Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 807925, 10 pages
http://dx.doi.org/10.1155/2015/807925

Research Article

Sequence and Direction Planning of Multiobjective Attack in Virtual Navigation Based on Variable Granularity Optimization Method

Chengwei Ruan, Lei Yu, Zhongliang Zhou, and Jinfu Wang

Aeronautics and Astronautics Engineering College, Air Force Engineering University, Xi'an 710038, China

Correspondence should be addressed to Chengwei Ruan; rcwrff@126.com

Received 22 April 2015; Accepted 10 September 2015

Academic Editor: Konstantinos Karamanos

In multiobjective air attacking tasks, it is essential to find the optimal combat preplanning for the attacking flight. This paper solves the planning problem by decomposing it into two subproblems: attacking sequence planning and attacking direction planning. According to this decomposition, we propose the VGHPSO (Variable Granularity Hybrid Particle Swarm Optimization) method. VGHPSO employs the Particle Swarm Optimization, a metaheuristic global optimization method, to solve the planning problem in multiple granularities, including optimizing the high-level attacking sequence and optimizing the low-level attacking directions for engagements. Furthermore, VGHPSO utilizes infeasible individuals in the swarm in order to enhance the capability of searching in the boundary of the feasible solution space. Simulation results show that the proposed model is feasible in the combat, and the VGHPSO method is efficient to complete the preplanning process.

1. Introduction

The traditional multi-target-attack can be presented as a unique attack style that a single flight fighter or multiple flight fighters launch and guide weapons towards multiple targets at the same time. With the development of many advanced technologies, the air combat scenario becomes more and more complex, the needed condition to attack different targets at the same time is hard to satisfy, and then the sequential attack style is born to make the air combat more reasonable.

The multi-target-attack in this paper is defined as an attack process that the multiple targets which are located in distributed spaces are attacked in sequence. Here, the attack sequence is of a great importance according to different relative situations brought by different targets and the reasonable attack sequence may make the task easier and more effective to complete. The other important point in this paper is about the transform time during the mentioned multi-target-attack. It is not avoidable to research on how to enter the attack area of next target after completing the last attack mission. The concept of attack direction is given to describe the transform style during mission process and a proper attack direction can shorten the total time of executing the attack task and keep our flight fighter safer.

Current researches on multi-target-attack problem mainly focused on the algorithms of calculating the target threat [1, 2], methods of target assignment [3–5], tactical decision [6, 7], and so on. Only few references are concerned about the transform time during the attack task, and the common researches on sequential attack pay much attention to the optimization algorithms [8]; the systematic researches from a broader tactical view still need to be focused on.

In this study, a new sequential attack method based on the classic multi-target-attack problem is put forward which can make the abstract route planning problem more concrete, and the whole problem has been transformed into a multiobjective optimization problem based on virtual navigation planning. The classic weapon-target-assignment (WTA) problem [9] has been changed to a PSO (Particle Swarm Optimization) problem [10] in solution space. The classic complicated route planning problem has been decomposed into a hierarchical problem which pay more attentions to the optimizing of attack sequence and attack direction. In the process of optimization, the attack sequence and the attack direction are optimized with relative setting granularities and a surviving policy of noninferior solutions is given to make sure that the best solutions in the Pareto boundary [11–14] will not be neglected.

The paper is organized as follows: Section 1 is introduction. Section 2 presents the basic sequential attack model and its simplified method. Section 3 describes the modified swarm optimization algorithm with a new surviving policy. Section 4 exhibits the simulation results and Section 5 is conclusion.

2. Background-Basic Scenario

2.1. The Analysis of Target Threatening Area. In modern combat scenario, the flight fighter may have many advanced sensor and weapon systems which can bring much more threats [15]. And in this paper, we define the target's threat as the threat of target's comprehensive ability and the threat of target's destroy ability.

Figure 1 describes the traditional 1 versus 1 air combat situation, A is our flight fighter and T is the target, v_a and v_t are the velocities, R is the relative distance between A and T, q is the target's approaching angle, and φ is the target's azimuth angle. In this paper, we suppose that the attitude angle exposed to the target's radar is equal to the target's azimuth angle on condition that A and T are in the same horizontal plane.

In air combat, the greatest threat to our flight fighter comes from the target's detection threat and the target's weapon threat. Thus we need to choose an appropriate approaching method to avoid the threatening area in order to increase our task success rate. So it is necessary to analyze the attack area and the exposure area.

Attack Area. We define the attack area in this content as a special air-to-air missile attack area around the target and only in this area our flight fighter can get a probability within a given threshold to destroy the target after launching a proper missile. The destroy probability may decrease rapidly to zero when launching a missile outside this area [16]. The real-time calculating method to get the attack area is hard to satisfy the needs of the modern combat, so we apply the data fitting method to acquire the approximate target's attack area [17]. Suppose our member carries the middle-range-missile; then the far and near boundary of the attack area around the target can be calculated by the following formulas:

$$D_{\max} = f\left(H, V_0, V_T, \dot{V}_T, V_r, \dot{V}_r, \eta_y, \omega, q_T, t_0\right),$$
$$D_{\min} = f\left(H, V_0, V_T, \dot{V}_T, V_r, \dot{V}_r, \eta_y, \omega, q_T, t_0\right). \quad (1)$$

where $D_{\max}$, $D_{\min}$ are far and near boundary of the attack area around the target, H: is attack height, V_0: is missile launching velocity, V_T: is target velocity, $\dot{V}_T$: is target accelerated velocity, V_r: is target angular velocity, $\dot{V}_r$: is target accelerated angular velocity, n_y: is maneuver overload, ω: is maximum missile off-boresight launch angle, q_T: is target approaching angle, and t_0: is missile working time.

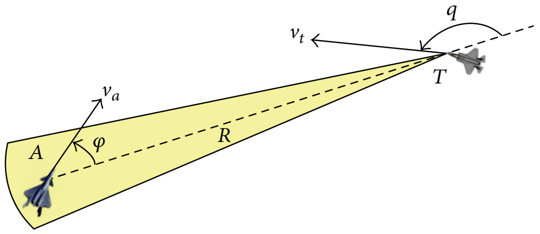


FIGURE 1: The relative situation of friend and foe.

Then we can get the whole attack area around the target in polar coordinate:

$$Z_A = \{(q_T, d, h) \mid q_T \in [-\pi, \pi], h \in [H_{\min}, H_{\max}], d(q_T, h) \in [D_{\min}(q_T, h), D_{\max}(q_T, h)]\}, \quad (2)$$

where q_T is the target's approaching angle, d is the distance of the target, and h is the relative height to the target, where $[H_{\min}, H_{\max}]$ means the attack range in height of missile.

