Abstract—Path planning of air robot is a complicated global optimum problem. Intelligent Water Drops (IWD) algorithm is newly presented under the inspiration of the dynamic of river systems and the actions that water drops do in the rivers, and it is easy to combine with other methods in optimization. In this paper, we propose an improved IWD optimization algorithm for solving the air robot path planning problems in various environments. The water drops can act as an agent in searching the optimal path. The detailed realization procedure for this novel approach is also presented. Series experimental comparison results show the proposed IWD optimization algorithm is more effective and feasible in the air robot path planning than the basic IWD model.

I. INTRODUCTION

AIR robot is one of inevitable trends of the modern aerial equipments which develop in the direction of unmanned attendance and intelligence [1]. Research on air robot directly affects battle effectiveness of the air force and is fatal and fundamental research related to safeness of a nation, also to the civil applications. Path planning is to generate a space path between an initial location and the desired destination that has an optimal or near-optimal performance under specific constraint conditions, and it is an imperative task required in the design of air robot. The flight path planning in a large mission area is a typical large scale optimization problem, a series of algorithms have been proposed to solve this complicated multi-constrained optimization problem, such as the A* algorithm, evolutionary computation [1]. However, those methods can hardly solve the contradiction genetic algorithm [2], ant colony algorithm [4-6], between the global optimization and excessive information.

The intelligent water drops (IWD) algorithm was firstly proposed by S. H. Hamed [7, 8], which mimic the dynamic of river systems and the actions that water drops do in the rivers. The ideas that are taken from natural water drops are used in order to develop artificial water drops. It has been applied successfully to benchmark problems such as the Traveling Salesman Problem (TSP).

Combining the basic model of IWD, a novel path planning approach for air robot has been proposed in this paper. Our approach overcomes the deficiencies of existing path planning algorithms for air robot, which is the contradiction between the global optimization and excessive information. The experimental results illustrate that our approach can generate a feasible optimal path of air robot more quickly than the basic IWD algorithm.

The remainder of this paper is organized as follows. Section II introduces the threat resources and objective function in air robot path planning. Subsequently, the principle of the basic IWD is explained in Section III. Then, in Section IV, we propose a strategy for air robot path planning by using IWD optimization. The simulation results are given in Section V. Our concluding remarks are contained in Section VI.

II. THEREAT RESOURCES MODELLING IN AIR ROBOT PATH PLANNING

A. Environments and path statements

Modeling of the threat sources is the key task in air robot optimal path planning. In order to simplify the air robot path planning problem, the air robot task region can be divided into two-dimensional mesh, thus forming a two-dimensional network diagram connecting the starting point and goal point, which can be shown in Figure 1. In this way, the problem of air robot optimal path planning is the general path optimization problem in essence.

Fig. 1 Typical air robot mission area

In Figure 1, suppose the flight task for air robot is from node B to node A. There are some threatening areas in the task region [3]. Let OA be the x axis, and OB be the y axis, a
coordinate system is then established. We divide OA into m sub-sections, and divide OB and OC into n sub-sections equally. There are (m-1) vertical lines between node B and node A, which can be labelled with \(L_1, L_2, \ldots, L_{m-1}\). The (m-1) vertical lines and the (2n+1) horizontal lines cross-constitute (m-1)\(\times\)(2n+1) nodes. We named these nodes as \(L_1(x_1, y_1), L_2(x_2, y_1), \ldots, L_m(x_m, y_1), \ldots, L_m(x_1, y_{2n+1}), \ldots, L_{m-1}(x_{m-1}, y_{2n+1})\). Where \(L_i(x_i, y_i)\) is the i-th node in the vertical line \(L_i\). In this way, the path from the starting node (A) to the target node (B) can be described as follows:

\[\text{Path} = \{o, L_1(x_1, y_1), L_2(x_2, y_2), \ldots, L_{m-1}(x_{m-1}, y_{(m-1)}), A\} \quad (1)\]

Where \(k_i = 1, 2, \ldots, 2n + 1\).

B. The objective function of the paths

Firstly, we define the objective function of air robot in path planning. The mission survival probability is a function of the probabilities of not being detected by enemy radar, not being killed if detected, and not crashing against the terrain, so an objective function is used in the air robot path planning can be defined as follows:

\[J = \int \left( \alpha_1 C^2 + \alpha_2 h^2 + \alpha_3 f_{m_i} \right) dt \quad (2)\]

Where the first term denotes the large cross-track deviations from the line connecting the start and target points, the second term penalizes the penetration paths that come dangerously close to known threat sites, and the third term minimizes the air robot’s altitude above level \(h\).

