Modeling and Simulation of CGF Aerial Targets for Simulation Training

Siqing Wang, Yunzhu Liu

ABSTRACT

Aiming at the requirement for the establishment of intelligent and autonomous targets for air defense weapon simulation training, this paper uses modular and componentized model structure to construct the framework and granular analysis of CGF air target modeling. In this paper, a target maneuvering model, a radar fire control model and an intelligent expert decision-making model are established, and there are two target construction methods: autonomous and controlled. The simulation experiment is conducted with a certain type of air defense weapon target simulator as the background. The results show that this model solves the problem of single target parameter in the CGF simulation training system, training complexity and lack of realism, and meets the requirements of simulation training and strengthens training. The immersion of the personnel has effectively improved the training level. And the component-based development model gives the model good reusability and interactivity.

KEYWORDS

Simulation Training, Computer-generated Forces, Componentization, Modeling and Simulation.

INTRODUCTION

As the air combat environment becomes more and more complex, warfighters face the difficulties of huge installation costs, difficulties in the organization and scheduling of personnel and equipment, and high security pressure when conducting tactical training such as air confrontation, multi-aircraft coordination, and air defense operations. In this case, simulation training is an important means for combat commanders to conduct large-scale simulations such as tactical exercises and equipment training, which can achieve the purpose of reducing equipment loss, enhancing training safety, and improving combat capabilities. This paper uses computer programs to control the generation and behavior of CGF air targets, so that the number of targets increases and their behavior has an autonomous intelligence level. It can respond, judge and make decisions based on the real-time situation of the battlefield, so as to provide high-quality combat trainers with opponents or friendly forces. At present, most scholars in this research field focus on the architecture design of the CGF system, and the construction of its
entity model is too simple to meet the requirements of combat training. Therefore, this article analyzes and studies the construction of the CGF system model. According to the needs of combat training, the target maneuvering model, radar fire control model and intelligent decision-making model are established, and a certain type of air defense weapon target simulator is used as the research object, and the computer force generation technology is used to generate the target and have autonomous behaviors to achieve offensive and defensive confrontation. Effective actual combat tactical training. This paper realizes the visual simulation of the system interface and combat situation through the mechanism of ACoreOS Tianmai operating system and system simulation platform.

CGF AIR TARGET ANALYSIS

Analysis of Model Granularity

In the process of modeling and simulation, the choice of model granularity is related to the degree of description of the details. Simulation entities described with different granularities have different functions and influences in the model. Therefore, choosing the right granularity is very important to the application scope and functional purpose of the established model. CGF air targets are composed of several functional components, including maneuver model, radar fire control model, decision model, tactical rule library and action library. The relationship between them is relatively independent, and the calling methods between these components are carried out. Therefore, the CGF air target belongs to the platform-level combat model, which is a collection of all functional components. In addition, the CGF air target constructed in the simulation training system has the ability to perceive the battlefield situation, respond to changes in the external environment, and make intelligent decisions and actions, and can accept the centralized and unified guidance management of the entire CGF system.

System Composition and Model Framework

Based on model granularity analysis, the CGF system model is generally composed of an entity maneuver model, an entity decision model and an action execution model. The entity maneuvering model performs physical understanding and calculation of the moving entity, and obtains its state parameters such as attitude and position through the dynamic equation; the entity decision model analyzes the real-time battlefield situation and makes the optimal decision according to the tactical rule base; the action execution model draws conclusions through the reasoning mechanism and further solves the corresponding parameters of the entity.

In order to make the CGF system more complete and efficient, this article adds a command parsing module and embodies each module of the CGF system, as shown in Figure 1. Including command analysis module, target maneuver model, radar fire control model, intelligent decision model, tactical rule library and action library. The command analysis module mainly parses the command into a form that can be recognized and processed by the computer; the target maneuver model simulates the real flight status of an aircraft or missile, and outputs its important
status parameters; the radar fire control model realizes basic simulation entity offensive and defensive countermeasures through the simulation of basic functions such as radar search, interception and tracking, missile aiming, and launch guidance; the intelligent decision-making model analyzes the battlefield situation and the situation of the enemy and the enemy, transmits information to the mobile module, and makes a series of intelligent decision-making actions based on the tactical rule base and the action base. In addition, CGF air targets can be constructed in two ways: autonomous and controlled. Controlled means that the target parameters are artificially modified through the external interface, and the command information is transmitted to the analysis module through the network. The system will solve the instructions according to the instructions to complete the tactical action and feed the results back to the battlefield environment; autonomy means the simulator autonomously generates intelligent targets which have basic offensive and defensive capabilities and intelligent decision-making capabilities. They make decisions based on the entity information provided by the radar fire control model, and guide the targets to attack, defend, and evade tactical actions.

