TELKOMNIKA, Vol.11, No.2, February 2013, pp. 623~629
ISSN: 2302-4046

A Quantity Optimization Method on Integrated-Loading-and-Unloading-Missile Vehicles

Liu Ai-zhen*, Chen Li-yun, Wang Yin-long, Wang Lu
Department of Information Engineering, Ordnance Engineering College, Shijiazhuang, China, 050000
*Corresponding author, e-mail: liuaizhen_163@163.com

Abstract

In this paper, an optimization method based on Genetic Algorithm is proposed to solve the integrated-loading-and-unloading-missile vehicle assignment problem. On the basis of analyzing the characters of high cost, low usage and real time loading with missile and the vehicle, the mathematical model was built with the bi-objective of cost and loading time. And the resolving method based on Genetic Algorithm was specially designed to get the optimum allocation scheme. Considering particularity of the problem, variable-list encoding method and valid initial population generation method, special one chromosome cross genetic operator; special one chromosome greedy-and-balanced migration operator and mutation operator are proposed. Simulation results show that the solutions by this algorithm can complete all loading tasks, and also can get an optimum balance between cost and loading time of missiles, which can meet the character of accurate support during wartime.

Keywords: *Integrated-loading-and-unloading-missile vehicle, Accurate Support, Genetic Algorithm, Genetic Operators*

1. Introduction

New military reform has been influencing and changing every aspects of military field profoundly. As an important content during wartime, the way of ammunition support is undergoing a process of transforming from “rough calculation” to “accurate support” [1,2]. In order to complete the ammunition support task with low cost and high efficiency, it’s very meaningful to rationally ensure numbers of kinds of support equipments according to the needs of battlefield. According to the process of ammunition support in future warfare, the types and quantity of all kinds of ammunition that the battlefields needs could be real-time perceived and ascertained. Among them the ammunition of precise-navigation weapons, missile, has special characters. The missile is high cost and is generally used on key or important military targets. So the number of missile is extremely less than other ammunitions. Since its cost and function missiles use must be tested before being used in order to ensure the ammunition is in good condition in battlefield. Then missiles are transported by special vehicles such as the integrated-loading-and-unloading vehicle used by the army. Also, the kind of vehicle is high cost and can loading several missiles once. It is not only responsible for transporting missiles but also responsible for loading missiles to the missile launchers in real time before missiles will be launched. At present, researches on transport equipment in the army are focused on the quantity of the loading method based on universal transporting and loading vehicles [3]. As far as few people research the accurate quantitative method of the kind of special vehicle with considering the launch time of some missile launching vehicles in battlefields. Currently, the number of the vehicles are always obtained from experience or estimation of experts. The methods are lack of rational quantitative analysis, lead to consequences like a waste or lack of the support resource. If resources is overestimed, it will lead to some problems such as unnecessary vehicles usage, high cost, and more difficult support task; on the contrast, it will lead to irretrievable effect on the war due to underestimation. To alleviate the defect of traditional estimation methods, the problem was anlysed, the mathematical model was defined, and an optimum method based on Genetic Algorithm was employed to get the optimum number of the vechile for meeting the real time loading needs of missiles in battlefields.

Received October 5, 2012; Revised December 21, 2012; Accepted January 11, 2013

2. Description and Establishment of Mathematical Model

Some field forces are equipped with a number of missile launcher (assuming M units), which is used to launch missiles in battlefields. Kinds of parameters (such as location, the next launch time, prearranged missile loading time) on each launcher are known before launched. Adapting the needs of the missile launcher, missiles must be loaded to specific launcher in permitted time, so the integrated-loading-and-unloading vehicles are always support launcher with accompany ways. Let's assume the number of the vehicles a variable N, the question are divided into two kinds of considerations as follows:

1. M≤N. If M≤N, this case shows that there are enough or more integrated-loading-and-unloading missile vehicles compared with missiles. Each launcher can be supported by one integrated-loading-and-unloading missile vehicle. The number of vehicles can meet the need of launchers without any optimization planning.

