J Intell Robot Syst (2015) 79:3–19
DOI 10.1007/s10846-014-0079-9

A New Cloud Model Based Human-Machine Cooperative Path Planning Method

Xixia Sun · Chao Cai · Xubang Shen

Received: 28 October 2013 / Accepted: 23 June 2014 / Published online: 12 August 2014

Abstract In this paper, a fast human-in-the-loop path planning strategy in cluttered environments based on cloud model is proposed, and it is implemented in a human-machine cooperative Unmanned Aerial Vehicle (UAV) path planning system. Firstly, a dynamic guidance A* (DGA*) search algorithm is proposed to allow human's participation in machine searching loop. Secondly, online uncertainty reasoning based on cloud model is introduced to allow human's fuzzy decision about path direction and trending, then human's perception, expertise, and preferences are incorporated into the DGA* optimality process. Therefore, this effective cooperative decision support can provide a robust solution exploration space, overcoming some shortages of original A* algorithm, such as slow search speed, easily falling into local dead-ends, and so on. Experimental results demonstrate that the proposed method is much more efficient than original A* planner, and generates good solutions that match mission considerations and personal preferences.

Keywords Path planning · Human-machine cooperation · Dynamic guidance A* (DGA*) search · Cloud model · Uncertainty reasoning

X. Sun · C. Cai (✉) · X. Shen
State Key Laboratory for Multispectral Information Processing Technologies School of Automation, Huazhong University of Science and Technology Wuhan, Hubei, 430074, People's Republic of China
e-mail: caichao@hust.edu.cn

X. Sun
e-mail: sunxixia@126.com

X. Shen
e-mail: shenxubang@163.net

1 Introduction

Path planning is one of the fundamental issues in UAV research, which aims at obtaining the optimal or sub-optimal path between an initial position and the desired destination [1], satisfying a series of constraints. In the past decades, various automated path planning approaches have been proposed and the majority of them fall into the following categories [2]: 1) Graphic-based algorithms[3]. 2) Grid-based algorithms [4]. 3) Potential field algorithms [5].

It has been demonstrated that an optimal solution of the path planning problem is NP-complete problem in nature [1]. When the search space is complex and a resultant path must satisfy the vehicle dynamics, direct applications of automated path planners are associated with huge computational resources and time consumptions. For example, A* algorithm, which is the most often used search algorithm for path generation [2, 6], is easy to get trapped in map dead-ends and requires significantly more computation time in the presence of map dead-ends [7]. Unlike automated path planners, which often fall into local minima, humans are well

capable of making intuitive decisions because they have powerful heuristics, qualitatively described as the ability to apply their perceptive grasp of the overall problem space and environment. Therefore, it is important to incorporate human's intelligence into the search process of automated planner.

In recent years, multiple studies explored the benefits of allowing a human operator and a computer algorithm to work cooperatively in the path planning tasks, and investigated how humans generate and optimize paths to support human-machine cooperative systems developments [8–14]. Cummings et al. studied the benefits of allowing a human operator to manually adjust robot paths generated by an automated planner if they do not reflect objectives of the users [13]. Andrew et al. developed a Collaborative Mission Planning & Autonomous Control Technology system implementing evolutionary algorithms to carry out mission planning, where the human operator could specify mission functions, constraints and priorities [14]. There has also been little research so far that investigates strategies by which humans could guide a computer algorithm in a collaborative process, especially when working in time-pressing situations with significant uncertainty [15–17]. For example, recent research by Clare et al. suggested that providing operators with the ability to dynamically modify the objective function weightings of an automated planner during planning process could have performance benefits [16]. Badillo et al. proposed a user-centric memetic algorithm, and they reported that interactive memetic algorithms could take advantage of good quality human feedback and behave in a proactive manner [17]. Most of these previous research works were focused on human-machine interactions such as the addition or removal of waypoints and changing the inputs to the algorithm. Human experience and high level knowledge that are important for the path planning problems were not effectively implemented in their systems. Therefore, it is important to take a balance between human and automated planner collaboration to take full advantage of computational power of automation, as well as the experience and knowledge based reasoning of humans.