CONSTRUCTION OF CGF AIR TARGET MODEL

In order to realize the effect of actual combat, the constructed air target should accurately reflect its real flight characteristics and tactical indicators, and can accurately simulate the performance of the actual combat flying target. After the simulation training begins, the combatants set target parameters according to the training needs, and autonomously call air targets with different parameters according to the battlefield situation to achieve tactical actions such as take-off, attack, defense, and flight, and complete tactical behavior such as formation maintenance, radar detection, and missile guidance.

Target Maneuver Model

The current literature generally uses six-degree-of-freedom dynamic parameter simulation for the construction of entity maneuvering models. Its basic idea is to regard the entity as a rigid body, and all the motion of the entity is decomposed into the motion of the center of mass, the motion around the center of mass, or the
combination of the two motions. The operation process is to input the control amount of the control stick or the rudder surface for parameter calculation, calculate the real-time state of the entity according to the dynamic equation, speed and acceleration equation, and then output the aircraft action. The six-degree-of-freedom dynamic parameter simulation method can accurately describe the motion state of the entity, but the calculation is complicated and the calculation is large. The controller design requires repeated tests to adjust the parameters, and it is difficult to realize real-time dynamic simulation in the application. In addition, combat operators usually determine the maneuver of the aircraft based on tactical objectives, and then solve various parameters based on dynamic equations. However, the use of the six-degree-of-freedom maneuvering model cannot closely link the amount of control and aircraft maneuvering, and cannot reflect good dynamic characteristics and tactical indicators. Therefore, when the accuracy requirements are not particularly high, consider the use of a combat mission-oriented aircraft CGF maneuver model that can accurately describe the aircraft maneuver and closely contact the combat target. The model inputs are parameters such as the magnitude and direction of the maneuvering overload, engine state or speed, etc., which transform the combat operator's combat intentions into the input of aircraft maneuver control.

There are two advantages to adopting this method. First of all, it has a certain accuracy. The parameters calculated according to the aircraft maneuvering model are more in line with the actual dynamic characteristics and can ensure the availability of tactical inferences. It should be noted that the relevant limitations of aircraft maneuverability should be fully considered, and feasible maneuvering overloads should be used to determine aircraft parameters. Secondly, it has certain practicality. Combat operators pay attention to the maneuvering of the aircraft, and decompose the expected motion of the aircraft into speed, overload, direction, etc. according to the combat purpose, and use them as the model input interface. The maneuvering model established in this way can keep the expected maneuvering actions of the aircraft consistent with various parameters, and finally achieve combat simulation based on certain tactical purposes. Based on the above analysis, the aircraft maneuver model established in this paper can not only meet the combat purpose in the simulation, but also accurately describe the dynamic characteristics of the aircraft, and convert the combat operator's combat intention into the maneuver control of the aircraft.

The aircraft CGF maneuver model for combat missions is composed of four parts: aircraft dynamics equations, engine power models, aerodynamic and gravity calculation models, and aircraft motion basic parameter solution models, as shown in Figure 2. The engine power model uses a steady-state calculation method to obtain the thrust response process according to the engine's throttle change, thereby obtaining the thrust time history P. The aerodynamic force calculation uses a steady-state calculation model. The gravity calculation adopts the constant value model. The operator control expression is the description of the operator's input to the aircraft maneuver, and the control of the aircraft maneuver is described as maneuver overload, overload direction, engine state and other parameters, which is more in line with the maneuver expectations of combat. Choosing these parameters as input is also more in line with the operator's control of aircraft maneuvering. For example, when completing the horizontal circling action, the operation process is to
deviate the normal overload from the vertical plane and maintain a certain roll angle to make the overload reach \( n_{s} = \frac{1}{\cos \gamma} \).

