Optimal Search Plan Model OnThe Sea

Luyao Zhu

China University of Political Science and Law, No. 27, Fuxue Street, Changping District,Beijing, China

email:luyao.z@hotmail.com

Keywords: Maximum Flow Problem, BFS, Genetic Algorithm Model

Abstract. In this passage, we intend to determinate the specific searching plan for lost aircraft on the basis of big data application. First, we cope with Maximum Flow Problem by BFS, in order to the determination of cruise route. Then, in the determination of numbers of search planes and warships along each path, we can use the Genetic Algorithm Model.

Introduction

With development of computer science and statistics, *Big Data* is no longer a new word to us all. In this passage, we are focusing on figuring out the optimal searching plan for lost plane on the basis of big data application. It is not a difficulty to determine the best searching region, but how to find out the optimal searching route and allocate the searching power in the searching region is very troublesome since we have so much information to deal with during the searching period. Now we are going to make search and rescue plans, which may be related to our courses of Operational Research, Computer Science, Topology, Computer Simulation and so on. First, we'll attempt to find a best searching route which allows us to find the lost aircraft in the least time at the lowest cost. Then, we'll determine the quantity of search planes and warships along each path. The searching stage can be divided into 2 sections according to spatial location, which are searching above and on the sea and searching under water. Here we are focusing on the searching above and on the sea.

Notations

Notations in Model 1
1) *nd* is the number of nodes.
2) *i* or *j* is the node, $0<i,j<nd$.
3) *e(i,j)* is the path from *i* to *j*.
4) *le(i,j)* is the value of the path *e(i,j)*.

Notations in Model 2
1) *H* is the area of the objective searched-path.
2) *Y* is maximum number of available search warships.
3) *Z* is maximum number of available search planes.
4) A_i^w is the search area of each search warship, $i=1{:}Y$.
5) A_j^p is the search area of each search plane, $j=1{:}Z$.
6) T_j^L is the endurance of each search plane, $j=1{:}Z$.
7) T_i^w is the time spent by each search warship on searching, $i=1{:}Y$.
8) T_j^p is the time spent by each search plane on searching, $j=1{:}Z$.
9) $\bar{T}_i^w$ is the time spent by each search warship on arriving at the objective path to be searched, $i=1{:}Y$.
10) $\bar{T}_j^p$ is the time spent by each search plane on arriving at and depart from the objective path to be searched, $j=1{:}Z$.
11) *T* is the time spent in the whole searching activity.
12) $y_i = \begin{cases} 1, & \text{if search warship i participates in the serching activity.} \\ 0, & \text{if search warship i doesn't participates in the searching activity.} \end{cases}$

13) $z_j = \begin{cases} 1, & \textit{if search plane j participates in the serching activity.} \\ 0, & \textit{if search plane j doesn't participates in the searching activity.} \end{cases}$

14) $Q_{cd} = le(c, d)$ is the maximum value of path $e(c, d)$ determined in section 2.4.

Model 1: Maximum Flow Problem in Determination of Cruise Route

In this section, we have to figure out some dispersive objective searched-areas in searching region and make out the optimal cruise route. This is related to Maximum Flow Problem. Here, we have to make some assumptions.

Assumptions.(1)Assume that the number of the objective searched-areas is limited and we have made sure location and features of them in advance.

(2)Assume that the size scattered islands in the searching regions is small enough so that we mustn't consider how to bypass them in our searching process.

(3)Assume that the nodes represent the objective searched-areas.

Maximum Flow Problem. In order to improve the universality of our whole model, we will relax the assumption, which induce us to find a better solution to the problem. In fact, the new problem based on the new situation is just the Maximum Flow Problem. In optimization theory, maximum flow problems involve finding a feasible flow through a single-source, single-sink flow network that is maximal.

The maximum flow problem can be seen as a special case of more complex network flow problems, such as the circulation problem. The maximum value of an s-t flow (i.e., flow from source s to sink t) is equal to the minimum capacity of an s-t cut (i.e., cut severing s from t) in the network, as stated in the max-flow min-cut theorem.[1]

Basic Concept of the Maximum Flow Problem.1). State space is typically modeled as a directed graph.

