# Optimization Deployment of Multi-sensor Platforms in Near-space Based on Adaptive Genetic Algorithm

Yin Yefei
Dept. of Air Defense and Command
Missile Institute, AFEU
Xi'an, China
Navigation2009@163.com

Huang Shucai
Dept. of Missile Engineering
Missile Institute, AFEU
Xi'an, China

***Abstract*—Four optimization deployment performance indexes have been proposed for the problem of multi-sensor platforms optimization deployment in near-space. An Adaptive GA algorithm is used to optimize the latitude and longitude points' positions of multi-sensor platforms in near-space. Then a linear fitness function is constructed based on the four indexes. The crossing probability and mutation probability have been made to adapt to the change of the fitness values. For the appointed ground area, optimization deployment simulations of multi-sensor platforms are carried out and the latitude and longitude values of every platform for different number of platforms have been calculated. The optimization results demonstrate that the indexes proposed in this paper are reasonable and the latitude and longitude values calculated by the Adaptive GA algorithm completely meet the need of multi-sensor platforms deployment, and the optimization speed also meets the synchronization's need for acquiring informations.**

***Keywords-near-space; multi-sensor platform; multi-sensor co-sight; adaptive genetic algorithm; adaptive crossing and mutation***

## I. INTRODUCTION

The near-space is a very important space for application which is above the troposphere. It is usually assumed that it ranges from 30km to 120km above the surface of the earth, which means it is the transition space of aviation and space flight [1]. So hardly any of the existed aircrafts or satellites could operate in near-space. But now some new platforms have been developed which could operate in it, such as balloons, airships and other lighter-than-air vehicles. Therefore, sensors carried by these platforms in near-space have the following advantages. First, defending weapons become out of reach of them, so they could operate safely. Secondly, their operating height is lower than that of the low-orbit satellites and higher than that of the aviation aircrafts, on one hand, their detection precision is better than the former, on the other hand, they would have a larger detection and surveillance area than the latter without reducing the precision very much. Thirdly, the number of sensors and platforms is between the aviation case and space case, so it is cost-effective, and for the reason that it can operate safely, we can execute assignment and deployment to the platforms dynamically, which makes the needed number of platforms reduce sharply, also for that reason, the precision could also be improved obviously. With the being improved techniques of platform in near-space, the application of near-space has been promoted by developed military countries.

By now, the USA have kept ahead on low-orbit satellites, airships, high altitude balloons, unmanned surveillance vehicles and high altitude vehicles propelled by rocket engines. A new detection and surveillance network consists of these platforms with sensors carried on them have been set up gradually, by the year of 2005, the USA Air Force has developed a testing detection subsystem of near-space using low-orbit satellites, airships and high altitude unmanned vehicles.

The scientists and engineers in our country have implemented corresponding research and study in the early 90's last century, and have acquired some improvements on low-orbit satellites, high altitude balloons and airships. At the beginning of the new century, the high-tech program of "863" has set up an exclusive fund on the research for application of near-space.

In this paper, four indexes have been presented to optimize the deployment of multi-sensor platforms in near-space. With their weighted sum, we construct a linear optimization objective function, which is used as the fitness function of the adaptive genetic algorithm (AGA) applied in the simulation. We compute the best individual through the evolution progress, and then decode the best individual to acquire the positions of platforms. The AGA is able to reset the crossing probability and mutation probability adaptively to adapt to the change of fitness value, which prevents genetic algorithm from converging too fast and assures the AGA converge to the global best solution. The simulation results have proved that the indexes presented in this paper is reasonable and the optimization algorithm is able to compute the best deployment scheme.

## II. OPTIMIZATION DEPLOYMENT INDEXES OF MULTI-SENSOR PLATFORM

The detection network in near-space mainly has two influencing factors which determine the detection precision and detection scope. One is the detection performances of the sensors; the other is the dynamic deployment of multi-sensor platforms. Usually the detection performance indexes of sensors have been set as fixed values, so dynamic deployment of multi-sensor platforms play a dominant role in optimization detection. Therefore, we present the following four indexes to

Funded by the National "863" Hi-tech Program

978-1-4244-4994-1/09/$25.00 ©2009 IEEE

describe the affection of dynamic deployment of multi-sensor platforms [2][9]. Then a linear model of optimization deployment is set up based on the indexes.

### A. Area coverage proportion percentage index P aintaining the Integrity of the Specifications

Area coverage proportion percentage index is a very important performance index to assess the detection capacity of detection network.