In order to meet different combat requirements and maneuvering requirements, we divide the model into two parts, given engine thrust state or given flight speed. When the engine thrust state is given, the engine power model calculates the thrust and provides constraint conditions for the solution speed. When the flight speed is given, the engine power model calculates the power limit conditions to constrain the flight speed. Therefore, the input of the maneuver model is \( (n_s, \gamma, v_w) \) or \( (n_s, \gamma, \phi) \). In this formula, \( \phi \) is the throttle position and \( v_w \) is the vacuum speed.

The centroid dynamic equation of the machine is:

\[
\begin{align*}
\dot{V}_h &= \frac{1}{m} X_h \\
\dot{\theta} &= \frac{1}{mv_w} Y_h \\
\dot{\phi} &= -\frac{1}{mv_w \cos \theta} Z_h \\
\end{align*}
\]

(1)

In order to simplify the model, it is assumed that the operator always makes the sideslip angle small, let \( \beta = 0 \), \( Z = 0 \), which simplifies the calculation of the input volume and aerodynamic side force. Decompose the input into the resultant force, and the resultant force expression is:

\[
C_m \begin{pmatrix} p \\ 0 \end{pmatrix} + C_{nq} \begin{pmatrix} -Q \\ Y \end{pmatrix} + C_{nd} \begin{pmatrix} 0 \\ -mg \end{pmatrix} = \begin{pmatrix} X_h \\ Y_h \\ Z_h \end{pmatrix}
\]

(2)
In the formula, $P$ is the thrust value of the engine power model. When the maneuvering input is speed, $P$ satisfies the balance of the resultant force in the tangent direction of the track, $Q$ is the resistance, $Y$ is the lift, which is calculated from the overload and aerodynamic derivatives, and $Z$ is the side force. $C_{ht}$, $C_{ia}$, and $C_{hd}$ are the coordinate conversion matrix of the aircraft body axis system, the speed axis system, and the ground axis system to the aircraft track axis system. It can be calculated by parameters such as $\theta$, $\gamma$, $\varphi$, $\alpha$, and $\beta$.

According to the kinematic relationship, the space position of the aircraft is:

$$\begin{bmatrix}
  v_{dx} \\
  v_{dy} \\
  v_{dz}
\end{bmatrix} = \begin{bmatrix}
  \cos \theta \cos \varphi_j \\
  \sin \theta \\
  \cos \theta \cos \varphi_j
\end{bmatrix} \begin{bmatrix}
  v_h \\
  0 \\
  0
\end{bmatrix}$$

(3)

The aircraft attitude parameters are determined by the following formula:

$$\sin \theta = \sin \theta \cos \alpha + \cos \theta \sin \alpha \cos \gamma_j + \Delta \vartheta_w$$

(4)

$$C_{td} = C_{ia} \cdot C_{hd}$$

(5)

In the formula, $\Delta \vartheta_w$ is the wind speed influence correction amount, $C_{td}$ and $C_{ia}$ are the ground axis system, the coordinate conversion matrix from the track axis system to the aircraft body axis system, and $C_{hd}$ is the coordinate conversion matrix from the ground axis system to the track axis system.

**Radar Fire Control Model**

The flight equipment is generally equipped with fire control computers, weapon systems, and radar infrared detection equipment. Combatants control the equipment switch buttons in the cockpit to achieve missile launch, guidance interception, radar detection, identification and tracking operations. The process is relatively complicated. However, in simulation training, the focus of simulation lies in the reflection of weapon equipment functions, and the requirements for high-level restoration of other details are not high. Therefore, the operation process in flight equipment can be simplified and refined, and only the most critical operating steps are retained. The relatively important combat links are shown in Figure 3.
When establishing the CGF air target radar model, it is assumed that the radar automatically enters the air combat mode. When the radar receives the track information of the target entity, it judges whether the target can be detected according to the radar distance equation, the radar interference equation, and the target reflection cross-sectional area RCS. The target echo power intensity is:

$$P_s = \frac{P_G G_r \lambda^2 \sigma}{(4\pi)^3 R^4}$$

(6)

The average power intensity of the thermal noise of the radar receiver is:

$$N = KT_0 B_f F$$

(7)

In the case of active suppressive interference on the radar, the interference power intensity of the radar receiver is:

$$P_{ij} = \frac{P_G G_r (\theta) \lambda^2 r_j}{(4\pi R_j)^2}$$

(8)