It is well recognized that human knowledge and judgments are represented by uncertain and imprecise patterns for a complex problem. Although various studies demonstrated the capacity of fuzzy logic in uncertainty management and human knowledge implementation [18, 19], due to the complexity of path planning problems, a large number of fuzziness and randomness existing in human knowledge were not fully considered in their key steps. Cloud model proposed by Li et al. [20] is based on both fuzzy theory and probability, combining fuzziness and randomness together to map qualitative concepts and quantitative data. Hence, it overcomes subjective randomness in fuzzy membership grade when being determined, which has been successfully applied to various areas [21–23].

Therefore, based on cloud model, this paper proposes an effective human-in-the-loop path planning strategy. The motivation is to combine the capability of a human to quickly compute an approximate and coarse solution with the computational capability of the automation. During the planning process, human operator guides the planner's searching direction through specifying the command of "turn right" or "turn left", it means which side the path will pass by to avoid the closest threat. To enable human operator to actively participate in A* search process, a DGA* planner is proposed. To cope with uncertainty and imprecision in human knowledge and judgments about path direction and trending, cloud model based online reasoning is introduced to provide reasonable instant guide points, effectively implementing human's perception, knowledge, and preferences to guide the DGA* planner in a collaborative process. Therefore, the problem of getting trapped in map dead-ends is avoided, and complexity of the path planning problem is effectively reduced, enabling humans and computers to discover effective solutions efficiently.

The rest of this paper is organized as follows. Section 2 presents details of the proposed human-machine cooperative techniques, followed by the proposed DGA* path planner. The introduction of cloud model to generate instant guide fields for DGA* planner is briefly described in Section 3. The experimental results are given in Section 4. Finally, the paper is concluded in Section 5.

2 Human-Machine Cooperative Path Planning Method

This section presents techniques to combine strengths of both a human operator and A* planner, and

proposes a DGA* planner. The developed DGA* planner can effectively utilize the heuristic information provided by guide points to judge search direction, and return the optimal path passing by guide points

2.1 Guide Points in Path Planning

In our human-machine cooperative path planning system, the planning environment is a digital terrain elevation map (DTEM) with $m \times n$ grid cells (or, nodes). In all figures, S and D denote the start node and destination node, respectively. As depicted in Fig. 1, circles on the map represent Air Defense Units (ADUs) like united detect radar and/or surface to air missile (SAM) groups, while the radius represents the relative risk range of each ADUs. UAVs are not permitted to enter the regions with these circles, even close to these circles UAVs will be vulnerable to the threats with a certain probability associated with the distance away from the threat centers.

As previously mentioned, A* solver is the most often used graph search algorithm for path generation [2, 6]. In this study, A* is selected as the automated path planner since it is an optimal, robust and informed search method. A* algorithm could be able to effectively collaborate with the human operator since we can provide reasonable heuristics to guide the search. Despite the need for in-flight re-planning, a heuristic re-planning algorithm (e.g. D*) is not selected because of the presence of many dynamic elements [24]. For example, changing battlefield environments can invalidate prior search results for a large number of nodes in the search space. In these

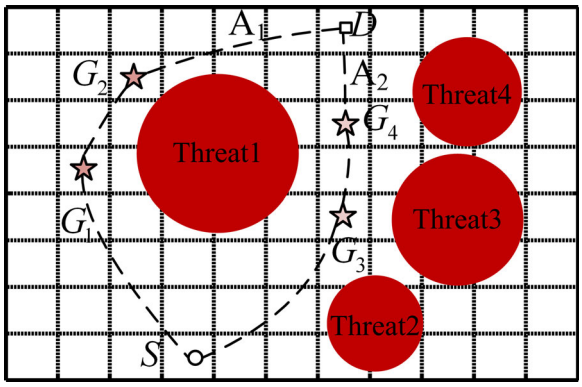

Fig. 1 Influences of human's awareness and preferences on the resultant path solutions. G_1, G_2, G_3, and G_4 are guide points specified by human operator. The two *dash curves* A_1 and A_2 represent two reference paths

