Modeling and Algorithm Application of Weapon Assignment System

Wang Yuhui(王玉惠)[1,2*], *Chen Shaodong*(陈哨东)[2], *Han Zhanpeng*(韩占朋)[3], *Wang Wenjing*(王文敬)[2], *Zhang Hongbo*(张洪波)[2]

1. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, P. R. China;
2. Science and Technology on Electro-Optic Control Laboratory, Luoyang, 471009, P. R. China;
3. No. 650 Institute, AVIC Jiangxi Hongdu Aviation Group Co. Ltd., Nanchang, 330024, P. R. China

(Received 14 October 2013; revised 19 December 2013; accepted 6 January 2014)

Abstract: In order to improve weapon assignment (WA) accuracy in real scenario, an artificial neural network (ANN) model is built to calculate real-time weapon kill probabilities. Considering the WA characteristic, each input representing one assessment index should be normalized properly. Therefore, the modified WA model is oriented from constant value to dynamic computation. Then an improved invasive weed optimization algorithm is applied to solve the WA problem. During search process, local search is used to improve the initial population, and seed reproduction is redefined to guarantee the mutation from multipoint to single point. In addition, the idea of vaccination and immune selection in biology is added into optimization process. Finally, simulation results verify the model's rationality and effectiveness of the proposed algorithm.

Key words: intelligent control; weapon assignment (WA); modeling; artificial neural network (ANN); invasive weed optimization (IWO)

CLC number: E833;TP391 **Document code**: A **Article ID**: 1005-1120(2014)06-0693-08

1 Introduction

Weapon assignment (WA) is a key component of decision support system (DSS). It is usually assumed that all weapons are allocated in a single stage. Then it can be considered as a static problem, which includes target-based and asset-based WA problems[1-2]. In both cases, the optimization all depends on predefined kill probability (constant value). However, kill probability is a mean value representing the weapon lethality on a specific target in weapon killing zone. Different killing zone has different kill probabilities. Therefore, theoretically approaching to actual lethality at the encounter point can achieve higher control accuracy. For this, an artificial neural network (ANN) model is trained to build the mapping relationship between theoretical and real kill probabilities.

In addition, WA is known as NP-complete problem. The computation time uiu increase rapidly with the number of threats and weapors[3]. Therefore, it is difficult to solve optimally in a time-pressed scenario. The vigorous development of intelligent algorithm provides a powerful tool to solve it. Many methods such as genetic algorithms, improved particle swarm algorithm, tabu search and particle swarm optimization algorithm, and ant colony optimization, are applied to solve the WA problem[4-7]. However, raost of notable results cost too much time in search process. It is a fatal defect for the WA optimization. Therefore, a novel numerical stochastic optimization algorithm called invasive weed optimization (IWO) is introduced. Local search is used

Foundation items: Supported by the National Natural Science Foundation of China (11102080, 61374212); the Science and Technology on Electro-Optic Control Laboratory and Aeronautical Science Foundation of China (20135152047).

* **Corresponding author**: Wang Yuhui, Associate Professor, E-mail: wangyh@nuaa.edu.cn.

to improve initial population and accelerate the whole search process. Then the redefined seeds reproduction mechanism with variable standard deviation is proposed, which can ensure the process with robust adaptive characteristic and achieve a global optimal solution quickly[8].

2 Weapon Assignment

2.1 Description

It is assumed that m group anti-aircraft weapons protect e defended assets (DAs). And v_q is protection value, which represents the DA′s important degree, where $q=1,2,\cdots,e$. In certain time, there are n targets detected by remote radar. Target value w_j is the target′s important degree, where $j=1,2,\cdots,n$.

Firstly, the kill matrix $\boldsymbol{P}=[p_{ij}]_{m\times n}$ is given, where p_{ij} is the kill probability for the ith weapon on the jth target. Then the fire decision matrix $\boldsymbol{X}=[x_{ij}]_{m\times n}$ is defined, where x_{ij} is a binary variable denoting whether the ith weapon is assigned to attack the jth target.

Based on the above, the static target-based WA problem can be formalized as a non-linear optimization problem as follows

$$\begin{cases} F=\min\sum_{j=1}^{n}w_j\prod_{i=1}^{m}(1-p_{ij})^{x_{ij}} \\ \sum_{j=1}^{n}x_{ij}=1, x_{ij}\in\{0,1\}, i=1,\cdots,m \end{cases} \tag{1}$$

where F is the total targets escape probability.