The signal-to-noise ratio in free space can be obtained by formula (6) and formula (7):

$$SNR = \frac{P_s}{N} = \frac{P_G G_r \lambda^2 \sigma}{(4\pi)^3 R^4 KT_0 B_f F}$$

(9)
In a complex electromagnetic environment, the signal-to-interference ratio at the output of the radar receiver can be obtained by formula (6) and formula (8):

\[
SIR = \frac{P_j}{P_{nj} / D_j}
\]

In the formula: \(D_j\) is the comprehensive improvement factor for comprehensive anti-active interference. On the basis of obtaining the signal-to-noise ratio or signal-to-interference ratio, the discovery probability \(P_d\) and the false alarm probability \(P_f\) can specifically quantify the radar detection performance. The discovery probability \(P_d\) can be simplified and calculated as:

\[
P_d = \left( P_f \right)^{\frac{1}{\text{ratio}}}
\]

\(u\) is to take a random variable in the interval of \([0,1]\) and make a uniform distribution, and when \(u \leq P_d\), it is judged that the radar has found the target. At this time, the CGF air entity will fly to the target with the largest threat index until the radar intercepts the target; when \(u > P_d\), it is judged that the radar has not found the target.

At the same time, assuming that the missiles are in good condition during air combat, the radar determines that the target is intercepted and the system immediately performs related fire control calculations. If the target meets the following launch conditions:

\[
\begin{cases}
P_i \leq P_{\max} \\
H_{\min} \leq H_i \leq H_{\max} \\
V_i \leq V_{\max}
\end{cases}
\]

Then the i-th batch of targets may enter the kill zone and be intercepted. \(P_i\) is the route shortcut of the target, \(H_i\) is the target altitude, \(V_i\) is the target speed; \(P_{\max}\) is the maximum route shortcut of the fire unit to ensure that the target can be killed with the expected probability, \(H_{\min}\) is the minimum altitude, \(H_{\max}\) is the maximum altitude, and \(V_{\max}\) is the maximum speed. The three-point guidance method (also weighing method) can be used to fit the missile speed. The three-point guidance method refers to a guidance method that always controls the guidance station, missile and target in a straight line, as shown in Figure 4.
The guiding equation of the three-point method is:

\[
\begin{align*}
\varepsilon_d &= \varepsilon_m \\
\beta_d &= \beta_m
\end{align*}
\]  

(13)

\(\varepsilon_d\) is the high and low angle of the line of sight of the missile; \(\varepsilon_m\) is the high and low angle of the line of sight of the target; \(\beta_d\) is the azimuth of the line of sight of the missile; \(\beta_m\) is the azimuth of the line of sight of the target.

**Intelligent Decision Model**

Simulation training is aimed at a constantly changing real-time battlefield situation. The key to improving the training effect is to make the CGF entity the same "intelligence" as the combatant, and finally realize coordinated combat or tactical confrontation. The intelligent decision-making model of CGF air target is mainly realized through expert system, which is composed of knowledge base, rule base, action base and decision module.

In the actual implementation process, we convert the combat personnel's operational principles, tactical methods, strategic thinking and other related knowledge into a form that can be recognized and processed by the computer, and stored in the knowledge base. The decision-making part adopts a rule-based behavior mechanism, and uses the result information of previous situational awareness (such as target entity location information, radar detection and tracking results, missile attack results, etc.) as the entrance, and the decision information is generated in the knowledge base through the IF-THEN structure rules for matching, which can determine the parameters in the maneuvering model. The output action strategy is divided into the following three types: target control, target device operation, and response to external commands. Therefore, improving the level of CGF's intelligent decision-making can be achieved by supplementing professional tactical knowledge, improving the combat knowledge base and related reasoning mechanisms, building a comprehensive and specific action library, and accurately outputting the entity's maneuvering actions and tactical decision-making methods.
ACOREOS OPERATING SYSTEM

Tianmai 1 embedded real-time operating system (ACoreOS) and development environment is a strong real-time embedded system platform software for multi-task applications. Its software architecture uses a three-layer stack structure: module support layer (MSL), operating system layer (OSL), application layer (AL), as shown in Figure 5. This three-layer stack structure gives the operating system better portability, and also solves the problems of hardware equipment upgrades and application software reusability. The module support layer is mainly responsible for providing services for the operating system to access hardware resources; the operating system layer is responsible for managing computer resources and providing strong support for the operation of application software; the application layer is mainly responsible for the control and management of software functions.