situations, planning from scratch using A* is more efficient than re-planning algorithms [25]. A random algorithm like RRT is not selected because it can not guarantee optimality of the solution, and it is difficult to predict the behavior of the algorithm [12]. However, A* algorithm can not pre-perceive some zones without solution path from start position to destination position (e.g., in front of a threat zone), and can easily fall into map dead-ends [7]. Hence, it is foreseeable that it will perform better if the human has opportunities to intervene search tactics and applied his heuristic knowledge in an efficient way to guide the A* optimality process. As humans can visually identify promising areas of a solution search space, we can let human operator identify several intermediate waypoints along a potential solution and A* planner search the optimal path passing by guide points afterwards. As shown in Fig. 1, we can set the guide points G_1 and G_2 around threat zones to provide a general guideline. It is needed to be mentioned that a guide point G_i is such a point that can guide A* search process and does not have to be passed by the resultant path. Guide field is the region with the guide radius r_i associated with it. A guide point is considered reachable only if the path passes by any point in the guide field. If a point must be passed by the UAV, its radius is set to zero.

There are other benefits of incorporating human experience and preferences into optimization process. Automated planner usually automatically generates a minimum cost solution based on a predetermined cost function. In some cases, it is difficult to express the complete objective of a human through an a priori coded objective function, and the predetermined cost function may not reflect the true state of all variables [13, 17]. The optimal or near-optimal solutions provided by automated optimization algorithms may result in erroneous decision support [13, 26]. Therefore, it is desirable to incorporate human judgments into the optimization process of automated planner, as having a solution that is good enough, robust, desirable, and quickly reached is generally preferable to the optimal one that requires complex computation and extended periods of times [26].

2.2 Dynamic Guidance A* Planner

Once guide points are determined, we assume that, the order of the guide points from start position to

destination position is in ascending order and the destination node is regarded as the last guide point with guide radius zero. To exploit the information provided by guide points, the search engine employs a sparse A* search (SAS) algorithm similar in [4] to search the map and plan the minimal cost path passing through guide fields under various path constraints, including maximum turning angle $\alpha_{\max}$ and minimum turning radius $r_{\min}$.

SAS algorithm incorporates various path constraints into its planning process, and was further extended by Zheng et al. [27] to solve the on-line path planning problem in unknown planning environment. As illustrated in Fig. 2, SAS employs fan-tails to expand nodes until the destination node is reached. Note that when a node is expanded in SAS search process, expanding direction (i.e., direction of the line between the node and its parent node) determines the successor nodes to be explored. Therefore, expanding direction of each node is also recorded when being searched, for example, expanding direction at the node P_5 is the direction of $\overrightarrow{P_2P_5}$ as in Fig. 2.

With respect to the planning paths with guide fields, two major issues need to be addressed: Firstly, as shown in Fig. 2, the computed path must pass through all guide fields in order. Secondly, A* heuristic cost function needs to be carefully designed to effectively guide the planner to search the optimal path passing through guide fields. To handle these issues, a novel path planner called DGA* search is proposed in this research, described as follows.

To make sure the resultant path passes through guide fields in order, each node of the search tree has a guide number attribute recording which guide point the node should be expanded towards. As illustrated in Fig. 2, initially, set the start node's guide number to 1, indicating that the start node should be expanded towards G_1. At each expanding step, a node n is extracted from Open list and its successor nodes are being explored and evaluated. Assume that guide number of node n is k. When a successor node n' to n is explored, if n' lies in the guide field of G_k, set guide number of n' to $k+1$, indicating the search tree has passed by G_k. Otherwise, set guide number of n' to k, implying the computed path from the start node to node n' found in the search tree so far hasn't reached G_k yet.

By forcing path solution to pass through guide fields, search space is effectively reduced, enabling the path planning problem in this pruned search space to be more efficient to solve. Detailed implementations of the DGA* algorithm are summarized as follows:

Step (1) Set the start node's guide number to one and push it to the Open list.

Step (2) Remove the minimal cost node $n$from the Open list to the Close list.

Step (3) Generate all the successor nodes of node n.