2). The flow is a maximum flow, if and only if there's no augmenting path in the residual network.

3). Minimum cut is the minimal cut in all the ones. The minimum cut is not unique, while the value of it is the only one.

4). The flux of the maximum flow equals the capacity of the minimum cut.

In effect, the principle of the algorithm established in our paper is Ford–Fulkerson algorithm, which means as long as there is an open path through the residual graph, send the minimum of the residual capacities on the path. The Ford-Fulkerson algorithm works only if all weights are integers. Otherwise it is possible that the Ford–Fulkerson algorithm will not converge to the maximum value.

Detailed Process of the Solution.1). Find a path from source node to sink nodes via BFS(Breadth- First Search), which is called augmenting path.

2). If we cannot find such a path stated in step 1), the network solved in the last iteration is the maximum flow. Else, do as follows.

3). Find the minimum from the augmenting path values.

4). Construct a new network by subtracting the minimum value in step 3) from all the augmenting paths.

5). Construct the maximum flow network with the minimum value obtained from step 3), which is included in the initial network.

6). Do as step 1) in the new network.

Application of the Solution. Now we are going to translate the problem of determination of cruise route into Maximum Flow problem.

Firstly, obtaining information above, terrain and environment of the different search area, in order to gain the allocation of rescue resources and search order for distinctive area according to the varying degree of searching difficulty, we decides the area into six districts shown below. We assume that the following figure is just the searching region, with a radius of 87.18 *n miles.*

[1]http://en.wikipedia.org/wiki/Maximum_flow_problem#cite_note-9

$$b=\begin{bmatrix} 1 & 2 & 16 \\ 1 & 4 & 13 \\ 2 & 3 & 12 \\ 2 & 4 & 10 \\ 3 & 4 & 9 \\ 3 & 6 & 20 \\ 4 & 2 & 4 \\ 4 & 5 & 14 \\ 5 & 3 & 7 \\ 5 & 6 & 4 \end{bmatrix}$$
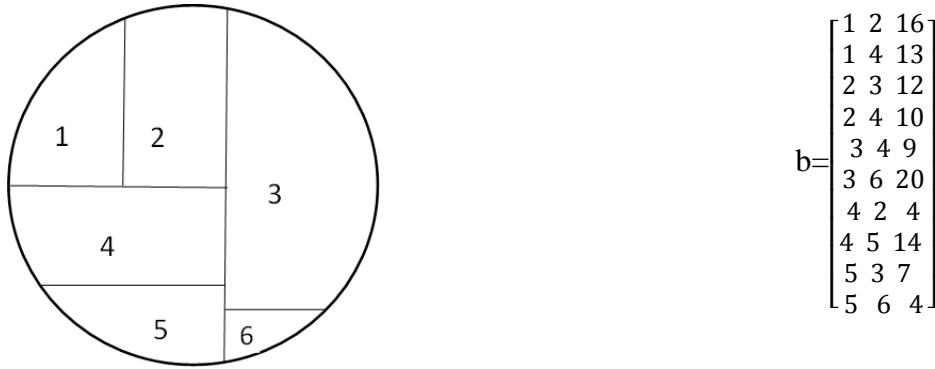
Fig. 1

Next, we make a stipulation: the maximal area we have traverse before we arrive at region j: the maximal value of path *le(i, j)* is equal to 100:1 and region i is corresponded to node i area in Network Topology Mode.