![Figure 5. Three-layer stack structure of ACoreOS.](image)

ACoreOS has a wide range of applications in defense equipment, rail transit, industrial control and other fields with a series of advantages. Moreover, after it is used in weapons and equipment, it can achieve the purpose of giving full play to the performance of weapons and equipment, effectively saving time and costs, and increasing the speed of application software development. As we all know, the use of foreign operating systems in the military field has great security risks and is likely to cause the leakage of military secrets. Using our own country-developed ACoreOS operating system can effectively avoid such problems and develop software safely and effectively.

SIMULATION EXPERIMENT AND RESULTS

In the research process of this subject, the hardware simulation platform is an ARM test box with ACoreOS operating system installed in it. The image file can be transferred to the test box (target machine) by loading and curing. The software platform is the development environment LambdaAE matched with the ACoreOS airborne embedded real-time operating system, used for the development and maintenance of the operating system and airborne application software. LambdaAE development environment has a series of practical and efficient features. It is an open embedded software integrated development environment, which provides strong support for the development of embedded software.
Design of Simulation Target Generation Module

Embedded target simulators can use three methods to generate simulated targets: combat scenario, manual single target mode and manual batch target mode.

Combat scenarios are hypothetical action plans for the purpose of implementing military operations and training exercises, pre-planning, evaluation, and research according to different training contents and objects. In equipment simulation training, combat scenarios are generally embodied in some form of script or configuration file, which contains elements such as battlefield environment description, force deployment, weapon equipment, action plans, and propulsion mechanisms. According to the combat scenario file, most of the information generated by the target can be obtained, and the motion trajectory and behavior parameters of the corresponding target can be generated.

The manual single target method is a method of manually generating a target temporarily according to the set parameters in the application. After the parameters are set, click the Generate Target button to extract the parameters from each parameter control and compose the target instruction information report to the training equipment according to the communication protocol.

In the manual batch target mode, the parameters of the benchmark target must first be set, and the setting method is the same as the manual single target mode. When generating targets in batches, multiple targets are generated according to the set angle distribution, distance distribution and its parameters.

The Design of the Battlefield Situation Module

For embedded target simulators, limited processing capabilities and high real-time requirements determine that it is not advisable to use a three-dimensional view to display the battlefield situation. Therefore, the target simulator studied in this paper adopts the traditional two-dimensional graphic display method in the form of a distance ring, and this method is particularly effective for air defense operations.

The drawing of battlefield situation graphics mainly includes coordinate axis, distance ring, scale, target symbol graphics, text and so on. The drawing method of situational graphics is similar to GDI drawing under Windows. The difference is that graphics programming under LambdaAE is based on emWin graphics library, which provides great convenience for LCD programming.

The resulting battlefield situation effect is shown in Figure 6:
Design of Equipment Status Display Module

In order to understand the status of each equipment of the weapon system in order to effectively control the training process in real time, the target simulator can analyze the working status and combat status report of the training equipment feedback, and display relevant information in real-time in the form of text, charts, etc.

For air defense missile weapon systems, the equipment status includes the training equipment status (whether the vehicle or the command instrument display and control console is turned on), the launch and control equipment status (normal or malfunctioning), whether the missile is in the state (with or without bombs), and the missile is good. Therefore, the status (good or faulty), the missile self-test status (no self-test or self-test), missile operational status (ready to launch, missile launched, launch failure, bounce, application for self-destruction, self-destruction order has been issued, mid-terminal guidance has been handed over).

The graphical display of the equipment status is relatively intuitive, so the nine legends shown in Figure 7 are used in the design of this article to display the missile equipment status of all firepower units. From left to right, the figure shows no bombs, bombs, ready to launch, missiles launched, launch failures, bounces, application for self-destruction, self-destruction instructions have been issued, and mid-terminal guidance has been handed over.

The embedded simulator software based on ACoreOS is different from the visual programming platform (such as Visual C++) under the traditional desktop operating system in the module design. The emWin graphics library neither supports reading bitmap data from files, nor does it support importing bitmap resources. A feasible way is to use a bitmap converter (such as Bitmap Converter for emWin) to convert the bitmap into a C file format, and then import the C file
into LambdaAE, and then use the emWin graphics library to display it on the target simulator interface.