In this paper, only the horizontal path optimization is considered, so the objective function can be simplified to contain only the forenamed two terms [9]:

\[J = L_A + \delta \frac{1}{\sum_{i=1}^{2n+1} d_{i \text{min}}^{-1}} \quad (3)\]

Where \(L_A\) is the flight distance, \(d_{i \text{min}}\) is the distance from the node to the nearest threat, \(\delta\) is the threat avoided coefficient, and the bigger \(\delta\) is, the safer the air robot flight would be.

The flight distance is showed to be the sum of line distances between nodes in the flight line. The distance from node \(a(x_i, y_j)\) in vertical line \(L_i\) to node \(b(x_{i+1}, y_j)\) in vertical line \(L_{i+1}\) can be described as:

\[d_{ij} = \sqrt{\left(\frac{\left|AB\right|}{m}\right)^2 + (y_j - y_i)^2} \quad (j, g = 1, 2, \ldots, 2n + 1) \quad (4)\]

Therefore, the flight distance can be described as follows:

\[L_A = \left(\frac{\left|AB\right|}{m}\right)^2 + (y_{i+1} - y_i)^2 \quad (j, g = 1, 2, \ldots, 2n + 1) \quad (5)\]

\[+ \sum_{i=1}^{2n+1} \left(\frac{\left|AB\right|}{m}\right)^2 + (y_{i+1} - y_i)^2 \quad (j, g = 1, 2, \ldots, 2n + 1) \quad (6)\]

Suppose that there are \(q\) threats, each of which is described by a circle with the centre point \((x_j, y_j)\), and the radius \(r_j\), the distance between the node \((x_i, y_j)\) and the threat \(j\) can be described as:

\[d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 - r_j} \quad (6)\]

Thus, the distance between the node \((x_i, y_j)\) and the nearest threat can be described as follows:

\[d_{\text{min}} = \{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 - r_j} \}
\]

\[\ldots \{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 - r_j}\} \quad (7)\]

III. PRINCIPLE OF IWD ALGORITHM

In nature, we often see water drops moving in rivers, lakes, and seas. Figure 2 shows natural water drops moving in river.

When water drops move, they change their environment in which they are flowing, and the environment itself has substantial effects on the paths that the water drops follow. Considering the case in Figure 2, the paths that the river follows are often full of twists and turns, and the water drops have no eyes to find the destination. As the gravitational force is straight towards the earth center, the water drops would follow a straight path toward the destination with no obstacles and barriers, which is the shortest path from the source to the destination. In this process, the water drops always try to change the real path to make it a better path in order to approach the ideal path. This continuous effort changes the path of the river passes by. One feature of the water drop is that it flows which enables the water drop transfer an amount of soil from one place to another place in the front. This soil is usually transferred from fast parts of the path to the slow parts. As the fast parts get deeper by being removed from soil, they can hold more volume of water and thus may attract more water. The removed soils which are carried in the water drops are unloaded in slower beds of the river system. Generally, a water drop prefers an easier path to a harder path when it has to choose between several branches that exist in the path from the source to the destination.

Based on the observation of the natural water drops’ behaviors, an IWD model is developed, which possesses some of the remarkable properties of the natural water drops [7]:

1. The amount of the soil it carries now, Soil(IWD).
2. The velocity that it is moving now, Velocity(IWD).

Fig. 2 Natural water drops in river
The values of the both properties may change as the IWD flows in its environment. This environment depends on the problem at hand. In an environment, there are usually lots of paths from the source to the destination, and the position of the destination may be known or unknown. If we know the position of the destination, the goal is to find the optimum destination in terms of cost or any suitable measure for the problem.

We consider an IWD moving in discrete finite-length steps. From its current position to its next position, the IWD velocity is increased by the amount nonlinearly proportional to the inverse of the soil between the two positions. Furthermore, the IWD’s soil is increased by removing some of the path joining the two positions. The amount of soil added to the IWD is inversely and nonlinearly proportional to the time needed for the IWD to pass from its current position to the next position. This duration of time is calculated by simple laws of physics for linear motion. Thus, the time taken is proportional to the velocity of the IWD and inversely proportional to the distance between the two positions.