Similarly, the static asset-based WA problem can be defined as follows[2]

$$\begin{cases} D_{\mathrm{e}}=\max\sum_{q=1}^{e}v_q\prod_{j\in G_e}\left[1-\pi_j\prod_{i=1}^{m}(1-p_{ij})^{x_{ij}}\right] \\ \sum_{j=1}^{n}x_{ij}=1, x_{ij}\in\{0,1\}, i=1,\cdots,m \end{cases} \tag{2}$$

where D_{e} is the total defensive efficiency, π_j the lethality value of the jth target, and G_{e} the target set attacked the eth DA.

When there is more than one weapon available, kill probability p_{ij} is the only criterion to choose which weapon is to attack targets. However, p_{ij} obtained from statistics is just an empirical value, which represents average kill ability of the weapon in the killing zone. It may be not suitable as the only criterion in real scenario. In order to solve the issue, an ANN model with seven inputs is designed to compute the kill probability dynamically.

2.2 ANN model

It is hard to find the internal relationship between the weapon kill probability and the real time battlefield information. But five aspects can be used to identify influencing factors upon the kill probability for single missile[9], such as launch condition, missile properties, target characteristic, missile target encounter conditions (MTEC) and course shortcut.

Aims to analyze the relationship between the battlefield environments and the kill probability in the paper. Therefore, seven factors are chosen to establish the assessment index set U, where $U=\{$velocity, overload, altitude, MTEC, shortcut, weapon property, target characteristic$\}$. With the assessment indexes as the inputs of the ANN model, the kill probability can be computed in real time. Fig. 1 illustrates the structure of the ANN model of kill probability.

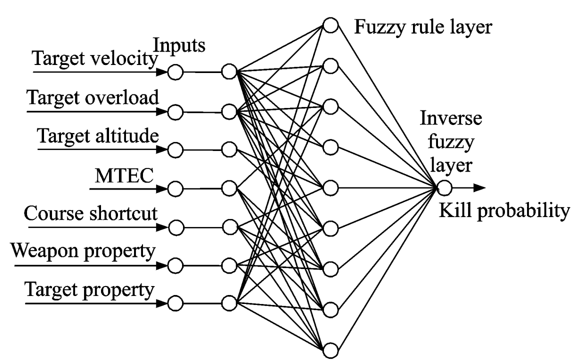

Fig. 1 ANN model structure

Next, each assessment index is expanded, as well as its normalization method.

Experimental result shows that subsonic velocity has little impact on the kill probability. But when the velocity keep increasing, the kill probability will gradually drop down. In addition, different airborne weapons have different speed, such as the speed of armed helicopter is about 0.3 Mach, the speed of guided bomb is nearly 1.3 Mach, and the speed of the fastest fighter can reach 2.5 Mach. Therefore, the higher the target velocity is, the greater the threat is. Based on the above analysis, descending half normal function is applied to deal with the speed normalization, the mathematical expression is given as follows

$$\mu(v)=\begin{cases}1 & v\leqslant a\\ e^{-\left(\frac{v-a}{b}\right)^2} & v>a, b>0\end{cases} \tag{3}$$

where v is the target velocity. a, b are design parameters, which a takes the subsonic value 0.8, and b is determined by the weapon type.

Target overload only exists during aerial maneuver. Advanced air vehicles are with high sustained-G, which makes it able to avoid being hit before missile locked on. Therefore, overload has a deep influence on the kill probability. Thus, overload apparently acts a similar manner as velocity that affect the kill probability. Eq. (3) can also be used as the overload normalization function $\mu(o)$, and a takes the value of 2. And the course shortcut $\mu(cs)$ has the same treatment. The larger the shortcut, the lower the kill probability. Eq. (3) can be also applied, and a, b depend on the specific weapon type.

Target altitude is predicted height at the encounter point in the killing zone. With the increase of altitude, air resistance becomes weaker. It causes higher missile tracking velocity, lower guidance precision and larger miss distance. Eventually, it leads to a smaller kill probability. The kill probability in the horizontal and vertical direction follows normal distribution which is easy to be verified[10]. Then Gaussian function is used to normalize the altitude, expressed as the following formula

$$\mu(h)=\frac{1}{1+\left|\frac{h-c}{d}\right|^{g}} \tag{4}$$

where h is the value of target altitude. c, d, g are design parameters, their values are highly related with the best attack height of the specific weapon.