For the multi-sensor multi-sensor platforms detection network in near-space, we assume that the area observed is bounded by longitude and latitude, that is

$$\text{Longitude: } \lambda \in (\lambda_{\min}, \lambda_{\max}) \text{ ; latitude: } \varphi \in (\varphi_{\min}, \varphi_{\max}) \quad (1)$$

The grid statistics method is used to compute the ACPP index P. We divide the rectangle into many small cells, which is encircled by the minimum longitude, the maximum longitude, the minimum latitude and the maximum latitude, and then each cell is regarded as an eigenpoint to be analyzed. In order to make each of the cells has equal area; the number of grid cells on latitude is set to be proportionate to the cosine of latitude. Each platform covers a certain number of grid cells indexed by $n_i$, and the whole detected area covers the total number of grid cells N. If there are m platforms, so all of the platforms will cover $n=\sum n_i$ $(i=1,\cdots,m)$ grid cells on premise of no overlapping. Using n to divide N, we will get the ACPP index P. Furthermore, the index P will get its maximum value if the detecting areas of the m platforms are all enclosed in the observing area without overlapping.

### B. Ground target overlaying index X

We assume there are N ground targets represented by $T_1, T_2, \cdots, T_n$. The position of Target $T_i$ is $(\lambda_i, \varphi_i)$ and its threatening grade weight is $\omega_i$ which can be computed by AHP method [8]. That is, by comparing all the targets one by one to acquire the relative importance, we use the relative importance as the threatening grade weights. We have the variable $x_i$ denoting whether the ground target Ti is overlaid.

$$x_i = \begin{cases} 1 & \text{target } T_i \text{ is overlayed} \\ 0 & \text{target } T_i \text{ is not overlayed} \end{cases} \quad (2)$$

Furthermore, we define the ground target overlaying index X of platform i as following:

$$X = \sum_{i=1}^{N} \omega_i x_i \quad (3)$$

It is applied to assess the capacity of platform $i$ on overlaying ground targets.

### C. Ground target resolution index R

Index $r_i$ is used to represent the resolution of target $T_i$, the smaller $r_i$ is, the higher resolution of target $T_i$ is. So for all the targets, we define the ground target resolution index (GTR) as the following:

$$R = \sum_{i=1}^{N} \omega_i r_0 / r_i \quad (4)$$

Where the index R is used to assess the resolution performance of sensors for ground targets, and $r_0$ is the vertical resolution of platform for ground targets under it.

### D. Co-sight index C for multi-sensor

Multi-sensor crossing detection will increase the detection probability and improve the detection precision. Therefore, the number of ground targets which lie in the crossing detection area of multi-sensor also is one of the most important indexes to assess the performance of multi-sensor multi-platform network. We name the index as co-sight index. Variable $c_i$ is to used to describe whether ground target $T_i$ lies in the multi-sensor crossing detection area, that is

$$c_i = \begin{cases} 1 & T_i \text{ lies in the crossing detection area} \\ 0 & T_i \text{ dosen't lie in the crossing detection area} \end{cases} \quad (5)$$

So the Crossing-detection index C for multi-sensor is define as:

$$C = \max \sum_{i=1}^{N} \omega_i c_i \quad (6)$$

Where, the parameter $C$ is used to assess the capability of multi-sensor for overlaying multi-target, and $\omega_i$ is the threat grade weight of each target.

## III. The Adaptive GA Method for Optimization Deployment of Multi-Platform

In this paper, the optimization deployment of multi-sensor platforms is mainly based on the above four performance indexes. We utilize them to construct a optimization objective function and transform the problem of optimization into the problem of single objective function optimization. We rank the four indexes depending on their importance as the following: ground target overlaying index X, area coverage proportion percentage P, Crossing-detection index C for multi-sensor, Ground target resolution index R. We assume that the normalized weights of the four indexes are 0.4, 0.3, 0.2, and 0.1 separately, and then the objective function is defined as:

$$f = 0.3P + 0.4X + 0.1R + 0.2C \quad (7)$$

Genetic algorithm is derived from the evolution theory that inferior individual will be eliminated through selection or contest and the most adaptive one will exist in the nature species evolution process. It has been widely applied in the optimization field [3] [7]. It uses a individual coded by genes to represent a potential solution for a certain problem, and makes the group of a number of individuals represent the potential solution sets. After the initial group is generated, then through evolution, the best group and individuals will be generated.