In this study, we suppose that our member has the same attack area with the targets, and both of them have nearly the same comprehensive combat capability. If there was a great gap between these two comprehensive combat capabilities, then we can describe the combat as an asymmetric combat, and the model is not efficient. According to current model, we need to find a proper route planning project to get a better attack situation which can be useful to avoid the target's attack and to make our member safer.

Exposure Area. It means an area we exposed to target's radar. The scanning range of airborne radar is about ±60°. Then we can also get the exposure area as follows:

$$Z_E = \{(q_T, d, h) \mid q_T \in [-\theta_R, \theta_R], h \in [H_{\min}, H_{\max}], d(q_T, h) \in [0, R_R(q_T, h)]\}, \quad (3)$$

where θ_R is the scanning angle of target's radar, R_R is the maximum detection distance, and $[H_{\min}, H_{\max}]$ means the detection range in height of radar.

2.2. The Basic Description of the Model. Sequential attack [18] problem can be regarded as a multiobjective optimization problem, and once the attack sequence and the attack direction are assured, the whole attack logic is born. In order to analyze the multi-target-attack process in detail, we focus on the situation with single target for the first step. Before bringing in the model, we should give some definitions about the relative problems.

Definition. (i) *Minimum safety distance*: this distance can be regarded as the maximum attack distance of the enemy's. When entering this area, our member may get the probability of being attacked by the enemies. (ii) *Attack preparing area*: this area is defined between the minimum safety distance boundary and the far attack area boundary where our flight

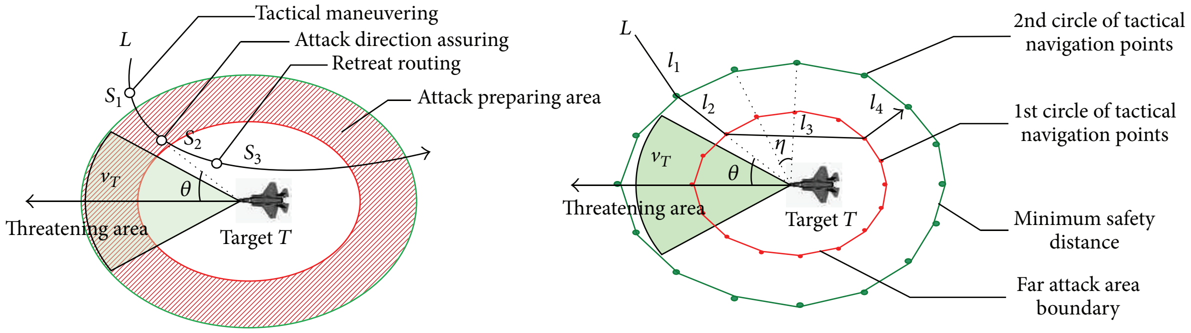


Figure 2: The simplified model of sequential attack.

fighter should complete a series of actions to complete its attack task.

From the left side of Figure 2, we can see the whole process of single-target-attack.

L is the virtual navigation route across the attack preparing area (an approximate area in this paper), and through the whole cross process, the flight fighter may deal with several key stages; they are listed as follows.

(a) Tactical Maneuvering Stage. This stage may decide which targets to attack and which routes to choose to enter the minimum safety distance. This stage may last till the missile launches.

(b) Attack Direction Assuring Stage. This stage is used to avoid the threaten area of enemies and find a more effective way to increase the success rate.

(c) Retreat Routing Stage. This stage is very important to offer an input for the next attack process and it is also useful to find an safe retreat route to get away from the last threaten area.

The three stages can describe the whole attack process and in order to build a mathematical model of the different stages, we simplified the model as the right side of Figure 2.

The attack preparing area is described by two circles of tactical navigation points. The inner circle is the boundary of far attack area and the outer circle is the boundary of minimum safety distance. The missile can be launched towards the target when the far attack area boundary is reached, so we center the target and take sample points in the far attack area boundary every η angle; then we can get the 1st circle tactical navigation points. We define the 1st circle tactical points of target t as $RP_1^t = \{P_1^t, \ldots, P_{N_1}^t\}$. In order to enter the 1st circle tactical points, we cannot avoid the minimum safety distance boundary which can be decided by the threatening degree of the target as well as the relative combat situations. So we also center the target and take sample points in the minimum safety distance boundary every η $(\eta = 2\pi/n\ (n = 1, 2, \ldots))$ angle; then we can get the 2nd circle tactical points. We define the 2nd circle tactical points of target t as $RP_2^t = \{Q_1^t, \ldots, Q_{N_2}^t\}$. Then the problem can be simplified as a route optimization problem which can be described as how to find a best route through the two circles with shortest time and least threats. Then the transformation between two targets can be completed.

If target i would be attacked before target j, then the partial order exists and we make it as $Q_1^i \succ Q_1^j$. The solution space of virtual navigation can be described as

$$PS = \{P_1^1, \ldots, P_{N_1}^1, Q_1^1, \ldots, Q_{N_2}^1, \ldots, P_1^T, \ldots, P_{N_1}^T, Q_1^T, \ldots, Q_{N_2}^T\}. \tag{4}$$

From the right side of Figure 2, we can see that L means the attack route through two circles of navigation points and θ is the angle between the velocity of target and the attack route and it can describe the attack direction towards the target.

From the analysis above, we can conclude the problem in two steps.

Step 1. Choose four tactical points from the two circles, and decide the attack sequence according to every single target in different space. We describe it as $Q_i^t \rightarrow P_m^t \rightarrow P_n^t \rightarrow Q_o^t$ $(m \neq n)$.

Step 2. Connect the four tactical points in sequence, and a new virtual navigation route is born. We can describe it as a tuple:

$$L = \langle N, S, Tg, Pr, E \rangle, \tag{5}$$

where N means the number of navigation route fragments which connect the two kinds of navigation points in different circles. S is the initial position of our member and it is also the first navigation point in the attack route. Tg is a set of serial numbers of targets which records the attack sequence. Pr contains all the chosen tactical points and their orders. E ends the navigation.

3. Modeling and Algorithm Optimizing

3.1. Analysis of Sequential Attack and the Attack Direction. The single-target-attack model is given in Section 2, and the multi-target-attack model is set up in this chapter. From Figure 2, we can get the general view of sequential attack in multiple targets' situation.