Then, the connection between different regions reflected by the figure below can be expressed in matrix, which is b. b is stated above. The first column is corresponded to start nodes. The second column is corresponded to sink nodes and the third column is corresponded to maximum value of path *le(i, j)*.
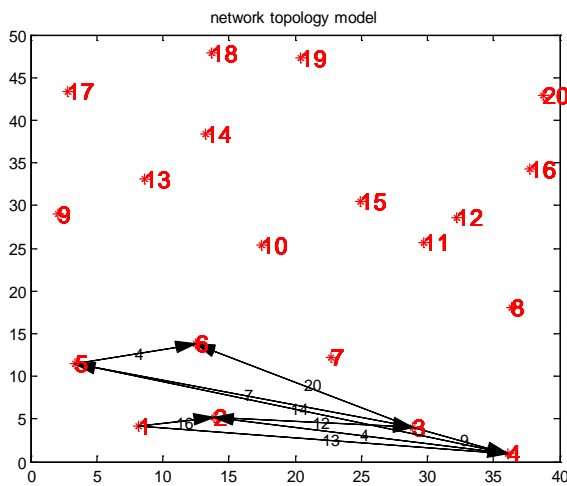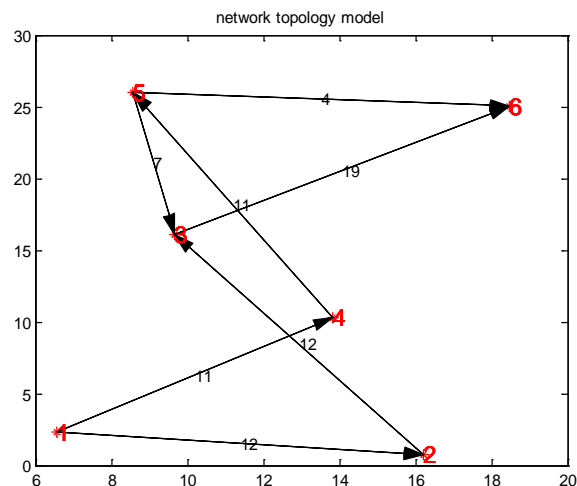

Fig. 2 Initial Network Topology of MF   Fig. 3 Ultimate Network Topology of MF

If we transform the connections into network model, it can be expressed as Figure 3.In this way, we translate the determination problem of cruise route into Maximum Flow problem. The result shows in Figure 4.Here are the results of our simulation.

The result shows that, in the searching region above, there are 6 objective searched-areas. The capacity of the path is as Figure 4, which can be transformed into the allocation of search planes warships along the path. The optimal cruise is as follows.

1) Path *e(1,2)*: allocate 12 search planes or warships along it.
2) Path *e(1,4)* : allocate 11 search planes or warships along it.
3) Path *e(2,3)*: after searching along path *e(1,2)*, allocate 12 search planes or warships along it.
4) Path *e(4,5)*: after searching along path *e(1,4)*, allocate 11 search planes or warships along it.
5) Path *e(5,3)*: after searching along path *e(4,5)*, allocate 7 from 11 search planes or warships along it.
6) Node3: 12 search planes or warships from node2 and 7 search planes or warships from node5 gather at node3.
7) Path *e(3,6)*: allocate all the 19 search planes or warships gathering at node3 along it.
8) Path *e(5,6)*: after searching along path *e(4,5)*, allocate 4 from 11 search planes or warships along it.

9) Node6: 19 search planes or warships from node3 and 4 search planes or warships from node5 gather at node6.

In real searching process, we need to number all the nodes and input the true nodes matrix. Since the matrix depends on the situations, here we use a simulate matrix to verify the feasibility of BFS in cruise route planning. If we would like to use this algorithm program in real searching process, we need to change the b matrix into true matrix in it. Note that in this section, *le(i,j)* obtained in the ultimate solution is the sum of search planes and warships.

Model 2: Genetic Algorithm in Determination of Numbers of Search Planes and Warships along Each Path e(i,j)

Although we have determined the maximal sum of search planes and warships along each searching path in order to complete the searching task, we don't know the exact number of each search power in the path. So we are going to solve this problem in this section.

Assumptions.(1)Assume that there are several available warships and search planes in the neighborhood of the objective path to be searched, usually a belt area. At least the sum of them is not less than *le(i,j)*.

(2)Assume that we ignore the time spent on fueling the plane at base.

(3)Assume that the time taken to arrive to and depart from the objective path to be searched is equal.

(4)Assume that each planes can be used more than once in one searching activity along the path.