Adaptive GA method in based on Simple GA method, which makes the crossing probability $P_c$ and mutation probability Pm adapt to the change of fitness value adaptively. While the fitness values of individuals in the group converge to local best point conformably or simultaneously, the crossing probability $P_c$ and mutation probability $P_m$ should be increased. Otherwise they should be decreased. Furthermore, the individual whose fitness mutation probability value exceeds the average fitness value of the group should have smaller $P_c$ and $P_m$ and should be preserved to get into next generation, while the individual whose fitness value is under the average fitness value should have bigger $P_c$ and Pm and should be eliminated. Therefore, the adaptive $P_c$ and Pm will generate the best individual corresponding to one of the potential solutions. The

Adaptive GA not only preserves the group diversity, but also guarantees the convergence [3] [4] [6].

The steps of optimization deployment algorithm using Adaptive GA for multi-platform in near-space are described as the following:

①Generate the initial group

The platforms are assumed to be deployed at a certain altitude. We utilize the real number coding method to generate the initial group, which is generated through uniformity random sampling between the minimum latitude and maximum latitude, as well as between the minimum longitude and the maximum longitude. The number of groups is N, each individual consists of $m$ vectors which is composed of the longitude and latitude of a platform. The structure is described as following:

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |

②Cross the individuals

We use the middle-recombination method to execute the operation of crossing, that is,

$$Y' = Y_1 + \alpha \cdot (Y_2 - Y_1) \tag{8}$$

Where, Y and Y' represents the father generation individual and filial generation individual respectively, α is a random number between 0 and 1.

The adaptive crossing probability can be selected through the following formula.

$$P_c = \begin{cases} P_{c1} - \dfrac{(P_{c1} - P_{c2})(f' - f_{avg})}{f_{\max} - f_{avg}}, & f' \ge f_{avg} \\ P_{c1}, & f' < f_{avg} \end{cases} \tag{9}$$

Where, $P_{c1} = 0.9$， $P_{c2} = 0.6$ .

③Compute the fitness value

We can straightforwardly compute the fitness value of each individual according to (7).

④Select the individuals of the next generation

In step ③, we can acquire $2N$ father generation individuals and filial generation individuals together, then sequence them in the order of their fitness values from the greatest to the least and select the best $N$ individuals as the new generation individuals.

⑤ Mutation

Real value mutation is adopted. The key is the selection of the mutation step. It can be changed to adapt to specific cases, small step has higher probability of success, but large step has higher convergence pace. So we adopt the following mutation operator[5]:

$$X' = X \pm 0.5L\Delta \tag{10}$$

The adaptive mutation probability can be selected according to the following formula.

$$P_m = \begin{cases} P_{m1} - \dfrac{(P_{m1} - P_{m2})(f_{\max} - f)}{f_{\max} - f_{avg}}, & f \ge f_{avg} \\ P_{m1}, & f < f_{avg} \end{cases} \tag{11}$$

Where, $P_{m1} = 0.1$ , $P_{m2} = 0.001$ , $\Delta = \sum_{i=0}^{m-1} \dfrac{a(i)}{2^i}$, and $a(i)$ is equal to 1 at the probability of 1/m and 0 at the probability of 1-1/m, usually it is assumed to be equal to 20. L is the variable range, for the platform vertical projection point longitude and latitude, it follows (1). $X$ is the real value before mutation, $X'$ is the result value after the mutation operation.

⑥ Estimate whether new generation group meets the halting condition, if meets, stop evolution, or else transfers to step ② to continue evolution computation.

## IV. RESULTS AND THEIR ANALYSIS OF SIMULATION FOR A SIMPLE SCENARIO

We assume that the detecting area of multi-sensor multi-platform network in near-space is described by longitude and latitude as the following:

*Longitude*： $\lambda \in [E119.5°, E122.5°]$ *Latitude*： $\varphi \in [N20.5°, N26.5°]$

There are 9 important targets needed to detect and surveillance, their positions and threat grade weights are described as table 1.

Table 1 positions and threat grade of ground targets

| Ground targets | Positions（Lon,Lat） | Threat grade weights |
|---|---|---|
| T1 | (N24.62，E120.64) | 0.3070 |
| T2 | (N22.32，E120.58) | 0.0370 |
| T3 | (N23.59，E121.03) | 0.0189 |
| T4 | (N21.98，E120.96) | 0.1543 |
| T5 | (N22.76，E121.2) | 0.2182 |
| T6 | (N25.1，E121.5) | 0.0533 |
| T7 | (N24.81，E121.95) | 0.1089 |
| T8 | (N23.19，E120.2) | 0.0764 |
| T9 | (N19，E117) | 0.0260 |