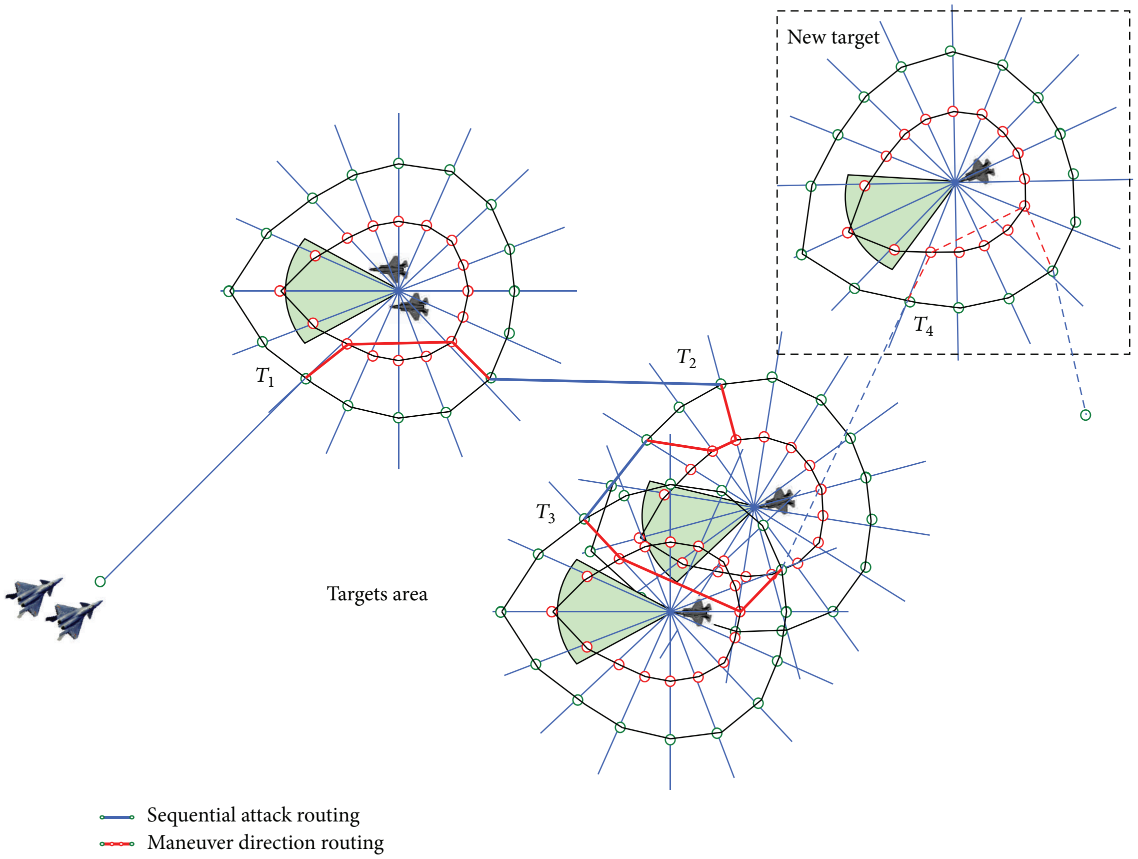


FIGURE 3: The sketch map of multiobjective sequential attack.

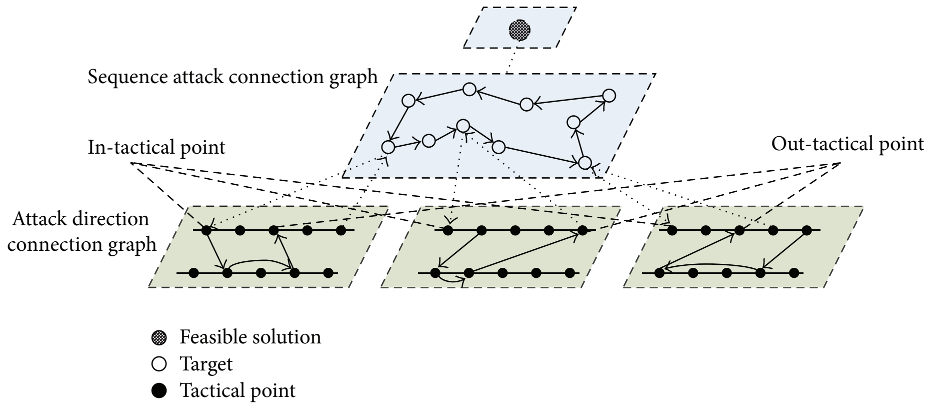


FIGURE 4: Mission decomposition in variable granularities.

In Figure 3, the blue line means the attack sequence and the red line means the attack direction. We can easily find that once the tactical points are decided, the whole attack plan including attack sequence and direction is acquired. From a tactical view of the combat situation, we can solve this problem in three different hierarchies.

In Figure 4, the problem has been divided into three parts which describe the feasible region, sequential attack, and attack direction, respectively. The second layer can be described as a directed graph $g_S = g(T, E_s)$, $T = \{T_1, T_2, T_3, \ldots, T_n\}$, where T is a set of targets and E_s is a set of connection arcs. $e(T_i, T_j) \in E_s$ means there is a partial order between T_i and T_j, and T_i will be attacked first. The third layer can also be described as a connected directed graph $T_i = g_i(P, E_p)$, where P is a set of tactical navigation points and E_p is a set of connection arcs. Then the whole problem can be rebuilt as a variable granularity directed graph $g_M = g(\{g_i(P, E_p)\}, E)$, E means the connection arcs between different granularities.

3.2. Modeling

3.2.1. Targets' Reformation. In current air combat scenario, the targets' flight fighters always appear in formation to execute associated tactical task. Thus we can divide these targets into different groups according to their unique characters in order to simplify the complex combat situation. And first, the group should satisfy the following formula:

$$\bar{l}_{\text{avg}} = \frac{l_{\text{avg}}}{L_{\text{avg}}}, \tag{6}$$

where l_{avg} is the mean distance between two targets in the same group and L_{avg} is the mean distance between two different groups. The minimum value of $\bar{l}_{\text{avg}}$ can be regarded as the important index to choose the proper projects.

We give the minimum distance $L_{\min}$ between groups and maximum distance $l_{\max}$ between targets in the same group in different projects according to the combat needs. When $l \geq l_{\max}$, then the target does not belong to the group. When $l < L_{\min}$, we can decide that the target is in the same group. Thus, we let $L_{\min} = l_{\max} = l_0 = \text{const}$ in order to use a common method for completing the targets' reformation.