Step (4) For node n with the guide number k, if each of its successor node n' is within the guide field of G_k, set its guide number to $k+1$, otherwise, its guide number is unchanged.

Step (5) For each successor node n', compute its total cost $f(n') = g(n') + h(n')$, where $g(n')$ is the actual cost of the path from the start position to node n'. $h(n')$ is the estimated cost of the path from node n' to the destination position.

1) If n' is neither in the Open list nor in the Close list, place it in the Open list.
2) If n' is already in the Open list or the Close list, and its new expanding direction is different from its previous expanding direction, place it in the Open list.
3) If n' is already in the Open list or the Close list and its new expanding direction is the same as its previous expanding direction. Compare the new $f(n')$ with the previously calculated $f(n')$. If the new value is lower, direct

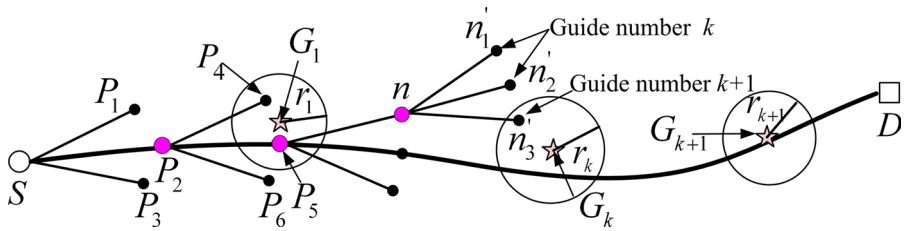

Fig. 2 Node expansion process with guide fields

its pointer to node n, and change its total cost to the new $f(n')$. If $n'$is already in the Close list, move it back to the Open list.

Step (6) Repeat Steps (2)–(5) until the termination condition is reached. The final path can be generated by tracing back up the search tree once the minimal cost node extracted from the Open list is the destination node.

2.3 Evaluation Function

This part describes the cost function that measures the fitness value for a specific path Z. In this study, the cost function is defined as follows:

$$J(Z) = \omega_1 J_{length} + \omega_2 J_{hazard} + \omega_3 J_{smooth} + \omega_4 J_{height} \tag{1}$$

where J_{length}, J_{hazard}, J_{smooth}, and J_{height} are costs of distance, threat, smooth, and height, respectively, and ω_i $(i = 1, \ldots, 4)$ are their corresponding weight coefficients, which can be assigned according to preferences of experts and mission situations. For a given path Z, it can be described as a sequence of nodes:

$$Z = (C_1, C_2, \ldots, C_M) \tag{2}$$

such that C_1 is the start node, C_M is the destination node, and M is the number of nodes. Each node C_i is specified by its 2-D coordinates (x_i, y_i). Then the four cost components can be formulated as follows.

Let $d(A, B)$ denote the distance from node A to node B. Path length cost J_{length} is defined as the total length of line segments from the start node to the destination node.

$$J_{length}(Z) = \sum_{i=1}^{M-1} d(C_i, C_{i+1}) \tag{3}$$

Hazard cost J_{hazard} penalizes paths dangerously close to threat sources. To simplify calculations, a computationally efficient and acceptably accurate approximation to the exact solution is to calculate the hazard cost at several locations along a path segment and take length of the path segment into consideration [28, 29]. In this study, J_{hazard} is expressed as:

$$J_{hazard}(Z) = \sum_{i=1}^{M-1} d(C_i, C_{i+1}) F_i \tag{4}$$

where F_i is hazard cost of the i th path segment. As Fig. 3 shows, F_i is calculated at three points (locations of these points can be changed according to shape of the threats) along each path segment [28], defined by:

$$F_i = \sum_{j=1}^{K} (P(d_{1/6}^j) + P(d_{1/2}^j) + P(d_{5/6}^j)) \tag{5}$$

where $d_{1/6}^j$ is the distance from the $1/6$ point on the i th path segment to the j th threat, and K is the number of threat sources. $P(d_{1/6}^j)$, $P(d_{1/2}^j)$, and $P(d_{5/6}^j)$ are instantaneous probabilities of radar detection of the UAV by the j th threat, calculated according to radar equation [30]:

$$P(R) = \frac{1}{1 + c_2 \left(\frac{R^4}{\sigma}\right)^{c_1}} \tag{6}$$

where c_1 and c_2 are constants associated with model and lethality of the radar. σ is radar cross section (RCS) of the UAV. R is the distance from the UAV to the threat source.