Another mechanism that exists in the behavior of an IWD is that it prefers the paths with low soils on its beds to the paths with higher soils on its beds. To implement this behavior of path choosing, we use a uniform random distribution among the soils of the available paths such that the probability of the next path to choose is inversely proportional to the soils of the available paths. The lower the soils of the path, the more chance it has for being selected by the IWD.

IV. IMPROVED IWD MODEL FOR AIR ROBOT PATH PLANNING

The basic mathematical model of IWD has firstly been applied to the TSP [7]. The aim of the TSP is to find the shortest path that traverses all cities in the problem exactly once, returning to the starting city. While the air robot path planning is to work out the optimal or suboptimal safe flight path in the proper time, along which the air robot is able to accomplish the prearranged task and avoid the hostile threats. There are some common points between TSP and air robot path planning, and IWD is a feasible way in solving air robot accomplishing the prearranged task and avoid the hostile threats. There are some common points between TSP and air robot path planning, and IWD is a feasible way in solving air robot path planning problem under complicated environments. The procedure of the improved IWD for air robot path planning can be described as follows:

Step 1. Initialization of static parameters: Set the number of water drops \( N_{\text{IWD}} \), the number of nodes decided by \( x \) axis and \( y \) axis, and the coordinates of node \((x, y)\) to their chosen constant values. The number of nodes and their coordinates depend on the problem at hand while the \( N_{\text{IWD}} \) is set by the user. The initial soil on each link is denoted by the constant \( \text{InitSoil} \) such that the soil in each node \((x_i, y_i)\) is set by \( \text{soil}(x_i, y_i) = \text{InitSoil} \). The initial velocity of IWDs is denoted by the constant \( \text{InitVel} \). The best path is denoted by \( P_B \) which is still unknown and its objective function value is initially set to infinity: \( J(P_B) = \infty \).

Step 2. Initialization of dynamic parameters: The velocity of each IWD is set to \( \text{InitVel} \) whereas the initial soil of each IWD is set to zero.

Step 3. For each IWD, place all of them in the starting point.

Step 4. For each IWD, choose the next node \((x, y)\) to be visited by the IWD when it is in node \((x_i, y_i)\) with the following probability:

\[
P^{\text{IWD}}(y_{i(j+1)}) = \frac{f(\text{soil}(x_{i+1}, y_{i(j+1)}))}{\sum f(\text{soil}(x_{i+1}, j))}
\]

\[
f(\text{soil}(x_{i+1}, y_{i(j+1)})) = \frac{1}{\xi_i + g(\text{soil}(x_{i+1}, y_{i(j+1)}))}
\]

\[
g(\text{soil}(x_{i+1}, y_{i(j+1)})) = \begin{cases} 
\text{soil}(x_{i+1}, y_{i(j+1)}) - \min(\text{soil}(x_{i+1}), 0) & \text{if } \min(\text{soil}(x_{i+1}, y_{i(j+1)})) \geq 0 \\
\text{soil}(x_{i+1}, y_{i(j+1)}) - \min(\text{soil}(x_{i+1}), 0) & \text{else}
\end{cases}
\]

Where \( \text{dis}(x_i, y_i), node(x_i, y_i) \) denotes the Euclidean distance between \( node(x_i, y_i) \) and \( node(x_i, y_{i(j+1)}) \). \( \xi_i \) denotes a small positive number to prevent a possible division by zero in the function \( f(.) \). The function \( \min(.) \) returns the minimum value among all available values for its argument.

Step 5. For each IWD moving from node \((x_i, y_i)\) to node \((x_{i+1}, y_{i(j+1)})\), update its velocity according to the following equation:

\[
\text{vel}^{\text{IWD}}(t + 1) = \text{vel}^{\text{IWD}}(t) + \frac{a_s}{b_s + c_s \cdot \text{soil}(x_{i+1}, y_{i(j+1)})}
\]

Step 6. For each IWD, compute the soil amount, \( \Delta \text{soil}(x_i, y_{i(j+1)}) \), that the current water drop IWD loads from its current path between two nodes \( i \) and \( j \):

\[
\Delta \text{soil}(x_{i+1}, y_{i(j+1)}) = \frac{a_s}{b_s + c_s \cdot \text{time}(node(x, y_i), node(x_{i+1}, y_{i(j+1)}); \text{vel}^{\text{IWD}})}
\]

\[
\text{time}(node(x, y_i), node(x_{i+1}, y_{i(j+1)}); \text{vel}^{\text{IWD}}) = \frac{\text{dis}(node(x, y_i), node(x_{i+1}, y_{i(j+1)}))}{\max(\xi, \text{vel}^{\text{IWD}})}
\]

Which computes the time taken to travel from \( node(x_i, y_i) \) to \( node(x_{i+1}, y_{i(j+1)}) \) with the velocity \( \text{vel}^{\text{IWD}} \). Here, the function \( c(\cdot, \cdot) \) represents the two dimensional positional vector for the node. The function \( \max(\cdot, \cdot) \) returns the maximum value among its arguments, which is used here to threshold the negative velocities to a very small positive number \( \xi \).