The concrete method to reform the target is given as follows:

(a) Set a tactical target's group as

$$G_t = \{e_1, e_2, \ldots, e_j, e_{j+1}\}; \tag{7}$$

$e_1, e_2, \ldots, e_j$ are members in the group and e_{j+1} is the $j + 1$ member.

(b) Set the attribute of the member as

$$e_i = e(x_i, y_i, z_i, v_i, \psi_i, k_i); \tag{8}$$

x_i, y_i, z_i is the space coordinate of the target in geographic coordinate system; v_i is the velocity in geographic coordinate system; ψ_i is the flight direction angle in geographic coordinate system; k_i is the target's types.

(c) Set the constraints to reform the target:

$$\begin{aligned} \text{s.t.} \quad & \left|x_{j+1}(t) - x_i(t)\right| < \Delta x \\ & \left|y_{j+1}(t) - y_i(t)\right| < \Delta y \\ & \left|z_{j+1}(t) - z_i(t)\right| < \Delta z \\ & \left|v_{j+1}(t) - v_i(t)\right| < \Delta v \\ & \left|\psi_{j+1}(t) - \psi_i(t)\right| < \Delta \psi \\ & j + 1 \leq M_{\max},\ i \in (1, 2, \ldots, j); \end{aligned} \tag{9}$$

$\Delta x, \Delta y, \Delta z$ is the threshold to reform the targets in targets' location; Δv is the velocity threshold of target; $\Delta\psi$ is the flight direction angle threshold of target; $M_{\max}$ is the maximum number of the group. Different constraints may bring different consequence, so it all depends on the combat situation.

3.2.2. Attack Sequence Optimization. In air combat scenario, the advantages or disadvantages towards enemies can be judged by four conditions such as aspect, relative distance, velocity, and height [19]. We can get the attack sequence by the formula below:

$$\begin{aligned} f &= \varphi\left[\mu(\theta), \mu(R), \mu(V), \mu(H), W\right], \\ W &= \{w_1, w_2, w_3, w_4\}, \\ \sum_{i=1}^{4} w_i &= 1, \end{aligned} \tag{10}$$

where $\mu(\theta)$, $\mu(R)$, $\mu(V)$, $\mu(H)$ are linguistic variables in which grade values are between 0 and 1; the closer the value of membership grade is to 1, the more advantages it is to attack the target. W is the weight variable set which contains four relative weight variables in order to stress the importance of the linguistic variables. Then the fitness function can be given as follows:

$$F = \max \sum_{i}^{n} f_i. \tag{11}$$

According to different targets, we can get a sequence of f_i $(i = 1, 2, \ldots, T)$, and then the total advantages can be confirmed.

3.2.3. Attack Direction Optimization. In the model we mentioned above, we can easily find that the attack direction can only be confirmed by connecting two tactical navigation points around the target. So once the best tactical points are chosen, the optimal attack direction is given automatically.

3.3. The Evaluating Indexes of the Optimization.

The evaluation function is needed to evaluate the attack route, and in this situation the distance between two different tactical points is far enough with the support of the ability of long-middle-range missile. Thus we set up four main indexes to evaluate the whole attack route which contains the total distance of attack route, the exposure rate, the constraints of attack direction, and the constraints when turning around. Figure 5 describes the detailed definition of each variable.

3.3.1. Total Distance of Attack Route. Consider

$$D_L = \sum_{i=1}^{N} l_i, \tag{12}$$

where l_i is the distance of each navigation fragment.

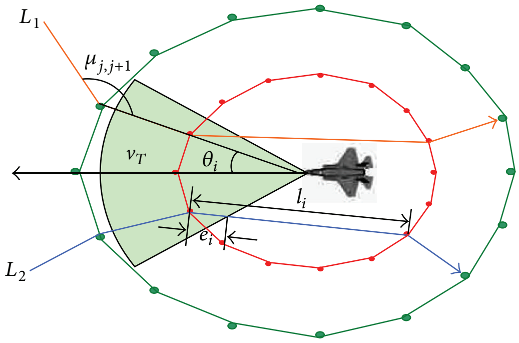


Figure 5: Definition of each variable.

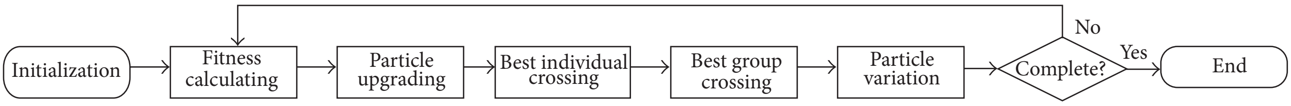


Figure 6: Calculating steps.

3.3.2. Attack Route under Threat. Consider

$$E_L = \sum_{i=1}^{N} e_i, \tag{13}$$

where e_i is the distance under threat (fire threat) of each navigation fragment, the distance under threat is in relationship with the target's attack area, and it can be calculated by analysis of the combat situation of enemies.

3.3.3. Attack Direction Constraints. The attack direction should satisfy the maximum missile off-boresight launch angle, so we give the constraint as

$$\theta_i = f\left(P_I^i, Q_I^i, T_i\right) \le \theta_{\max}, \tag{14}$$

where θ_i describes the attack direction towards target I and $\theta_{\max}$ is the maximum missile off-boresight launch angle.

3.3.4. Constraints When Turning Around. Consider

$$\begin{aligned} &\mu_{j,j+1} = \infty \quad \text{if } l_j + l_{j+1} < r_{\min} \\ &\mu_{j,j+1} \ge \mu_{\min} \quad \text{else} \\ &j = 1, 2, \ldots, N-1, \end{aligned} \tag{15}$$

where $\mu_{j,j+1}$ is the angle between route j and route $j+1$, l_j means the distance of route j, $r_{\min}$ is the minimum turn radius, and $\mu_{\min}$ is the minimum turn angle.