In view of the physical limitation of UAV, it usually does not wish to make severe turns in some flight scenarios. Turning angle at node C_i is defined as the angle difference between two adjacent path segments $\overrightarrow{C_{i-1}C_i}$ and $\overrightarrow{C_iC_{i+1}}$. Smooth cost J_{smooth} is measured by sum of all node turning angles. Height cost J_{height} is the average terrain altitude of the path.

2.4 Heuristic Function

To effectively guide the planner to search the optimal path passing through guide fields, the heuristic cost of

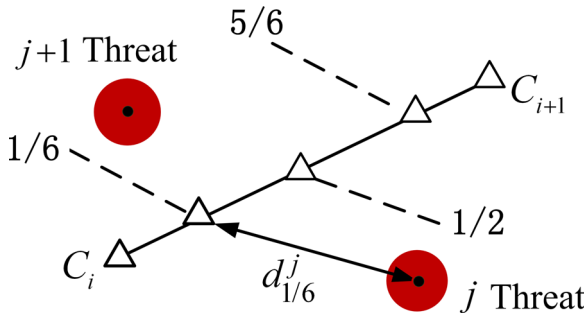

Fig. 3 Model of the UAV threat cost

the DGA* planner, $h(n)$, is designed here, expressed as:

$$h(n) = \begin{cases} \omega_1 h^{'}(n) & h^{'}(n) \geq 0 \\ 0 & h^{'}(n) < 0 \end{cases} \tag{7}$$

where ω_1 is the weight coefficient of distance cost. Let N denote the number of guide points between guide point G_k (guide point of node n) and the destination node D. $h'(n)$ is given by,

$$h'(n) = d(n, G_k) - r_k + \sum_{i=0}^{N-1} (d(G_{k+i}, G_{k+i+1}) - r_{k+i} - r_{k+i+1}) + d(G_{k+N}, D) - r_{k+N} \tag{8}$$

where $d(A, B)$ represent the distance between node A and node B. r_j is radius of the jth guide field (shown as Fig. 4). This heuristic function adopts Euclidean distance to the DGA* algorithm, and we have the following proposition:

Proposition 1 *The heuristic function designed as formula (7) and (8) for the DGA* planner is admissible.*

Proof Assume that path Z' is the minimum cost path from node n to the destination node in the reduced space imposed by guide fields, that is, Z' starts from node n, passing through guide field of G_k and the following guide fields, and terminates at the destination node D, i.e.,

$$Z' = (n, \ldots, T_k, \ldots, T_{k+1}, \ldots, T_{k+N} \ldots, D) \tag{9}$$

where $T_{k+i}\,(0 \leq i \leq N)$ are path nodes lying in the corresponding guide fields of $G_{k+i}(0 \leq i \leq N)$.

As demonstrated in Fig. 4, in triangle nG_kT_k, the difference between length of edge $\overrightarrow{nG_k}$ and length of edge $\overrightarrow{G_kT_k}$ is smaller than length of $\overrightarrow{nT_k}$, and length of $\overrightarrow{G_kT_k}$ is smaller than guide radius r_k, hence,

$$d(n, G_k) - r_k \leq d(n, T_k) \tag{10}$$

And for the same reason, one has:

$$d(G_{k+N}, D) - r_{k+N} \leq d(G_{k+N}, D) \tag{11}$$

Similarly, in quadrilaterals $G_{k+i}G_{k+i+1}T_{k+i+1}T_{k+i}$ $(0 \leq i \leq N-1)$,

$$d(G_{k+i}, G_{k+i+1}) - r_{k+i} - r_{k+i+1} \leq d(T_{k+i}, T_{k+i+1}) \tag{12}$$