*Optimization results for simulation are given as the following:*

1) Four platforms

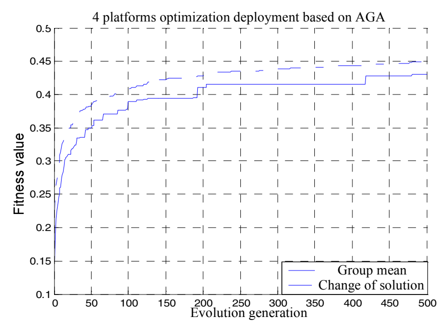


Figure 1 First optimization results

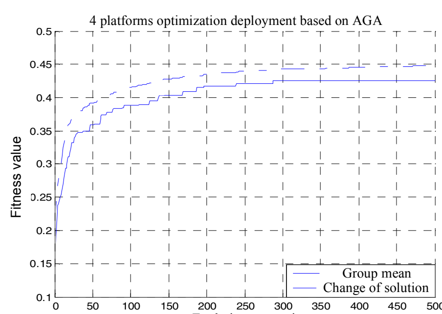


Figure 2 Second optimization results

We do the same simulation twice. Figure 1 and Figure 2 showed the results respectively. The maximum fitness values converge at *maxfitness*1=*0.45*, *maxfitness*2=*0.445*, respectively. The best individuals are showed as the following:

*bestindividual*1=[120.9516,22.6905,121.1452,20.6905,121.1458,22.0238,121.1252,21.6429];

*bestindividual*2=[119.7903,26.4048,120.0806,20.7857,120.0806,24.2143,122.4032,22.5952]

There is one problem need to note, that is, the AGA have acquired the suboptimal solution at the 300$^{th}$ generation, but for the reason of limited overlaying area of platform and too many targets, the fitness value may uniformly converges at the same point, but the best individual may be different for each simulation. This is not unreasonable, it can explained through theory and engineering.

2) Three platforms

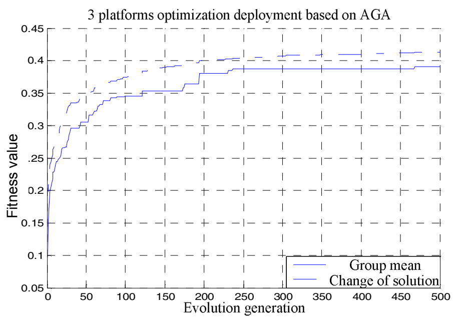


Figure 3 First optimization results

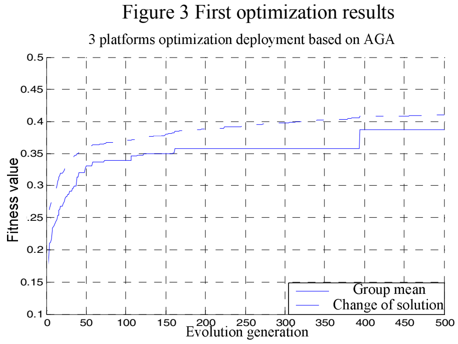


Figure 4 Second optimization results

We also do the same simulation twice. Figure 3 and 4 showed the results respectively. The maximum fitness values converge at *maxfitness*1=*0.392*, *maxfitness*2=*0.386*, respectively. The best individuals are showed as the following:

*bestindividual*1=[122.3065,20.9762,119.5968,22.8810,121.1452,25.7381];

*bestindividual*2=[120.0806,21.1667,119.5000,21.9286,119.5968,20.7857].

3) 2 platforms

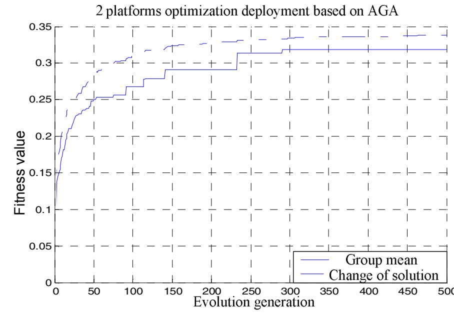


Figure 5 First optimization results

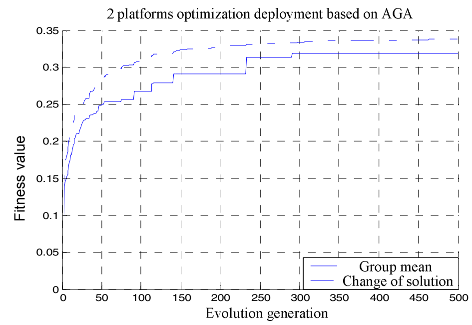


Figure 6 Second optimization results

For 2 platforms, Figure 5 and 6 showed the results respectively. The maximum fitness values converge at *maxfitness*1=*0.326*, *maxfitness*2=*0.332*, respectively. The best individuals are showed as the following:

*bestindividual*1=[120.8548,24.7857,120.7581,26.2143];

*bestindividual*2=[122.3065,26.2143,119.5968,25.1667].