These four indexes we mentioned above can make contributions to building the fitness function as follows:

$$F(L) = w_1 \frac{D_L}{D_M} + w_2 \frac{E_L}{E_M} + g_1(L) + g_2(L), \tag{16}$$

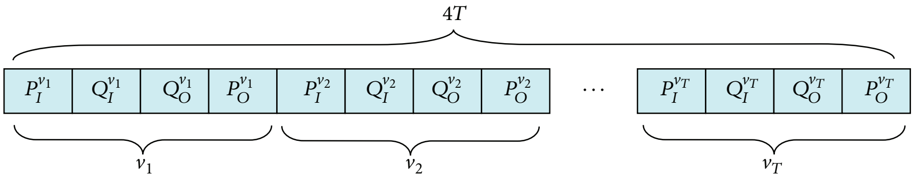
where $F(L)$ is the fitness function and the smaller $F(L)$ gets the better the result will be. If $F(L) \to \infty$, then the value would be meaningless. D_L is the total distance of the attack route, E_L is the distance under threat, D_M and E_M are normalized coefficients, w_1 and w_2 are weight coefficients, and $g_1(L)$ and $g_2(L)$ are correction terms to correct the conditions when the target approaching angle cannot satisfy the need of maximum missile off-boresight launch angle and the turning angle is too small to support our action.

3.4. Calculating Process. According to the problem simplification above, we have changed the complex model into a typical TSP model; thus we use hybrid Particle Swarm Optimization method to solve this problem. The detailed steps can be acquired in Figure 6.

3.4.1. Encoding Method. From Figure 7, we can see that the length of the gene code is $4T + 2$, each target may get 4 codes which can be described by decimalism, and each code ranges from 1 to the maximum number of tactical points. For better initializing the values, we make the two codes in the middle of the gene unequal, and all the code will get a random but not repeated number from 1 to the maximum number of tactical points.

3.4.2. Route Modifying. The two-circle-encode style has forced all the routes to cross through the tactical points, so the situations of route crossing and repeating exist. According to the rule in Figure 8, we can get the new attack route.

The project needs to be accessed again after modifying the virtual navigation. If the accessing result was not better than the old one, we would keep the old policy.



$$\overbrace{\underbrace{P_I^{v_1}\ Q_I^{v_1}\ Q_O^{v_1}\ P_O^{v_1}}_{v_1}\ \underbrace{P_I^{v_2}\ Q_I^{v_2}\ Q_O^{v_2}\ P_O^{v_2}}_{v_2}\ \cdots\ \underbrace{P_I^{v_T}\ Q_I^{v_T}\ Q_O^{v_T}\ P_O^{v_T}}_{v_T}}^{4T}$$

FIGURE 7: The encoding method.

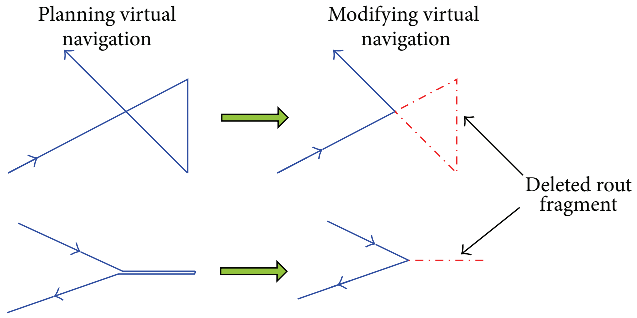


FIGURE 8: The display of route modifying method.

3.4.3. Result Space Expanding Policy. When calculating, the result may not get out of the local optimization, so after each iteration, we get the global optimal value:

$$\begin{aligned} &\mathit{bestRout} \\ &\quad = \left(P_I^{v_1} Q_I^{v_1} Q_O^{v_1} P_O^{v_1} P_I^{v_2} Q_I^{v_2} Q_O^{v_2} P_O^{v_2} \cdots P_I^{v_j} Q_I^{v_j} Q_O^{v_j} P_O^{v_j}\right). \end{aligned} \tag{17}$$

Then we can deal with the existing result as follows and the description of the space expanding policy can be seen in Figure 9:

(a) Rescue the abandoned route when the following condition is satisfied:

$$\left|\mathit{gbest} - \text{fitness}_i\right| < \varepsilon, \quad (i = 1, 2, 3, \ldots, n), \tag{18}$$

where *gbest* is the global optimal value in current time and fitness_i is the abandoned value in the current circulation; when the given condition is satisfied, we rescue the relative Rout_i $(i = 1, 2, 3, \ldots, n)$, and this route will take part in the next circulation.

(b) Get n random points around each best tactical navigation point within the distance $r_i = \theta R_i/2$ $(i = 1, 2)$. (R_i defines the radius of each tactical area.)

(c) Expand the result space with choosing points and get a new result space Ω_{new}.

(d) Calculating $f = \mathit{gbest}/t$, t is the simulation time. If $f < \delta$, then we save the current result.

3.4.4. Calculating Steps. See Algorithm 1.

4. Simulation

4.1. Hardware Condition. CPU is P4 2.8 GHz and RAM is 1.5 G, Matlab 2010.

4.2. The Tactical Scenario. Our flight fighter is flying towards the enemies' formation, and the initial locations of enemies and our member are given in Table 1.

The relative situation can be seen in Figure 10.

4.3. Simulation Result. We suppose the detecting area of enemies is stable at the moment, and we need to find the best attack sequence and attack direction at this moment. We set 3000 loops to optimize the final result, and the result can be seen in Figures 11 and 12.

From Figure 11, our member choose the proper direction to avoid the threaten areas in order to complete the task in a relative short way. The attack sequence can be described as $T_1 \rightarrow T_3 \rightarrow T_2 \rightarrow T_4$, and the attack direction (the direction of enemy velocity is the initial angle 0°) can be decided by the angles given in Table 2.

Figure 12 shows the updating process when choosing the best solution, and after 600 more loops, we get the stable value; the best route we choose can be shown as follows:

$$\mathit{bestRout} = \left(P_I^3 Q_I^3 Q_O^{23} P_O^{22} P_I^{10} Q_I^{15} Q_O^{17} P_O^{18} P_I^{10} Q_I^{15} Q_O^{17} P_O^{18} P_I^2 Q_I^5 Q_O^{19} P_O^{17} P_I^9 Q_I^4 Q_O^{21} P_O^{23}\right). \tag{19}$$

In Figure 12, three different methods are compared with each other. The blue line shows the result when using traditional PSO method without dividing them in different granularities; then it has cost more time to get a stable result because of much more formations of the available tactical points. The red line means the VGHPSO method we mentioned above, it has cost less time to get a stable value, but it is easy to get trapped in the local optimal value. Thus we use VGHPSO* method which means the VGHPSO method with result space expanding policy. We can see the result reflected in cyan line.

In the former chapters, we can know that the numbers of tactical points are depended on the sampling angle. Thus we

Table 1: Initial locations.