Combining Eqs. (10), (11), and (12), it follows that,

$$h'(n) \leq d(n, T_k) + d(T_{k+N}, D) + \sum_{i=0}^{N-1} d(T_{k+i}, T_{k+i+1}) \tag{13}$$

According to its definition, distance cost of Z', $J_{lengh}(Z')$, is sum of all path segment lengths from node n to the destination node, and T_{k+i} $(0 \leq i \leq N)$ are path nodes of Z', apparently,

$$d(n, T_k) + d(T_{k+N}, D) + \sum_{i=0}^{N-1} d(T_{k+i}, T_{k+i+1}) \leq J_{lengh}(Z') \tag{14}$$

Let $h^*(n)$ represent the actual cost of path Z'. Combining Eqs. (13) and (14), $h'(n) \leq J_{lengh}(Z')$ holds, which leads to $h(n) \leq h^*(n)$. Therefore, the designed heuristic function is admissible [31], and the DGA* planner will find the optimal path in the reduced search space imposed by guide fields if one exists. □

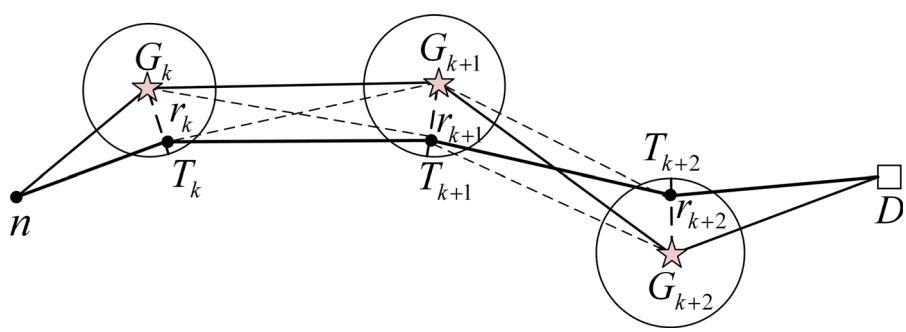

Fig. 4 The minimum cost path from node n to G_k, passing through the following guide fields, finally to the destination position

3 Cloud Model Based Guide Points Generation

This section proposes an effective human-in-the loop strategy based on cloud model. It begins with a description of the general scheme of the strategy in subsection 3.1. Then, cloud model and human knowledge representation based on cloud model are presented in subsections 3.2 and 3.3, respectively. Finally, the introduction of cloud model based online reasoning to mimic the process of human decision about path direction and trending is presented in subsection 3.4.

3.1 Human-machine Cooperation in the A* Search Process

This part presents general scheme of our human-in-the-loop strategy. In real applications, battlefield changes dynamically, and it is difficult to specify suitable guide points in advance in the presence of pop-up threats. Therefore, it would be valuable to allow human operator to participate in A* optimality process to actively guide the algorithm. As discussed previously, to avoid map dead-ends, A* search engine can be directed to guide points, thus our goal for handling threats avoidance is introducing effective human interactions and intelligence to provide reasonable instant guide points. The schematic of the concept is given in Fig. 5.

In Fig. 5, point A is the position of an expanded node before threat 2. The solid curve is the path from the start node to current expanded node that has been planned, and the line of dashes is the heuristic path (line between current expanded node and the destination node). During A* search process, if there is a large threat zone in front of the expanded node, huge numbers of useless nodes near the threat zone will be searched, till one tree can bypass this threat zone. To avoid such meaningless search, an instant guide point is necessary. Cloud model help us generate instant guide fields as follows.