2. M>N. If M>N, it means that the number of missiles is more than the number of vehicles. One integrated-loading-and-unloading missile vehicle must be assigned to support several missile launchers. In this case, we need to calculate how many integrated-loading-and-unloading missile vehicles can not only satisfy the real-time battlefield needs, but also reduce the cost of vehicles. Therefore all relevant factors must be considered to optimize the schedule of the vehicles.

We will analyze and build a model for the latter case.

According to the characteristics of the launcher, meeting the demands of all missiles' launch times should be as the primary target. And the relevant factors are set as follows.

1. Since the kinds of information such as layout of the M sets of launcher can be attained, then before the missiles be launched the distances (denoted as matrix D) among the launchers can be assumed as following.

$$D = \begin{bmatrix} d_{11} & d_{12} & ... & d_{1M} \\ d_{21} & d_{22} & ... & d_{2M} \\ ... & ... & ... & ... \\ d_{M1} & d_{M2} & ... & d_{MM} \end{bmatrix}$$

2. Each launcher mainly has the following time states: loading time, launching time and stopping time. Accompany with the launcher states, each integrated-loading-and-unloading missile vehicle on the battlefield may have the three time states: maneuvering time, loading time and waiting time.

3. M sets of launcher next launch times (denoted as matix T) were as follows.

$$T = \begin{bmatrix} t_1 & t_2 & ... & t_M \end{bmatrix}$$

4. M sets of launchers have the same loading time denoted as LT (load time), so the latest loading time denoted as matrix TT were as following.

$$TT = T - LT = [tt_1, tt_2, ..., tt_M]$$

5. The missile number each loading and unloading vehicles remain were as follows:

$$A = [a_1, a_2, ..., a_N] \quad (a_i > 0)$$

6. The distances matrix (denoted as DD) of the integrated-loading-and-unloading vechiles is away from each launcher to be loaded are as follows:

$$DD = \begin{bmatrix} dd_{11} & dd_{12} & ... & dd_{1M} \\ dd_{21} & dd_{22} & ... & dd_{2M} \\ ... & ... & ... & ... \\ dd_{C1} & dd_{C2} & ... & dd_{CM} \end{bmatrix}$$

7. Assuming the integrated-loading-and-unloading vechiles have a speed of V, the times (denoted as matrix DT) they spend from current location to each launcher to be loaded are as following:

$$DT = dd_{ij} / V = \begin{bmatrix} dt_{11} & dt_{12} & ... & dt_{1M} \\ dt_{21} & dt_{22} & ... & dt_{2M} \\ ... & ... & ... & ... \\ dt_{C1} & dt_{C2} & ... & dt_{CM} \end{bmatrix}$$

As considered here is the situation of M>N, that is the integrated-loading-and-unloading vechile must be assigned to support for several missile launcher. Then N vehicles means N loading sequences, each sequence includes a number of launchers location list, the vehicles must complete loading missiles task for each launcher in list order. Whether the launchers times are fulfilled mainly depends on the number of the vehicles. Due to the time factor of the different launchers times, the more of vehicles the better support efficiency. But considering the cost factor the less the better. So the number of vehicles is a key factor getting a balance between cost and time.

Therefore it's key to find a method, with which we can determine the number of the entire loading and vehicles loading list sequence, and each loading time of the list should be no later than the latest loading time. The time all the vehicles spend is equal to the maximal time that derived the vehicles spending the most time to complete its sequence.

First, calculate the time that a vehicle loads its sequence. Assume finished loading the cars ready sequence corresponding to the launch vehicle from the entire unloading (assuming Vi is the sequence set that is chosen by the No. i units. Vi=（V_{i1},…, V_{ij}, ……, V_{in}）, Where n <M (n for the number of elements in the set), then the time the No.i units finished loading sequence corresponding to the launch vehicle from the entire unloading time can be calculated:

$$\begin{aligned} T_{iseq} &= t_{infinish} \\ &= AT_{in} + LT_{in} \\ &= t_{i(n-1)finish} + MT_{i(n-1,n)} + LT_{in} \\ &= t_{i(n-2)finish} + MT_{i(n-1,n-1)} + LT_{i(n-1)} + MT_{i(n-1,n)} + LT_{in} \\ &= MT_{i(0,1)} + LT_{i1} + MT_{i(1,2)} + LT_{i2} + ... + MT_{i(n-1,n)} + LT_{in} \\ &= \sum_{j=1}^{n} (MT_{i(j-1,j)} + LT_{ij}) \end{aligned} \quad (1)$$

s.t.

