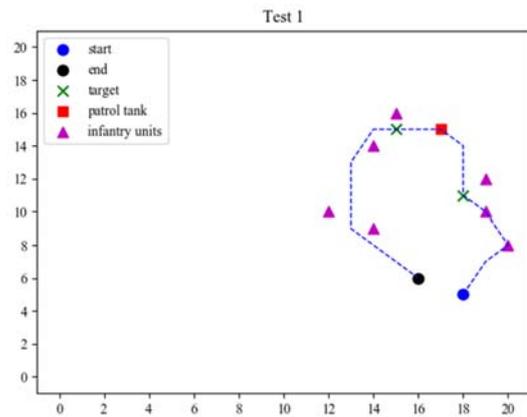


FIGURE VII. THE MAP OF TEST 1.

There are 3 testing missions. In FIGURE VII, Test 1 is a bit similar to Mission 1 and Mission 2, but it has more enemy units. It is used to test the situation where the path is short and the enemy units are more intensive. In FIGURE VIII, Test 2 resembles Mission 3. The number of enemy infantry units is ten more than that in Mission 3, and there are more targets at the same time. In FIGURE IX, Test 3 resembles Mission 4 and Mission 5. It is used to test the situation in which the route is long and the number of enemy units is large.

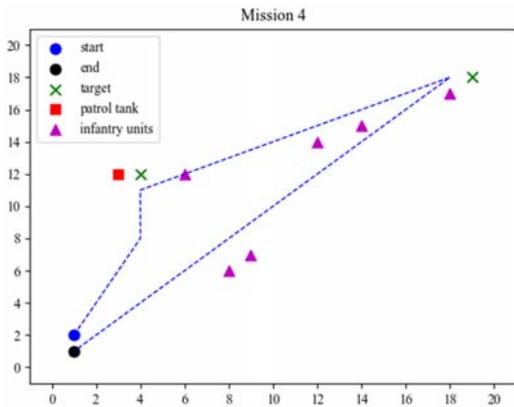


FIGURE V. THE MAP OF MISSION 4.

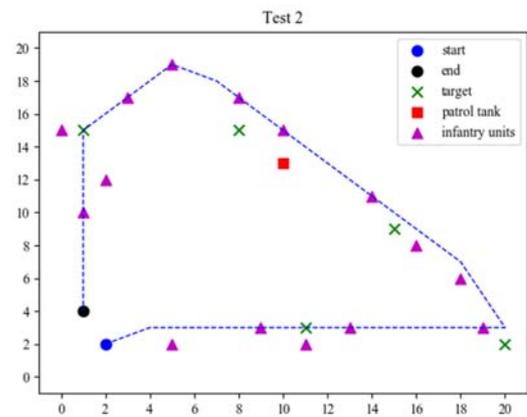


FIGURE VIII. THE MAP OF TEST 2.

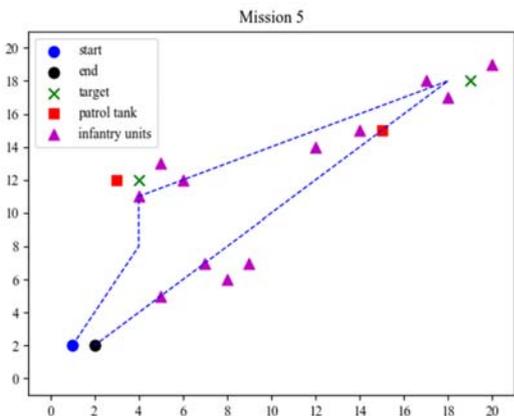


FIGURE VI. THE MAP OF MISSION 5.

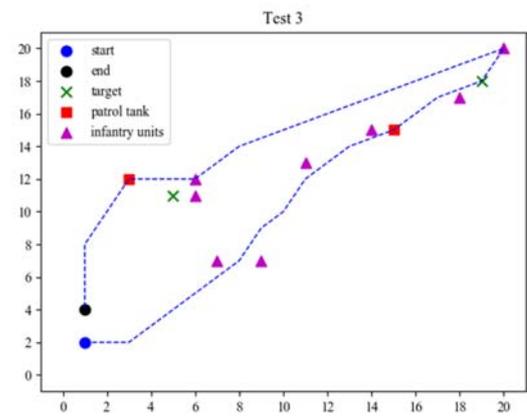


FIGURE IX. THE MAP OF TEST 3.

B. Results and Analysis of Training and Testing Missions

In this system, the optimal rule base of the weapon control module is obtained using some or all of the five training missions mentioned above. Three testing missions are used to test the success rate.