MTEC can be represented by the missile/target encounter angle. It is an independent variable with common influence on the kill probability. It can cause a big damage when the angle is 20°—60° or 95°—160°. When the angle is less than 10° or bigger than 165° or between 60°—80°, the damage is not big enough. Simulation results that relationship of MTEC and the kill probability is shown in Fig. 2.

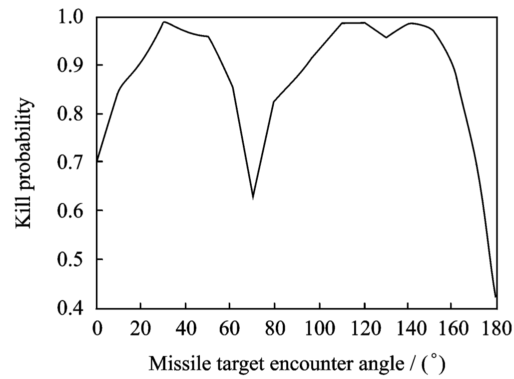

Fig. 2 Relationship of MTEC and kill probability[9]

To describe MTEC effect on the kill probability, the following equation is designed as

$$\mu(\theta)=\begin{cases}\dfrac{1}{1+\left|\dfrac{\theta-35}{38}\right|^{3.5}} & \theta\in(0°,70°)\\[2ex] \dfrac{1}{1+\left|\dfrac{\theta-120}{52}\right|^{8}} & \theta\in(70°,180°)\end{cases} \tag{5}$$

where θ refers to the value of encounter angle.

Target characteristic mainly refers to radar cross section (RCS), radar signal features and infrared characteristic. They are important features that show the target ability of survivability. Considering target characteristic $\mu(tr)$ is not computable, it is classified into three grades: strong (0.7), medium (0.5), weak (0.3). The characteristic value is lower, the target has better chance to survival.

The last index is weapon property. In fact, weapon property is the most important index that determines the kill probability. However, we do not aim to study the weapon intrinsic internal factors. Therefore, it is an indicator to monitor the weapon state. While the weapon is in good condition, it is natural to achieve the best performance. Because it is not computable, the weapon property is divided into four classes, as shown below

$$\mu(w)=\begin{cases}1 & \text{Normal}\\ 0.7 & \text{Damage}\\ 0.3 & \text{Ruin}\\ 0 & \text{Destroy}\end{cases} \tag{6}$$

These classifications indicate four different weapon states:

Normal—the weapon is able to launch and guide missiles to destroy target.

Damage—the weapon is able to launch, and with possibility to guide missiles to hit target.

Ruin—the weapon can not launch missiles, but it is repairable.

Destroy—the damage is beyond repairable.

After all indexes normalized, the ANN model is ready to train. Algorithms such as error back propagation (BP), radial basis function (RBF) are all neural network training methods. As soon as training is done, the kill probability can be obtained from the output of the ANN model. The specific algorithm here is no longer discussed, please refer to Refs. [11—12] for details.

3 Invasive Weed Optimization

3.1 IWO algorithm

Invasive weed optimization (IWO) inspired from weed colonization was first introduced to apply to linear problem by Mehrabian and Lucas in 2006[13]. The algorithm includes four stages.

(1) Initialization

A population of M plants is dispread over the N dimensional problem space with random positions. Meanwhile other parameters such as colony maximum capacity Q, max iteration $\text{iter}_{\max}$, nonlinear modulation index n, min possible seeds production $S_{\min}$ and its opposite $S_{\max}$, initial (final) standard deviations σ_{init} (σ_{final}) should be assigned, respectively.

(2) Reproduction

Each plant is able to produce seeds. The yield is determined by its fitness value, the colony's the lowest fitness $\text{fit}_{\min}$ and the highest fitness $\text{fit}_{\max}$. This step adds a significant property to the search algorithm by allowing all plants to participate in the reproduction contest which lead convergence to the global optimal solution theoretically. Seeds reproduction can be expressed as the following formula

$$S_{\text{individual}}=\left[\frac{S_{\max}-S_{\min}}{\text{fit}_{\max}-\text{fit}_{\min}}\text{fit}_{\text{individual}}\right] \tag{7}$$

where [•] is rounding operation, $\text{fit}_{\text{individual}}$ the fitness of the weed, and $S_{\text{individual}}$ the weed seeds production.