$$t_{ijfinish} = \sum_{k=1}^{j} (MT_{i(k-1,k)} + LT_{ik}) \le t_j \quad (2)$$

Tiseq---- The loading time the i-th sequence of the integrated-loading-and-unloading vechiles.（0<i<N，i is an integer）; tj----- The launch time the j-th launch vehicle selected by the set .（0<j<M, j is an integer; (2) is constraints, the actual load time of each launch vehicle in the sequence is less than or equal to the latest loading time .

Similarly, we can calculate sequence of loading time of other units and then get the loading time Tseq of N units. Obviously, different values of N mean different Tseq. Even the same N, different sequence may cause different Tseq. So, if we want to get the minimum number of integrated-loading-and-unloading missile vehicles that meet the need of each launch vehicle, we must not only consider the number of N, but also the sequence of the N vehicles , thus we can get ideal result.

In summary, show as the following mathematical formula:

The objective function is as follows.

$$X=\min(N) \quad (3)$$

s.t.

$$Tseq \le T_{max}$$

$$Tseq = \max(T_{1seq}, T_{2seq}, ..., T_{Nseq}) \quad (4)$$

Note:

N--number of the integrated-loading-and-unloading vechiles

Tiseq-- the time the i-th loading and unloading vehicles finished loading the selected launch vehicle series. Calculation method, see (1), (2).

3. Solving the Model

The problem is a kind of combinatorial optimization problems. Here we adopt Genetic Algorithm to solve.

3.1 Encoding Strategy

First, let $t = [t_1 \quad t_2 \quad ... \quad t_M]$ be the set of missile-launch-vehicle emission time. (here subscripts 1,2, ..., M indicates the number of launch vehicles, $t_1,\ldots,t_M$, indicates the launch time) then let N be the quantity of integrated-loading-and-unloading missile vehicles. And let the set V_1, V_2, …,V_N be the number of its vehicles. Encoding strategy is designed as following.

The quantity of vehicles + every vehicle assigned missile-launcher list. Illustrates it as Figure 1. (M=13, N=4).

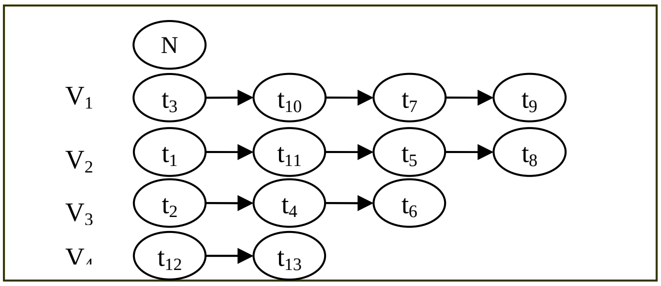

Figure 1. Encoding Strategy

3.2. The Fitness Function

The fitness function used to evaluate all the individuals in the population, and control the genetic operation. To find the minimum value N and the dispatch list, which meet the scheduling launch time of the missile-launch-vehicles. But the selection operation in genetic manipulation is always trying to fitness maximize. Therefore, define fit(s) as the fitness function. s.t.

$$fit(s) = MAX - T_{seq}(S) \quad (5)$$

In order to ensure fit(s)≥0, the MAX is a sufficiently large constant.

3.3. Generation Process of the Initial Population

Step1: Let p be the quantity of population, and set new launcher time sequence t' equaling to t sorted time by minim to maxim. Here $tt' = t' = [t_{o1}, t_{o2}, ..., t_{oM}]$ are the new sorted time sequence; and counter variable i=0.

Step 2: Select N randomly from [2,M-1];

Step 3: If t' <> Φ, go to step 4;
else tt' =t'

Step 4: First, get an identify number x from the set V_N randomly; and randomly get a less launcher time vehicle y according to tt'; then calculate the time that the number of integrated-loading-and-unloading missile vehicles arrive on y. and adding on the time of loading gets the actual launch time T_y, if $T_y \le t_y$, then add it to the task list; then delete t_y which corresponding to y from tt'; If $T_y > t_y$, start to re-select a less launcher time vehicle y according to tt' and re-calculation of Ty, long as to $T_y \le t_y$.