The Establishment of Global Optimization Model. In order to cover the whole objective path to be searched, we need to satisfy the following equation.

$$\sum_{i=1}^{Y} T_i^w A_i^W y_i + \sum_{j=1}^{Z} T_j^p A_j^P z_j = H \tag{1}$$

That is ,

$$\sum_{i=1}^{Y} (T - \bar{T}_i^w) A_i^W y_i + \sum_{j=1}^{Z} (T - \frac{T}{T_j^L} \bar{T}_j^p) A_j^P z_j = H \tag{2}$$

In this way, we can solve *T*.

We can establish a Global Optimization Model as follows.

$$\begin{cases} global\ minT = \dfrac{\sum_{i=1}^{Y} \bar{T}_i^w A_i^W y_i + H}{\sum_{i=1}^{Y} A_i^W y_i + \sum_{j=1}^{Z} (1 - \frac{1}{T_j^L} \bar{T}_j^p) A_j^P z_j} \\ \quad s.t.\ \sum_{i=1}^{Y} y_i + \sum_{j=1}^{Z} z_j = Q_{cd} \end{cases} \tag{3}$$

The goal of the GOM is to solve the minimum *T* spent in the whole searching activity. We add the constraint of the sum of search and rescue power to solve the minimal *T* while using different numbers of search power, so that we can synthesizes the search cost and *T*, and work out a searching plan at the lowest cost in the shortest time.

The Solution of Global Optimization Model: The Application of GA. In the field of artificial intelligence, a genetic algorithm (GA) is a search heuristic that mimics the process of natural selection. This heuristic (also sometimes called a metaheuristic) is routinely used to generate useful solutions to optimization and search problems.[2] Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

The Genetic Algorithm has an advantage in the precision of seeking solutions, the efficiency of computing and the avoidance of gaining local optimal solutions. So we use GA to solve the GOM in this section.Here are the results of our simulation.

---

[2]Mitchell 1996, p. 2.

Table 1.search plane

| Type	Length (m)	Diameter (m)	Range (km)	Cruising speed(km/h)	Maximum speed(km/h)	Maximum flying height(m)	Endurance (h)	Radar	Radar range(km)	Satellite
EC-135	12.16	1.86	735	253	278	3000	3.5	2	20	0
EC-145	13.03	1.96	680	246	268	5240	4.5	1	200	1
EC-155	14.4	2.16	830	280	315	4572	3	2	20	1
EC-225	16.79	2.4	857	260.5	275.5	5900	4.5	1	200	1
AW101	22.81	3.36	1389	285	309	4575	4	1	400	1
AW609	13.3	1.12	1390	465	510	7620	3.7	1	345	0
AW139	16.65	2.36	568	290	310	3200	3.9	1	300	0
AS365	13.68	1.1	897	260	296	4300	4	1	500	0
NH90	19.4	3.2	1100	285	310	6000	4.8	1	300	1
MH-60G	19.76	2.36	600	331	361	5790	4.87	3	600	1
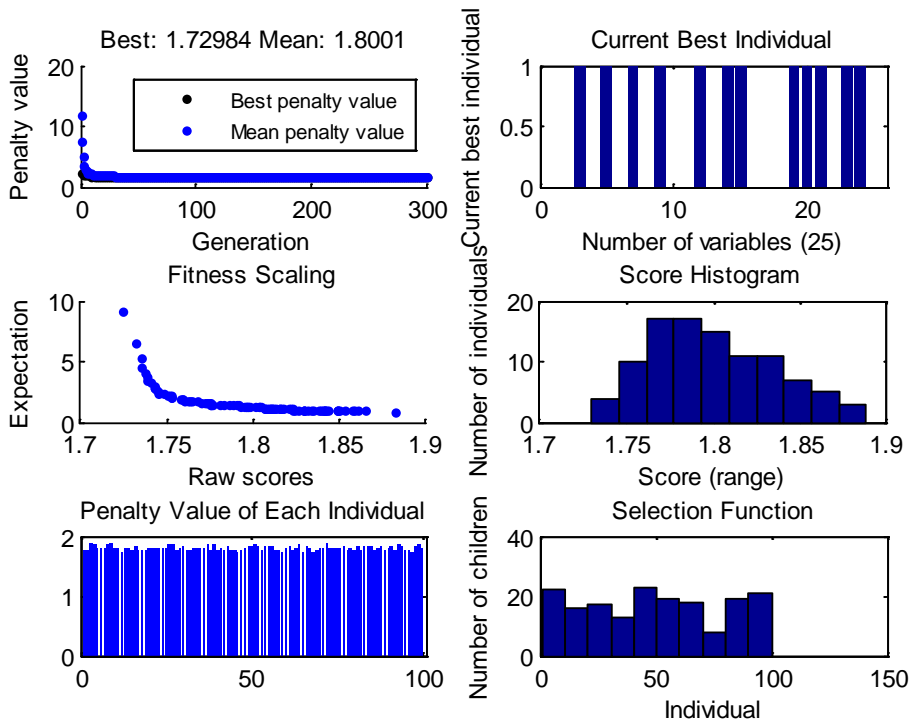
Fig. 4 Results of GA