(3) Spatial dispersal

The produced seeds in this stage are being randomly distributed over the search space near their parents. The way to produce seeds is add or subtract a random distance D which follows normal distribution. The current standard deviation (SD) σ_{cur} can be obtained by the following formula

$$\sigma_{\text{cur}}=\frac{(\text{iter}_{\max}-\text{iter})^n}{\text{iter}_{\max}^n}(\sigma_{\text{init}}-\sigma_{\text{final}})+\sigma_{\text{final}} \tag{8}$$

This alteration ensures that the probability of dropping a seed in a distant area decreases nonlinearly at every iteration. It leads to the results that grouping fitter plants and elimination of inappropriate plants.

(4) Competitive exclusion

After all offspring are mature, they will be ranked together with their parents. The colony maximum capacity Q will be reached when the process keeps iterating. Then the plants with poor fitness will be eliminated. The survived plants can produce new seeds based on their ranking in the colony. The process will repeat until termination condition is met.

3.2 Improvement and application

The standard IWO algorithm is proposed based on continuous numerical optimization. However WA is a typical combinatorial optimization problem. The discrete characteristic makes it hard to involve IWO directly. Here an improvement method is introduced on the topic.

First, in order to have a faster calculate speed, fire decision matrix $\boldsymbol{X}$ detailed in Section 2.1 as individual weed is coded in its corresponding decimal form.

Then the algorithm search process can be expressed as follows: Firstly, the seed reproduction process is redefined that a seed (new weed) is born through crossover and mutation by parent itself. Therefore, we can take advantage of the idea of genetic algorithm[14]. But there are some differences in our algorithm. Crossover will only work in the early stages to maintain the diversity of the population. Secondly, mutation takes charge of the control. Distance D in stage three is redefined as the number of mutation element in parent weed. The distance value follows normal distribution and standard deviation σ_{cur} is calculated by Eq. (8). And all elements in seeds have equal chance to mutate. As the iteration proceeds, the number of variation elements will non-linearly drop down. It makes the solution matrix execute from multi-variation to single point mutation which makes the whole process more robust. Fig. 3 illustrates the process of the algorithm.

In order to accelerate the search process, local search is employed to improve the search efficiency. In fact, local search can explore the neighborhood in an attempt to enhance the fitness value of the solution in a local manner[7]. Inspired from immune genetic algorithm (IGA)[14], an immune operator including vaccination and immune selection is applied to the IWO search process.

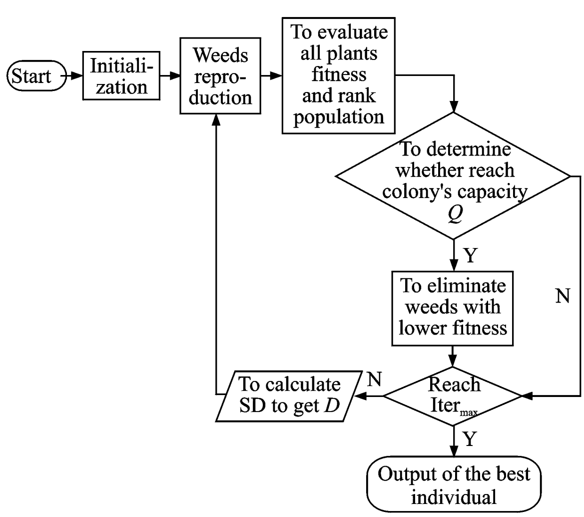

Fig. 3 Flow chart of IWO

4 Cases and Result Discussion

First, the ANN model is trained. A certain type weapon system has the following capability: detection range 21 km, effective height 0.01—6 km, effective hit range 1—12 km, maximum course shortcut 10 km. Therefore, the design parameters mentioned in section 2.2 can be initialized with the specific value which are shown in Table 1.