Step 5: Repeatly execute step 4 up to t' is empty. That all missile launch vehicle has been fully allocated and the current schedule has been generated.

Step 6: i=i+1;

Step 7: if i<p, Go to step 3;
else this means the initial population has been generated, exit.

3.4. Genetic Operators
3.4.1. Cross operator based on a chromosome

Particularly, it's hard to make sure the regulation to the chromosomes can meet the launch-time's need while two chromosomes cross randomly. So we design a cross operator for one chromosome. It makes the chromosome meet the launch-time's need while the chromosome crossed inner itself. The process is shown as following.

Step 1: Select a chromosome randomly (e.g: in Figure 2).

Step 2: Select a task from one sequence of the integrated-loading-and-unloading vechiles randomly which from the selected chromosome in step1 (e.g: t_{13} in V_4), then compute its actual finish-loading time, tagged it as t_{13}.

Step 3: Compute and find out a task set A whose actual finish-loading time is the first exceeding over T13, if A is null, return step 1 and reselect a chromosome, or else turn to step 4.

Step 4: Suppose A isn't null. Select an element from set A randomly (e.g: A={ t7 , t11}),e.g: t7 , note the sequence V_1 which contains t_7.

Step 5: Exchange all the latter tasks after corresponding tasks in V_1 and V_4.

3.4.2. Greedy-and-Balanced Migration Operator

In order to increase the convergence speed and get optimum solution as soon as possible, a kind of greedy-and-balanced migration operator is designed. Take Figure 2 and Figure 3 for example to illustrate how the migration process is:

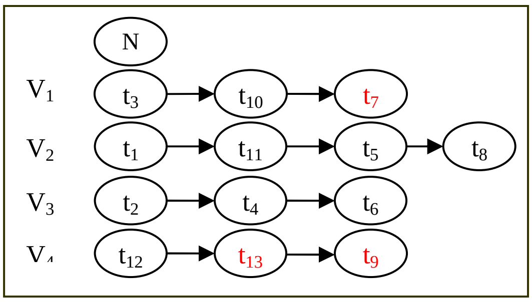

Figure 2. Chromosome After Using Crossing Operator

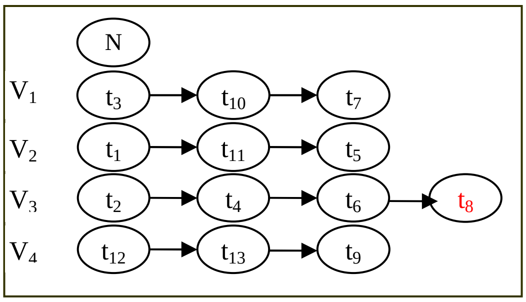

Figure 3. Chromosome After Using Greedy-And-Balanced Operator

1) Randomly selected a chromosome, here we choose the one shown as Figure 3.

2) Computing the actual loading beginning time of the last task of each integrated-loading-and-unloading missile vehicles, take them in time order, suppose it as : $t_6 > t_7 > t_9 > t_8$;

3) Choosing the last task (t_8) of latest beginning vehicles, move it randomly to the sequence whose finishing time is earlier for loading request(such as moving to after t_6 in sequence V_3), then the chromosome changes to the one as shown in Figure 3.

3.4.3. Mutation operator

First choose one chromosome randomly, then choosing one assignment of two vehicles randomly, finally exchange the chosen assignment. The transformation process is as follows (Figure 3, Figure 4, for example).

1) Randomly select task t_{11} from V_2
2) Randomly select a corresponding task list from another vehicle, assume it as V_3 , computing the loading time the list of tasks actually finished, using t_2 ($T_{v3(i-1)}$) ($T_{v3(i-1)} \le t_{11} \le T_{v3i}$) replace t_{11} to the appropriate location of the queue V_3. In the same way, we replace t_{11} from V_3 to a appropriate location in V_2.
3) Recalculation the actual loading time of the two task list, and updates.