Start point	T_1	T_2	T_3	T_4	End point
(400, 700)	(700, 1200)	(1000, 1000)	(1100, 1150)	(1500, 1350)	(1700, 1000)

Table 2: Attack direction display.

Attack sequence	T_1	T_3	T_2	T_4
Approach angle (°)	107.28	160.76	75.43	133.06
Attack angle (°)	80	127.93	29.14	172.93
Retreat angle (°)	166.73	136.73	20.51	160.51

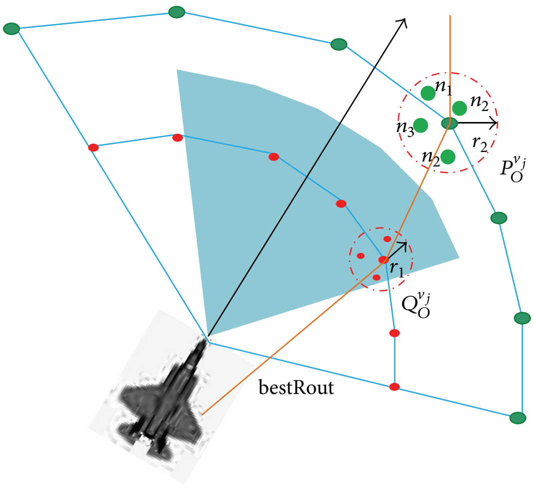


Figure 9: Result space expanding policy.

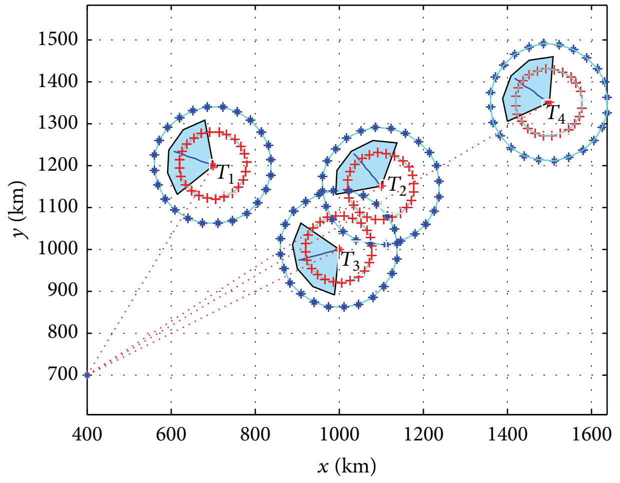


Figure 10: The initial locations sketch map.

do some simulation with different sampling angles, and the results can be seen in Figure 13.

From left side of Figure 13, the blue parts mean the total distances in different sampling angles, and the red parts describe the distances under enemies' threats. Then we can get the ratio as shown in Table 3.

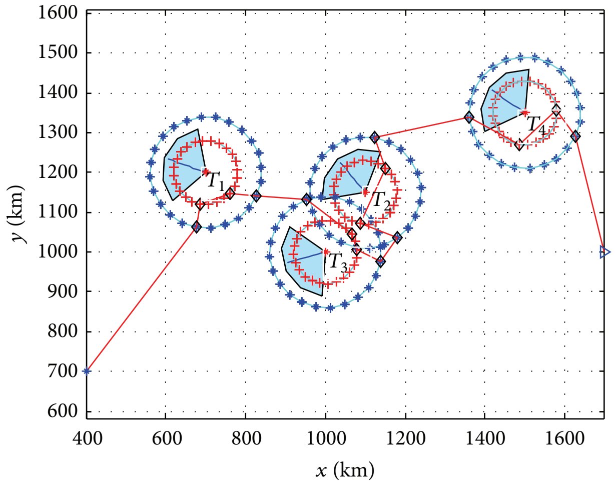


Figure 11: The optimal route planning project.

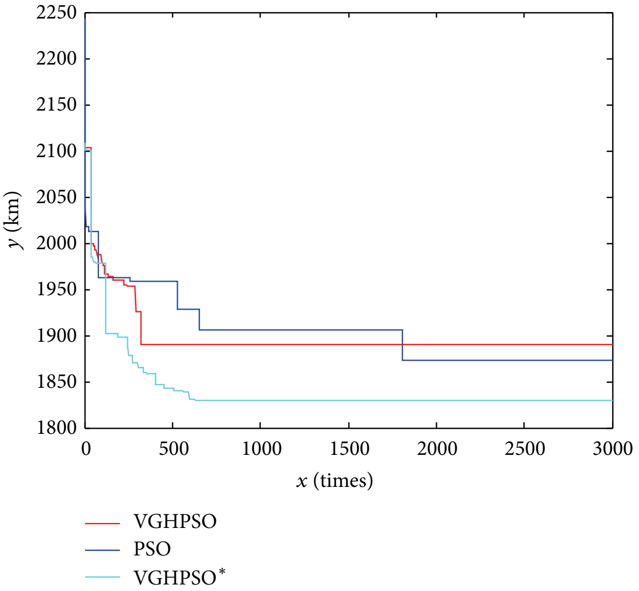


Figure 12: Updating curves in different algorithms.

From Table 3, we can see that when choosing the sampling angle as 15°, we can get a minimum comprehensive index which indicates that we may get a more proper result efficiently with less threats and shorter times.

The usage of the results in this paper can only be regarded as the equal description of attack sequence and attack direction, and the tactical points in virtual navigation are not the real navigation points, but of great importance to simplify the engineering problems.

```
Input: (1) Targets' attributes, including the targets' number $N$, the coordinate $T(x_T, y_T, z_T)$, the velocity $v_T$ and the attack area $W$; (2) Our flight fighters' attributes, including the number of our fighters $M$, the angle $\theta$ to decide the number of tactical navigation points, the coordinate $F(x_F, y_F, z_F)$, and the results space $\Omega$; (3) A result space expanding policy $\rho^*$ to optimize the result.
Suppose Condition: (1) The targets stay flying in a stable velocity and an unchanged direction; (2) The tactical points' formation should obey the constraints $\xi\{sequence, direction\}$.
Output: (1) The global optimal fitness result $gbest$; (2) Best tactical points formation $bestRout$.
Initialize: random $\Omega$, set the initial state
Begin
  For each iteration $i = 1 : n$ // $n$ is the iteration number.
    Cross($F$) $\rightarrow$ $\Omega_{\text{new}}$ // executing the cross process towards the setting codes.
      If $F_i$ $(i = 1, 2, \ldots, \pi/\theta)$ $(F_i \in \Omega_{\text{new}})$ do not obey the constraints $\xi\{sequence, direction\}$
        While ($F_i$ obey the constraints $\xi\{sequence, direction\}$)
          Random $F_i$ // make sure that all the elements in $\Omega_{\text{new}}$ obey the constraints.
        end
          Update ($\Omega_{\text{new}}$)
      end
          Calculating ($fbest_i$)
            If $fbest_i < \min\{fbest_1, fbest_2, \ldots, fbest_{i-1}\}$ Then
            $gbest_i = fbest_i$
          Variate($F$) $\rightarrow$ $\Omega'_{\text{new}}$ // executing the variation process towards the setting codes.
      Cycling the process as Cross method.
      Executing ($\rho^*$) // the saving policy is used to expand the result space to optimize the final result.
  End
            Return ($gbest$, $bestRout$, $bestindex$)
End
```