Table 1 Normalization parameters

Parameter	$\mu(v)$	$\mu(o)$	$\mu(cs)$
a	0.8Ma	2g	5 km
b	0.6	5	2.5

And the altitude normalization formula is expressed as follows

$$\mu(h)=\frac{1}{1+\left|\frac{h-3\ 500}{3\ 000}\right|^{2\times 3.5}}$$

Then an ANN model with four layers is established. Table 2 is the training data used for the network. The last two groups are used to test the results (see Table 3).

Table 2 Training data

T	v/Ma	o/g	h/km	$\theta/(°)$	cs/km	$\mu(tr)$	$\mu(w)$	Kill probability
1	0.8	4	3.0	35	4	0.5	1.0	0.95
2	1.5	6	5.0	50	6	0.7	0.8	0.80
3	0.3	2	2.5	120	3.5	0.3	0.8	0.98
4	0.9	7	3.5	45	1	0.7	0.8	0.90
5	1.3	12	2.0	32	3	0.5	1.0	0.82
6	0.8	10	0.5	15	2	0.7	0.5	0.65
7	2.2	8	5.0	20	12	0.5	0.8	0.50
8	0.6	3	3.5	80	3	0.5	0.8	0.95
9	0.85	4	2.5	150	2	0.7	1.0	0.88
10	0.75	14	0.4	10	1	0.7	0.8	0.72

Table 3 Comparison of kill probability

Target	Theoretical	Trained
9	0.88	0.910 5
10	0.72	0.763 9

To test the performance of the network model, the trained kill probability and the theoretical kill probability are contrasted as follows.

The theoretical kill probability is a statistical average value and it is difficult to comply with the real situation. The kill probability trained by the ANN model can avoid this. When the difference between the trained one and the theoretical one is within the permitted value, the training work is complete. In addition, the outputs from different experiments are not consistent all the time, because the training samples are not enough. With the increase of sample data, the network performance will be improved.

When the training is completed, it can be used to calculate the probability directly. In a scenario with m weapon vs n targets, the kill probability is required to be calculated $m \times n$ times. It is very important for the real-time control application.

Now considering a scenario with 10 weapon vs 10 targets, the kill probability, the target value, and the target lethality are listed in Table 4.

Table 4 Scenario parameters

Weapon	Target									
	1	2	3	4	5	6	7	8	9	10
1	0.789	0.742	0.712	0.682	0.682	0.651	0.603	0.646	0.746	0.884
2	0.842	0.806	0.751	0.706	0.682	0.655	0.674	0.659	0.771	0.745
3	0.654	0.689	0.675	0.661	0.687	0.620	0.635	0.645	0.812	0.842
4	0.318	0.423	0.864	0.857	0.863	0.894	0.941	0.771	0.818	0.756
5	0.157	0.120	0.885	0.876	0.746	0.901	0.798	0.804	0.652	0.521
6	0.254	0.331	0.954	0.934	0.852	0.799	0.798	0.756	0.649	0.786
7	0.245	0.115	0.725	0.779	0.836	0.972	0.952	0.857	0.775	0.756
8	0.961	0.974	0.632	0.354	0.387	0.453	0.528	0.654	0.525	0.741
9	0.963	0.972	0.582	0.642	0.365	0.240	0.102	0.645	0.332	0.741
10	0.254	0.852	0.333	0.560	0.287	0.426	0.539	0.352	0.255	0.489
TarVal	80	75	80	85	85	70	80	90	70	75
TarLty	0.76	0.73	0.85	0.78	0.84	0.89	0.76	0.92	0.74	0.82

Note: TarVal denotes target value, TarLty denotes target lethality.

Assuming that six DAs are under protection, the protection values are 80, 76, 85, 75, 90, 60, respectively. Then the WA problem is solved by using IWO, the needed parameters are listed in Table 5.

Table 5 IWO parameters

Parameter	M	Q	$\mathrm{Iter}_{\max}$	$S_{\max}$	$S_{\min}$	n	σ_{init}	σ_{final}
Value	20	100	50	5	0	3	20	3

Note: the parameter descriptor is provided in Section 3.1.

In order to illustrate the advantages of IWO, genetic algorithm (GA) and IGA are also simulated. Parameters of GA and IGA, such as population size, crossover rate and mutation rate, are set with 100, 0.6 and 0.01, respectively.