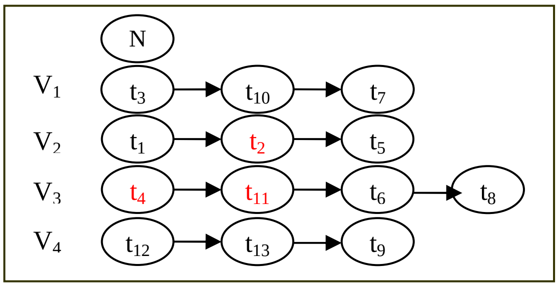

Figure 4. Chromosome After Using Mutation Operator

4. Simulatation Results and Analysis

In order to test the validity of the algorithm, we simulate calculation many times, it's done well . The following are the main data and the results of one simulation operation.

Experimental data is set as follows: take 10 missile launch vehicles, distance between two of them (unit:km).

$$D = \begin{bmatrix} 0 & 2 & 7 & 6 & 9 & 6 & 14 & 15 & 13 & 13 \\ & 0 & 4 & 5 & 8 & 7 & 14 & 14 & 12 & 12 \\ & & 0 & 4 & 6 & 10 & 11 & 11 & 8 & 10 \\ & & & 0 & 3 & 7 & 7 & 9 & 8 & 6 \\ & & & & 0 & 9 & 6 & 6 & 5 & 4 \\ & & & & & 0 & 10 & 15 & 15 & 12 \\ & & & & & & 0 & 6 & 9 & 3 \\ & & & & & & & 0 & 6 & 4 \\ & & & & & & & & 0 & 7 \\ & & & & & & & & & 0 \end{bmatrix}$$

The next launch of the 10 sets of launch vehicles (the current time: 0 hour) were as follows.

$$T = [15\ \ 16\ \ 3\ \ 5\ \ 18\ \ 10\ \ 20\ \ 21\ \ 18\ \ 17]$$

The time launch vehicles spent for loading is LT (load time) = 1 hour.

All the integrated-loading-and-unloading vehicles are at one place, distance from each launch vehicle to be loaded is as follows.

$$DD = [10\ \ 11\ \ 11\ \ 7\ \ 8\ \ 6\ \ 7\ \ 11\ \ 13\ \ 8]$$

The integrated-loading-and-unloading vehicles has a speed of V = 60 km / h (1 km / min).The scale of the population is 20, evolve 200 generations. We can see from table 1 that only when there are more than 3 integrated-loading-and-unloading missile vehicles that can meet the needs. And the best quantity of vehicles is 4.

This result tells us that with this algorithm we can get the optimum accurate solution, especially when there are a number of missile type, location, launch time and number, It can get the optimization number of the integrated-loading-and-unloading vehicles that fulfill the time need, and achieve the aim of optimizing the allocation and saving cost.

Table 1. The Simulation Results

vehicles number	2	3	4	5	6
hours	24	23	20	18.1	18

5. Conclusion

To adapt the real-time-loading accurate support need of precise-navigation weapons before launch, it is necessary to study the quantity determine method of loading vehicles. Aiming at decreasing cost, fulfilling loading tasks, and improving support efficiency, we investigated the problem of optimizing configuration of the quantity of integrated-loading-and-unloading missile vehicles. Compared with the traditional experience method, the algorithm in this paper can determine the number of integrated-loading-and-unloading missile vehicles accurately, which not only can the accurate support need of battlefield, but also can get an optimum balance between cost and loading times based on meeting the different launch times of missiles every vehicle.

References

[1] Tang Huifeng. Research on Scheme Optimization of Equipment Support Ammunition Loading. *Ordnance Engineering College*. 2005; 31-42.
[2] Rong Xiaojian. Study on Combination of Ammunition Packing and Optimization Algorithms. *Ordnance Engineering College*. 2007: 35-41.
[3] Chen Yu-bo,Yu Yong-li, Nie Cheng-lon, etc. Research on Ammo Supply Model for Complex Mission. *Fire Control and Command Control*. 2006; 31(10): 23-25.
[4] Wang Xiaoping, Cao Liming. *Genetic Algorithm-Its Theory, Application and Software Realization*. Xi'an Jiaotong University press. 2002; 01:12-15.
[5] Nirwan Ansari, Edwin Hou. *Computional Intelligence for optimization*. Qinghua University press, Beijing. 1999;12: 49-52.