Step 7. For each IWD, update the soil of the path...
traversed by that IWD using the following equations:

\[ \text{Soil}(x_i, y_j) = (1 - \rho) \cdot \text{Soil}(x_i, y_j) - \rho \cdot \Delta \text{soil}(x_i, y_j) \]  

(14)

\[ \text{soil}^\text{IWD} = \text{soil}^\text{IWD} + \Delta \text{soil}(x_i, y_j) \]  

(15)

Where \( \text{soil}^\text{IWD} \) represents the soil that the IWD carries. The IWD goes from \( \text{node}(x_i, y_j) \) to \( \text{node}(x_i, y_{j+1}) \). The parameter \( \rho \) is a small positive number less than one.

**Step 8.** For each IWD, complete its path by using Step 4 to Step 8 repeatedly. Then, calculate \( L_k \) according to Equation (5) traversed by the IWD, and calculate the threat cost value according to Equation (6) and (7). In this way, calculate the path with the minimum objective function value \( J_{\min} \) among all IWD paths in this iteration. We denote this minimum path by \( P_M \).

**Step 9.** Update the soils of paths included in the current minimum path of the IWD, denoted by \( P_M \):

\[ \text{soil}(x_i, y_j) = (1 - \rho) \cdot \text{soil}(x_i, y_j) + \rho \cdot \frac{2 \cdot \text{soil}^\text{IWD}}{n_{\text{num}} \cdot (n_{\text{num}} - 1)} \quad \forall (x_i, y_j) \in P_M \]  

(16)

Where \( n_{\text{num}} \) denotes the number of nodes in \( y \) axis.

**Step 10.** If the minimum path \( P_M \) is the shorter than the best path found so far denoted by \( P_B \), then we update the best path by \( P_B = P_M \) and \( J(P_B) = J(P_M) \).

**Step 11.** Go to Step 2 unless the maximum number of iterations is reached or the defined termination condition is satisfied.

**Step 12.** The IWD algorithm stops here such that the best path is kept in \( P_B \) and its objective function value is \( J(P_B) \).

**V. EXPERIMENTAL RESULTS**

In order to investigate the feasibility and effectiveness of the proposed IWD approach to air robot path planning, a series of experiments have been conducted under complicated environments.

The basic IWD and improved IWD algorithm was implemented in a Matlab 7.2 programming environment on an Intel Core 2 PC running Windows XP SP2. No commercial path planning tools or IWD tools were used. In all experiments, the same set of parameter values were: \( a_s = 1000, b_s = 0.1, c_s = 1, a_v = 1000, b_v = 0.1, c_v = 1, N_{\text{IWD}} = 60, \text{InitSoil} = 1000, \text{InitVel} = 100, \rho = 0.05 \), the number of iterations \( N_I = 10 \).

Figure 3 shows the results comparison of air robot path planning by using basic IWD and improved IWD, and Figure 5 shows the results comparison of air robot path planning under more complicated environments.
Fig. 4 Case 2: results comparison of air robot path planning by using basic IWD and improved IWD under more complicated environments. (a) Path planning result by basic IWD. (b) Path planning result by improved IWD

From the experimental results, it is obvious that the proposed IWD optimization algorithm can find the feasible and optimal path for the air robot, while the basic IWD cannot effectively solve the path planning problem for air robot. This method provides a new way for path planning of air robot in exact application in the future.

VI. CONCLUSIONS

This paper presented an improved IWD optimization algorithm approach for air robot path planning in complicated environments. The simulation experiments show that this proposed method is a feasible and effective way in air robot path planning. It is also flexible, in that dynamic environments and pop-up threats are easily incorporated. Our future work will focus on the exact application of our proposed method in air robot path planning and re-planning, and multi air robots coordinated control is another problem in this field.

ACKNOWLEDGMENT

The authors are grateful to the anonymous referees for their valuable comments and suggestions.

REFERENCES