Based on the asset-based model, experiments are conducted on the same condition, and Fig. 4 illustrates the comparison results. The average values of three algorithm results are 334.641 8, 370.281 3 and 373.820 2. It can be seen that GA has a huge fluctuation with different initial populations, IGA shows a better performance than GA, and IWO is the best methods with higher convergence and robustness. Although IGA result is sometimes better than IWO, IWO achieves a better performance on the average level.

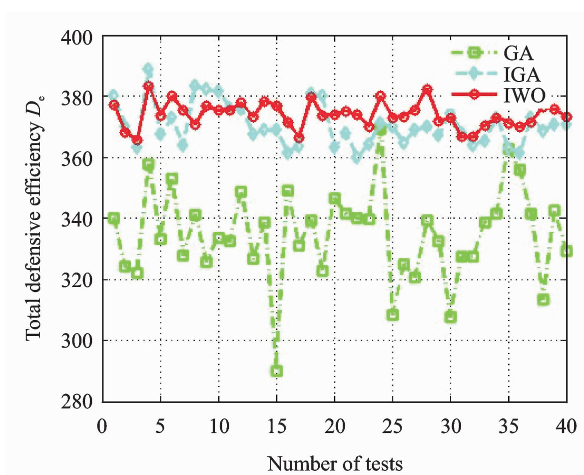

Fig. 4 Experiment results

Next, a more specific convergence process is simulated. As shown in Fig. 5, IWO achieves a higher convergence value than GA and IGA. Due to local search in initial stage, IWO and IGA have higher initialization values than GA. With the iteration increasing, the convergence value of IWO gets better.

In order to analysis the IWO performance,

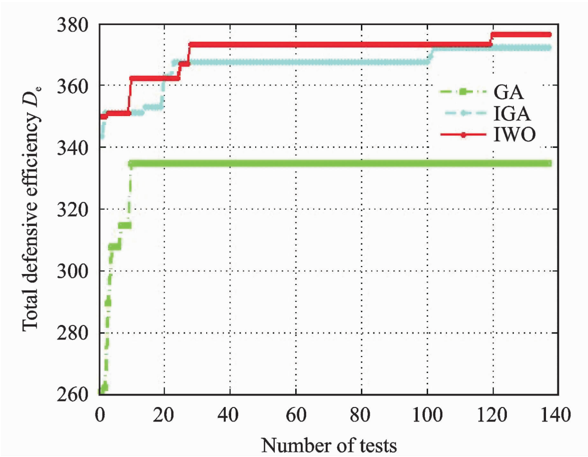

Fig. 5 Convergence process of WA

the experiments with a longer time (5 s, 10 s) are simulated. The best and the worst solutions in different run-time are chosen. Table 6 lists the solutions involved.

It is certain that the optimal solution and the average are better when the run-time is longer. But the worst solutions are closer to the average. For a specific weapon, there are only a few different targets to atlack, for instance weapons 1, 3 are always be assigned to attack targets 10, 9. However, some weapons are assigned to more different targets for the best solutions. For example, weapon 2 is assigned to attack targets 3, 5 and 8 separately in different experiments. In Table 4, we can see that weapon 2 has an average performance. The kill probabilities are 0.842, 0.806, 0.751, 0.706, 0.682, 0.655, 0.674, 0.659, 0.771, 0.745, respectively, with different targets. It is enough to attack any targets. Other weapons have similar explanation. Therefore, as long as the solution achieve a certain level, it is thought good enough to accomplish certain mission. This is just suitable for the application of IWO.

Table 6 Solutions in different run-time

t/s	Solution	Weapon										D_e	Average
		1	2	3	4	5	6	7	8	9	10		
2	Best	10	3	9	5	8	4	6	1	2	7	383.319	374.848
	Worst	9	3	10	5	8	4	6	1	2	7	368.342	
5	Best	10	5	9	7	8	3	6	2	1	4	384.686	378.683
	Worst	10	3	9	5	8	4	7	1	2	6	370.792	
10	Best	10	8	9	5	4	3	6	1	2	7	387.778	382.043
	Worst	10	3	9	5	8	4	6	2	1	7	370.876	

Finally, the model sensitivity is studied. Taken the best solutions in the 10 s for the test objects, the solution variance with the change of a weapon kill probability is observed. Here, we consider that the kill probability of weapon 2 vs target 3 is a variable, and the interval is [0,1]. When the step size is set 0.05, 0.01, 0.005 and 0.001, respectively, the simulation results are shown in Table 7.