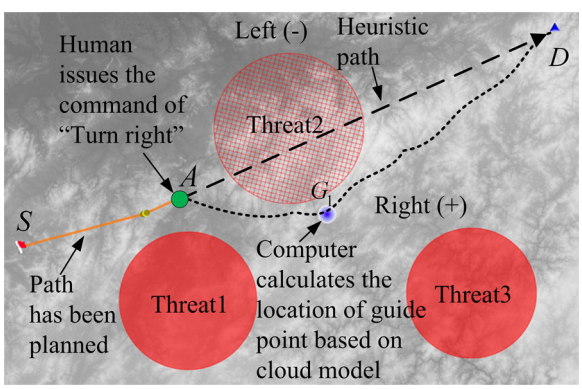

Fig. 5 Two-dimensional illustration of a simple threat zone avoidance problem. Threat 2 was shown as a gridding circle to alert the operator and the planning process of DGA* is suspended. After the operator had specified a coarse pass azimuth, on-line reasoning based on cloud model generated a guided field for the DGA* planner to circumvent the threat zone

When distance from the expanded node to the closest threat zone along the heuristic path is smaller than a given threshold, the closest threat zone that the heuristic path passes through will be shown as gridding circle to notify the human. Meanwhile, the planning process of DGA* planner pauses to wait for the operator to specify a coarse pass azimuth. Based on his perceptive grasp of the of the overall problem space and battlefield environment, the human operator issues the command of "turn right" or "turn left", representing the turn that the path must make to avoid the closest threat. As Fig. 5 demonstrates, direction of the heuristic path is always taken as the zero direction, with the negative direction to its left and the positive direction to its right, respectively. The automated planner interprets the command, and on-line reasoning based on cloud model provides a reasonable intermediate guide field (the smaller circle near threat 2). Then the DGA* planner effectively employs the information provided by this guide field to judge search direction and seek path going through it. In this way, different agents' exploration of the potential solution to the path planning problem is coordinated and the problem of getting trapped in dead-ends can be avoided.

When multiple UAVs are managed by a single operator and/or under complex environments, human operator plays a fundamental role as a high level mission manager, and will need to comprehend a large amount of information under time-pressure to make effective decisions. Cummings and Mitchell proposed an upper bound formulation for predicting the number of UAVs that an operator can control [32]. Because the process of DGA* planner can plause to wait for operator decision if there is a threat in front of the searching direction, the number of DGA* planning process that an operator can manage could be unlimited in this study. To measure the human-machine cooperation performance, the metric of workload is

considered. In this study, workload is characterized by a utilization metric, calculated as the ratio of the total operator "busy time" (including the three major elements of information processing: perception, cognition, and action) to the total planning time [15, 16].

Note that center of the guide field should be placed at a position with a certain safe distance away from the threat circle to ensure safety of the path. Refer to Fig. 6, point A is the position of the current expanded node. Line MN is either to the left or the right of line AD and tangent to the threat circle at point G. The guide field's center E is set in the normal direction of line AD with a safe distance d to tangent G.

Based on multi-objective optimization theory [33], safe distance d determines the risk-taking relations between the planned path and the given threat. A larger d results in a farther path away from the threat with longer distance and smaller threat risk, while a smaller d results in a shorter path with larger threat risk.

As mentioned previously, it is desirable to incorporate human knowledge and preferences into optimization process to generate paths satisfying personal preferences. Previous study demonstrated that operators tended to think less in terms of numerical optimization when planning paths but more in qualitative words about the overall goals or preferences [34]. It also has been argued that developing a method to communicate their goals and preferences to the optimization algorithm would result in solutions that match the desires of the operator [34]. However, human knowledge and judgments are fuzzy and uncertainty in nature, and some theoretical methods are required for dealing with both fuzziness and randomness in human knowledge and preferences.