The results are shown in TABLE III and TABLE IV. TABLE III shows the success rate of using the optimal rule base to test all training tasks after training. The first column in TABLE III represents the training missions used during the

training process. The first line from M1 to M5 represents Mission 1 to Mission 5. Table IV shows the success rate of using the optimal rule base in testing tasks after the training process is completed. The first column in TABLE III has the same meaning as TABLE IV.

TABLE III. SUCCESS RATE OF TRAINING MISSIONS

Training missions	M1	M2	M3	M4	M5
1,2,3,4,5	100%	100%	100%	100%	85%
2,3,4	100%	100%	100%	100%	66%
1,2	100%	100%	100%	100%	76%

TABLE IV. SUCCESS RATE OF TESTING MISSIONS

Training missions	Test 1	Test 2	Test 3
1,2,3,4,5	100%	76%	85%
2,3,4	99%	99%	100%
1,2	100%	89%	94%

From the results, it can be noticed that after training with all training missions, we can get high success rate in all training missions. Since the number of enemy combat units is larger than other training missions, the success rate of Mission 5 is a little lower than other training missions. Besides, using part of the training missions can also achieve good training results and can get high success rate in three testing missions. The fuzzy rule base obtained by training with only Mission 1 and Mission 2 even gets better results in three testing missions compared to rule base trained by all training missions. Through the results, it can be seen that the genetic fuzzy system has a good effect on the application of unmanned tanks weapon control module.

This algorithm can be further optimized. The number and type of missions set in the simulation system are limited. We can further increase the variety and quantity of training and testing missions to train and test the weapon control module more comprehensively.

ACKNOWLEDGMENTS

This work was supported in part by National Natural Science Foundation of China under grants 61373053 and 61572226, and Jilin Province Key Scientific and Technological Research and Development project under grants 20180201044GX and 20180201067GX.

REFERENCES

[1] N. Ernest, K. Cohen, C. Schumacher and D. Casbeer, "Learning of Intelligent Controllers for Autonomous Unmanned Combat Aerial Vehicles by Genetic Cascading Fuzzy Methods," SAE Technical Paper 2014-01-2174, Sep. 2014, doi:10.4271/2014-01-2174.

[2] N. Ernest, E. Garcia, D. Casbeer, K. Cohen and C. Schumacher, "Multi-agent Cooperative Decision Making using Genetic Cascading Fuzzy Systems," AIAA SciTech at Aerospace, Jan. 2015, doi:10.2514/6.2015-0888.

[3] N. Ernest, "Genetic Fuzzy Trees for Intelligent Control of Unmanned Combat Aerial Vehicles," PhD. Dissertation. Ohio: Univ. of Cincinnati, 2015.

[4] N. Ernest, K. Cohen, E. Kivelevitch, C. Schumacher and D. Casbeer, "Genetic Fuzzy Trees and their Application Towards Autonomous Training and Control of a Squadron of Unmanned Combat Aerial Vehicles," Unmanned Systems, vol. 3, May 2015, pp. 185-204, doi: 10.1142/S2301385015500120.

[5] N. Ernest, D. Carroll, C. Schumacher, M. Clark, K. Cohen and G. Lee, "Genetic Fuzzy based Artificial Intelligence for Unmanned Combat

Aerial Vehicle Control in Simulated Air Combat Missions," Journal of Defense Management, vol. 6, 2016, doi:10.4172/2167-0374.1000144.

[6] N. Ernest and K. Cohen, "Fuzzy Logic Based Intelligent Agents for Unmanned Combat Aerial Vehicle Control," Journal of Defense Management, vol. 6, 2015, doi: 10.4172/2167-0374.1000139.

[7] A. Sathyan, N. Ernest and K. Cohen, "An Efficient Genetic Fuzzy Approach to UAV Swarm Routing," Unmanned Systems, vol. 4, April 2016, pp. 117-127, doi: 10.1142/S2301385016500011.

[8] Z X Xu, L Cao and X L Chen, "Research on Multi-motivation Reinforcement Learning Algorithm and Simulation of Unmanned Tank Battle," Computer Engineering and Applications, in press, <http://kns.cnki.net/kcms/detail/11.2127.TP.20170622.1849.002.html>.

[9] F. Herrera and L. Magdalena, "Genetic Fuzzy Systems: A Tutorial," Tatra Mountains Mathematical Publications, vol. 13, June 1997, pp. 93-121.