Table 7 Simulation variance

Scale	Step size			
	0.05	0.01	0.005	0.001
10 vs 10	2.28	0.456	0.228	0.046
20 vs 20	1.25	0.25	0.125	0.025
40 vs 40	0.56	0.112	0.056	0.011

In Tables 6, 7, we can see that: if there is six default significant figures, then very small change (0.001) on the kill probability can be detected by the WA model, therefore, the sensitivity is 10^{-3} at least. In extreme conditions, take three significant figures for example, the sensitivity is bigger than 0.05 when the battle scale is 40 vs 40. While the normally weapon kill probability varies from 0.6 to 0.9, it is very necessary to adjust the kill probability online.

In addition, as battle scale grows, the sensitivity drops down. Therefore, set proper default significant figures is a key factor to balance high accuracy and sensitivity. In summary, the expectation that the weapon will be adaptively chosen under the current situation is reached.

5 Conclusions

WA has always been one of the key technologies in air defense operations. The paper discusses several questions of WA about its modeling and algorithm application. Taking the advantages of seven factors, a novel model trained by ANN is proposed, which can compute the kill probability under the real-time mode. In further step, an improved IWO algorithm is introduced to achieve the optimal WA result. Simulation results verify the rationality of the model and the effectiveness of the algorithm, which provide a new effective solution for anti-aircraft fire distribution optimization.

References:

[1] Fredrik J, Falkman G. SWARD: System for weapon allocation research & development[C]//13th Conference on Information Fusion (FUSION). Siegen, Germany: IEEE, 2010:1-7.

[2] Lee Z J, Lee C Y. A hybrid search algorithm with heuristics for resource allocation problem [J]. Information Sciences, 2005,173(3):155-167.

[3] Ahuja R, Kumar A, Krishna J, et al. Exact and heuristic methods for the weapon target assignment problem[J]. Operations Research, 2007, 55(6): 1136-1146.

[4] Lee Z J, Su S F, Lee C Y. Efficiently solving general weapon-target assignment problem by genetic algorithms with greedy eugenics[J]. IEEE Trans on Systems, Man and Cybernetics, Part B, 2003,33(1): 113-121.

[5] Wang S L, Chen W Y. Solving weapon-target assignment problems by cultural particle swarm optimization[C]//2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). Nanchang, China: IEEE, 2012,1:141-144.

[6] Ding Zhu, Ma Dawei, Yu Cungui, et al. Application of hybrid optimization strategy algorithm based on tabu search and particle swarm optimization algorithms for weapon-target assignment problems[J]. Acta Armamentaii, 2007,28(9):1127-1131. (in Chinese)

[7] Lee Z J, Lee C Y, Su S F. An immunity-based ant colony optimization algorithm for solving weapon-target assignment problem[J]. Applied Soft Computing, 2002,2(1):39-47.

[8] Zhang Qing, Chen Dandan, Qin Xianrong, et al. Convergence analysis of invasive weed optimization algorithm and its application in engineering[J]. Journal of Tongji University: Natural Science, 2010,38(11):1689-1693. (in Chinese)

[9] Xie Bangrong, Yang Jianying, Ying Jian, et al. Analysis of the influencing factors upon kill probability for single missile[J]. Fire Control & Command Control, 2004,29(2):60-64. (in Chinese)

[10] Zhou Lin, Zhang Wen, Lou Shouchun, et al. Study on the model of intercepting decision for multi channel surface to air missile weapon system[J]. Journal of System Simulation, 2002, 14(6): 698-699. (in Chinese)

[11] Haykin S. Neural networks: A comprehensive foundation[M]. New York: Macmillan, 1994.

[12] Jain A K, Mao J, Mohiuddin K M. Artificial neural networks—A tutorial[J]. Computer, 1996,29(3): 31-44.

[13] Mehrabian A R, Lucas C. A novel numerical optimization algorithm inspired from weed colonization[J]. Ecological Informatics, 2006,1(4):355-366.

[14] Jiao L, Wang L. Novel genetic algorithm based on immunity[J]. IEEE Trans Syst Man Cybernet: Part A, 2000,30(5):552-561.

(Executive editor: Xu Chengting)