Algorithm 1

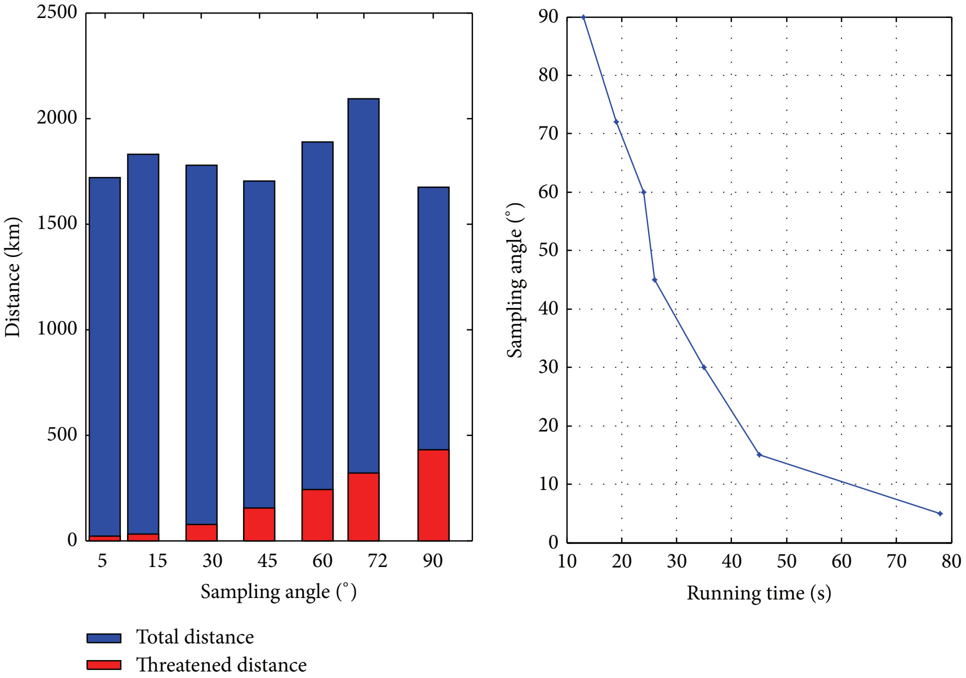


Figure 13: The comprehensive values in different sampling angles.

Table 3: Relative calculating results display.

Sampling angle (°)	5	15	30	45	60	72	90
Threats ratio	0.013	0.017	0.044	0.090	0.129	0.153	0.257
Running times (s)	78	45	35	26	24	19	13
Comprehensive index	1.014	0.765	1.54	2.34	3.096	2.907	3.341

5. Conclusion

The multiobjective sequence attack and attack direction planning methods are given in the paper. We put forward a simplified model to create the tactical points in virtual navigation, and the complex air-to-air combat planning problem has been transformed into a variable granularity TSP problem; then the VGHPSO method is used to solve the given problem. In the end, the result of the simulation shows the efficiency of the method and it also indicates that the model is available.

The model and method we mentioned in the paper are available only in static combat planning which belongs to precombat planning problem. But in real combat scenario, we need a dynamic data management system to optimize the real-time policies. Then we need to pay much attention to the prediction of the future situation according to the historical data [20]. Thus we may get a more proper and more accurate result.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] S. Kang, H. Park, S. Noh et al., "Autonomously deciding countermeasures against threats in electronic warfare settings," in *Proceedings of the International Conference on Complex, Intelligent and Software Intensive Systems (CISIS '09)*, pp. 177–184, IEEE, Fukuoka, Japan, March 2009.

[2] H. Guo, Y. Lv, P. Wang, P. Ren, and L. Zhu, "Target threat assessment of air combat based on intervals and SVR," *Fire Control & Command Control*, vol. 39, no. 8, pp. 17–21, 2014.

[3] A. Salman, I. Ahmad, and S. Al-Madani, "Particle swarm optimization for task assignment problem," *Microprocessors and Microsystems*, vol. 26, no. 8, pp. 363–371, 2002.

[4] P. Y. R. Ma, E. Y. S. Lee, and M. Tsuchiya, "A task allocation model for distributed computing systems," *IEEE Transactions on Computers*, vol. 31, no. 1, pp. 41–47, 1982.

[5] D. Xinghua, F. Yangwang, X. Bingsong, and et al, "Task allocation in cooperative air combat based on multi-agent coalition," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 40, no. 9, pp. 1268–1275, 2014.

[6] X.-G. Gao, Y.-H. Wang, and B. Li, "Decision-making method for the optimal coordination of fire and electronic warfare based on integrative effectiveness," *Systems Engineering and Electronics*, vol. 29, no. 12, pp. 2081–2084, 2007.

[7] H. Guo, H. Xu, D. Liu, and D. Wang, "Air combat decision-making for cooperative multiple target attack based on adaptive hybrid particle swarm algorithm," *Journal of Air Force Engineering University*, vol. 11, no. 2, pp. 16–19, 2010.

[8] K. Horie and B. A. Conway, "Optimal fighter pursuit-evasion maneuvers found via two-sided optimization," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 1, pp. 105–112, 2006.

[9] S. Matlin, "A review of the literature on missile-allocation problem," *Operations Research*, vol. 18, no. 2, pp. 334–373, 1970.

[10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, November-December 1995.

[11] W. A. Crossley, "Optimization for aerospace conceptual design through the use of genetic algorithms," in *Proceedings of the 1st NASA/DoD Workshop on Evolvable Hardware*, pp. 200–207, Pasadena, Calif, USA.