The result shows that for path $e(1,2)$, we assume that the area of it is around 1200 $n\ mile^2$, maximum number of available search warships Y=10 and that maximum number of available search planes Z=15. The capacity of the search warships is stated in the table provided in the Appendix 1. According to Figure 5, minimal time spent on path $e(1,2)$ is 1.729 h. The optimal solution is that all the 4 warships are appointed to carry on the searching task along the path $e(1,2)$ and 8 search planes numbered are assigned to search along the path.

So the allocation of the search power in other paths can also be solved in this way. Note that we first write up a function named "li4fun.m". Then we input "gatool" in command window in Matlab 2012b, which can open up the GUI of GA. Next input @ li4fun in Fitness function window; input 25 in Number of variables window; set generations as 300, fitness limit as 0.001, stall generations as 100 in Stopping criteria options group; select Best fitness, Best individual, Expectation, Score diversity, Scores and Selection in Plot function windows. Finally, set A=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1], b=12, Lower=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0], Upper=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1] and Integer variable indices as [1:25]. Now click the Start button, and we will get the result of GA.

Conclusion

The article builds a generic mathematical model that could help to made a useful plan to search for a lost plane that have crashed in open water on the basis of big data through stimulation. In the process, we use different models and find a new rescue method. Firstly, as to the determination of cruise route, we choose Maximum Flow Model as to solve the problem. Secondly, in the determination of numbers of search plans and warships along each path, we use the Genetic Algorithm Model. Though the algorithm above, we decided to offer ten warships and fifteen planes in one of the search paths, with search range around 1200 $n\ mile^2$. Besides, the minimal time we conclude to spend on the path is 1.729 *h.*

Appendix

Table2. Search Warships (right)

No.	Initial distance(n mile)	Max speed (n mile)	Range² (nmile²/h)
a	0	8	9
b	21	10	12
c	22	33	50
d	25	12	24
e	26	31	56
f	69	12	21
g	75	21	42
h	77	17	25

No.	Initial distance (n mile)	Max speed (n mile)	Range² (n mile²/h)
i	87	16	21
j	88	13	24
k	92	15	19
l	93	16	27
m	95	21	47
n	97	22	58
o	99	23	62

References

[1] Liu, Y.,and R.H. Weisberg (2011) "A review of self-organizing map applications in meteorology and oceanography." In: *Self-Organizing Maps-Applications and Novel Algorithm Design*, 253-272.

[2] Illustration is prepared using free software: Mirkes, Evgeny M.; Principal Component Analysis and Self-Organizing Maps: applet, University of Leicester, 2011.

[3]Schrijver, A. (2002). "On the history of the transportation and maximum flow problems" *Mathematical Programming* 91 (3): 437–445.doi:10.1007/s101070100259.

[4]Kelner, J. A.; Lee, Y. T.; Orecchia, L.; Sidford, A. (2014). "An Almost-Linear-Time Algorithm for Approximate Max Flow in Undirected Graphs, and its Multicommodity Generalizations" .*Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*. p. 217.