4) Single platform

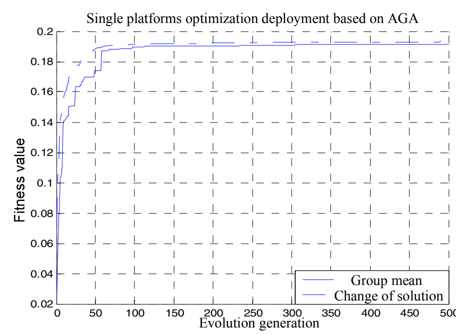


Figure 7 First optimization results

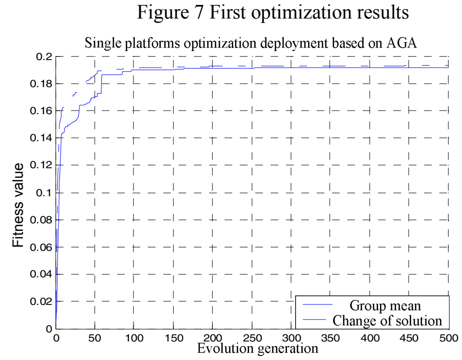


Figure 8 Second optimization results

For the single platform, Figure 7 and 8 showed the results respectively. The maximum fitness values converge at *maxfitness*1=*0.192*, *maxfitness*2=*0.193*, respectively. The best individuals are showed as the following:

*bestindividual*1=[121.9194,21.8333];

*bestindividual*2=[121.3387,22.2143].

From the simulation results, we could find that the smaller number of platforms is, the less the needed number evolution generations is and the faster AGA converges. For the reason of linear fitness objective function which is constructed by the four indexes has fixed weights for each index which is time-variant in-fact, so this optimization objective function do not always suitable for some specific evolution steps, which leads to generating fake best individual.

Moreover, the contrast relation between the number of platforms and the number of targets will have influences on the unique best solution. If the former is larger than the latter, there may be several best individuals; otherwise, there may be only one best individual. Therefore, the optimization objective function still needs to be improved to take the number contrast relation into account.

## V. CONCLUSION

From the simulation results, we find out that there are many results that meet the conditions. The positions of platforms are selected randomly which is caused by the random initiation of groups, but actually, the initial positions of platforms are not random, so we have to take the following two conditions into account, that is, the optimal path and the least number of tasks. Then the results meet the optimization conditions would reduce sharply. However, the method presented in this paper still can acquire better results for multi-sensor multi-platform optimization deployment in near-space.


## ACKNOWLEDGMENT

I would express my gratitude to the National high-tech program “863” for the fund it provides to me. Also I should attribute the success of my study works to my doctor teacher Mr. Huang, since he gives me so many instructions and so much help.


## REFERENCES


[1] Hampton Stephens. Near-Space.AIR FORCE Magazine, 2005,(7):38~39

[2] Huang Shucai. Theatre air defense sensor resource optimization deployment and assignment algorithm. Xi’an: Air Force Engineering University, 2005.

[3] Wang Xiaoping, Cao Liming. Genetic Algorithm---Theory, Application and Software Implementation. Xi’an: Publishing House of Xi’an Jiaotong University, 2002.3-6.

[4] Jiao Licheng, Bao zheng. Evolution Computation and Genetic Algorithm. System Engineering and Electronics, 1995, vol 6: 22-24.

[5] Srinivas M, Patnailk L.M. Adaptive Probabilities of Crossover and Mutations in Gas. IEEE Trans. On SMC, 1994,24(4): 656-667

[6] Zhang Xuejiang, Zhu Xiangyang, etc. Apaptive Genetic Algorithm and Its Application in Knowledge acquiring. System Engineering and Electronics,1997,29(6):67-72

[7] Zhou Ming, Sun Shudong. Genetic Algorithm Theory and Its Application. Beijing: Publishing House of National Defense Industry, 1999.2-3.

[8] Chen Qifeng, Dai Jinhai, Zhang Yukun.Evolution Algorithm for Aera Overlaying Structure and Synchronous Parameter Optimization. System Engineering and Electronics ,2004,26(4):549-552

[9] Jin Guang, Wu Xiaoyue, Gao Weibin. Resource Assignment Optimization Model for Ground Stations of Satellites and Its Heuristic Algorithm. System Engineering and Electronics, 2004,26(12):1839-1841