[12] M. Cámara, J. Ortega, and F. de Toro, "A single front genetic algorithm for parallel multi-objective optimization in dynamic environments," *Neurocomputing*, vol. 72, no. 16–18, pp. 3570–3579, 2009.

[13] C.-K. Goh and K. C. Tan, *Evolutionary Multi-objective Optimization in Uncertain Environments: Issues and Algorithms*, Springer, Berlin, Germany, 2009.

[14] Y. Yu and Z.-H. Zhou, "On the usefulness of infeasible solutions in evolutionary search: a theoretical study," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 835–840, IEEE, Hong Kong, June 2008.

[15] J. S. McGrew, J. P. How, B. Williams, and N. Roy, "Air-combat strategy using approximate dynamic programming," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 5, pp. 1641–1654, 2010.

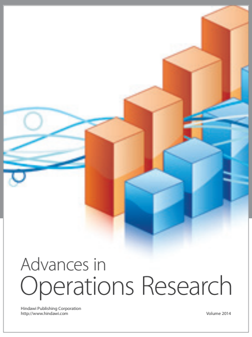
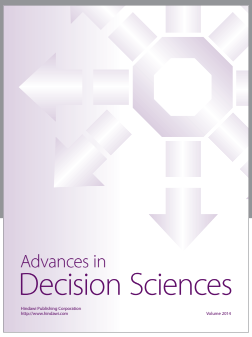
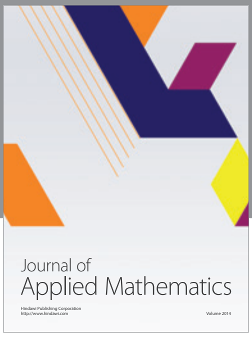
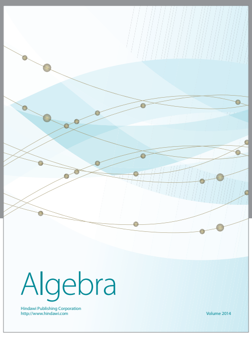
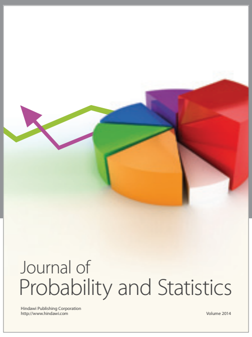
[16] Z. Zhou, *General Aviation Fire Control Theory*, National Defense Industry Press, Beijing, China, 2008.

[17] L. Xiao and J. Huang, "Study on calculation for air-to-air missile attack area based on improved RBF neural network," in *Proceedings of the International Conference on Computer Science and Artificial Intelligence Advances in Artificial Intelligence*, pp. 631–637, Sydney, Australia, December 2012.

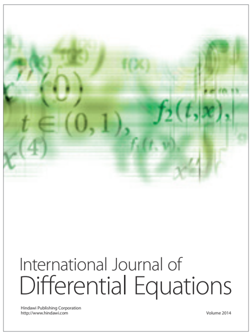
[18] H. Zhang and D. Zhou, "Route planning method in air-to-ground attack," *Electronics Optics & Control*, vol. 1, no. 1, pp. 37–42, 1999.

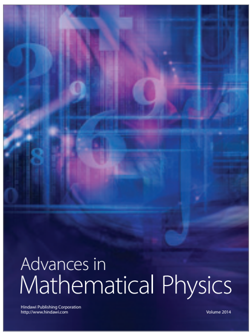
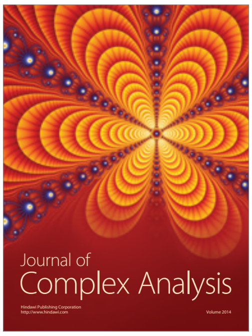
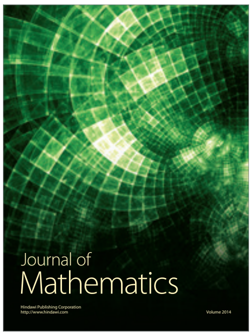
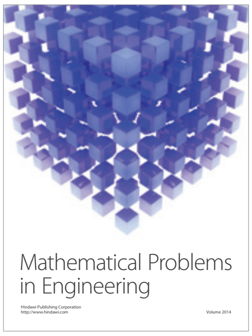
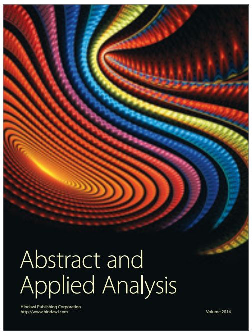
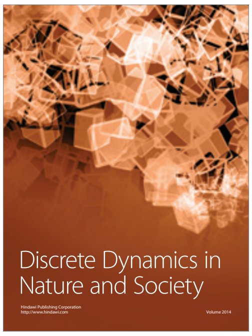
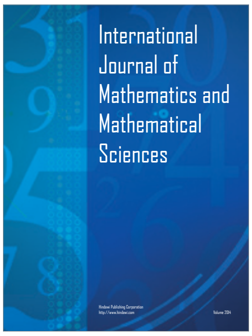
[19] T.-Y. Sun, S.-J. Tsai, Y.-N. Lee, S.-M. Yang, and S.-H. Ting, "The study on intelligent advanced fighter air combat decision support system," in *Proceedings of the IEEE International Conference on Information Reuse and Integration (IRI '06)*, pp. 39–44, IEEE, Waikoloa Village, Hawaii, USA, September 2006.

[20] X. Hai and Y. Lei, "A situation forecast method based on intuitionistic fuzzy reasoning," *Electronics Optics & Control*, vol. 18, no. 4, pp. 33–37, 2011.

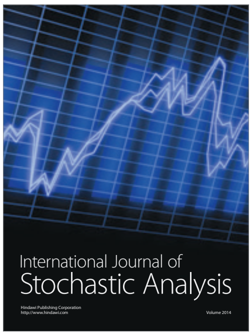

Advances in
Operations Research

Advances in
Decision Sciences

Journal of
Applied Mathematics

Algebra

Journal of
Probability and Statistics

The Scientific
World Journal


International Journal of
Differential Equations

International Journal of
Combinatorics

Advances in
Mathematical Physics

Journal of
Complex Analysis

Journal of
Mathematics

Mathematical Problems
in Engineering

Abstract and
Applied Analysis

Discrete Dynamics in
Nature and Society

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces

International Journal of
Stochastic Analysis

Journal of
Optimization