**Design of Target Parameter Display Module**

In order to monitor the motion and state parameters of the generated target, this article uses the usual table format to display all target parameters in the design of the target simulator software. Different from the direct drag-and-drop method of generating controls in conventional visual programming, all controls in emWin programming need to be generated by code. After the list tool is created, the target parameter line is not included. After the target is generated, new parameter lines are added in sequence according to the target generation order, and then each parameter value is converted into text form and displayed in the corresponding position in the list.

**CONCLUSION**

According to the actual needs of the simulation training system, this paper studies the modeling of CGF air targets by analyzing the basic structure and model granularity of the CGF system. A target maneuvering model, a radar fire control model and an intelligent expert decision-making model are constructed, and there are two ways to build targets, autonomous and controlled. This modeling and simulation method has been applied to a certain type of simulation simulator system. The results show that the built model has good reliability and authenticity, and the constructed target has a certain degree of autonomy and intelligence, which can meet the needs of simulation training and improve the training effect and level. And the model frame design is carried out in a component-based development model, which can make the model have good reusability and interactivity.

**ACKNOWLEDGEMENTS**

Corresponding author: Siqing Wang

**REFERENCES**

1. Toubman A, Poppinga G, Roessingh J J, et al. Modeling CGF Behavior with Machine Learning Techniques: Requirements and Future Directions[C]// I/itsec. 2015.
2. Teng T H, Tan A H, Ong W S, et al. Adaptive CGF for pilots training in air combat simulation[M]. 2012.
3. Liu Xingtang, Wan Shaosong, Zhang Shuangxuan. On the Development Trend of Military Simulator/System[J]. Journal of Air Force Engineering University Natural Science Edition, 2001, 2(4):647-649.
4. Anke. Research on the Key Technology of Team CGF Behavior Modeling in Battlefield Environment[D]. Beijing Institute of Technology, 2011.
5. Li Xiangdong, Xu Ning, Hao Yukai, et al. Development and configuration of embedded software based on domestic Tianmai 1 operating system[J]. Information & Communication, 2016(12):156-158.
7. Chen Lu, Zhang Renyou, Han Zhijun. Research on Tank Computer Generated Force Model[J]. Ordnance Industry Automation, 2017, 36(1):93-96.
8. Luo Xiao, Li Weizhong. Analysis of the function and development trend of the ground-to-air missile calculation-generated force system[J]. Flying Missile, 2014(6):54-58.
9. Song G, Mu X, Zhang H. Research on Behavior Modeling Method of Agent-Based CGF[J]. Journal of Software, 2012, 7(4).
10. Tan X, Lai S, Wang W, et al. Framework of wargame CGF system based on multi-agent[C]//IEEE International Conference on Systems, Man, and Cybernetics. IEEE, 2012:3141-3146.
11. Lai Miao, Gong Guanghong, Fang Cheng, et al. Research on the Key Factors of Aircraft CGF Used in Tactical Simulation Training[J]. Command Control and Simulation, 2011, 33(5):76-79.
12. Liu Peng, Wei Baohua, Han Jia. Modeling and Simulation of Air Attack Target Based on CGF[J]. Fire Power and Command Control, 2012, 37(3):200-203.
13. Yang Xin, Wang Xiaohu, Shen Gongzhang, et al. Six-degree-of-freedom model and simulation study of aircraft[J]. Journal of System Simulation, 2000, 12(3):210-213.
14. Qi X. Design of integrated testing system for a certain fire control radar[J]. Foreign Electronic Measurement Technology, 2012.
15. Coales J F, Calpine H C, Watson D S. Naval fire-control radar[J]. Journal of the Institution of Electrical Engineers - Part IIIA: Radiolocation, 2010, 93(2):349-379.
16. Hao Lishan, Xia Hongbo, Tian Shuchao, et al. Rule-based CGF entity behavior modeling technology[J]. Firepower and Command Control, 2015(1):96-99.
17. Hao Lishan, Wang Zhongchun, Tian Shuchao, et al. Research and Implementation of CGF Intelligent Decision Model for Ammunition Support[J]. Journal of Ordnance Engineering College, 2014(3):6-10.