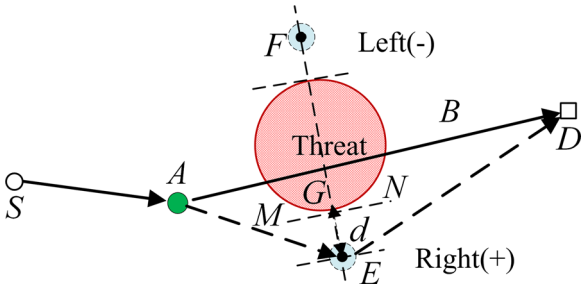

Fig. 6 The geometry relationship between the guide field and the threat zone

In previous research, safe distance was usually predetermined [35]. To obtain a reasonable path better matching battlefield situation and personal preferences, we introduced to generate instant field by using cloud model based uncertainty reasoning in our previous study [36], which can simulate the process of human reasoning in specifying a guide point. Cloud model integrates fuzziness and randomness to transform qualitative knowledge described in a natural language to distribution patterns of quantitative values, effectively expressing the uncertainty in human knowledge and the association of randomness and fuzziness [22, 23]. Therefore, it can allow the operator to communicate his/her desires to the automated planner in a highly flexible and intelligent manner. However, factors that affect position of the instant guide point considered in that paper were of limited representations of battlefield situations. In this research, we make further improvements and incorporate mission requirement into knowledge base, and therefore, the reasoning results are more reasonable. Meanwhile, the DGA* can effectively use the information provided by guide fields to find the optimal path passing through pre-determined guide fields. In this way, the respective strengths of the human and the automation can be balanced to create an effective and efficient planning process that can generate satisficing, or good enough paths. The following parts present a brief description of cloud model and how it is applied in the human-machine cooperative path planning process.

3.2 Cloud Model

Cloud model is an uncertainty conversion model between qualitative knowledge description and quantitative value expression, defined as follows [23].

Suppose that T is a language value of domain U, mapping $C_T(x)$: U $\rightarrow$ [0, 1], $\forall x \in U, x \rightarrow C_T(x)$. Then the distribution of $C_T(x)$ in U is called the membership cloud of T, or cloud in short, and each projection is called a cloud drop in the distribution. If distribution of $C_T(x)$ is normal, it is named normal cloud.

Normal cloud model employs expectation Ex, entropy En, and super entropy He to represent a qualitative concept. Ex determines the center of the cloud. En measures the uncertainty of the concept. He is the uncertainty measurement of the entropy, reflecting

cloud drops' dispersive degree. For example, a normal cloud $(0,\ 1/3,\ 0.03)$ representing linguistic term "about zero" is shown in Fig. 7.

3.3 Knowledge Representation Based on Cloud Model

This part describes how to represent human knowledge about location of the guide fields based on cloud model. If-then rules composed of linguistic variables provide a scientific formalism for implementing human knowledge. Therefore, we attempt to extract human knowledge in the form of if-then rules. Each rule is of the form "If A_1 and A_2 and $\ldots A_n$, then B", where A_1, A_2,..., A_n, and B are linguistic variables modeled by normal clouds, and n is the number of input reasoning parameters. Multiple if-then rules form a rule base, established as follows.

According to formula (6), it can be concluded that the major factors that affect detecting probability are lethality of the detecting radar and RCS of the UAV. Moreover, mission requirement can also affect the value of safe distance, the more important the mission is, the shorter the path should be to ensure that the UAV reaches the destination in a shorter time. To obtain a safe distance adaptively, input reasoning parameters are chosen as:

(1) Lethality of the detecting radar.

For each type of radar, its detecting efficiency is different from each other, and poses different lethality to the UAV. Values of radar parameters for various models of radars can be collected and estimated by sensor network. We can share and receive these values from the ground control station, other mobile device users in the network, or the sensors in the network through communication network [37]. Threat lethality can be provided by threat assessment and normalized to [0, 10]. In this research, linguistic terms "Very Low" (VL), "Low" (L), "Moderate" (M), "High" (H), and "Very High" (VH) are employed to represent lethality of the detecting radar. As Fig. 8 displays, these linguistic terms are transformed into normal clouds $(1, 0.5, 0.06)$, $(3, 0.5, 0.05)$, $(5, 0.5, 0.05)$, $(7, 0.5, 0.05)$ and $(9, 0.5, 0.06)$, respectively.

(2) RCS of the UAV

Radar cross section (RCS) of an UAV is a measure of the UAV's ability to reflect radar signals, and depends on factors such as material of which the UAV is made and size of the UAV. According to current open literature, RCSs of UAVs range from 0 to 8 or much bigger [38, 39]. Linguistic terms "Very Small" (VS), "Small" (S), "Medium"(M), "Large" (L), and "Very Large"(VL) are used for rating RCS of the UAV.

(3) Importance of the mission

Relative importance of the mission is established by human operator based on his judgment of mission requirement. It takes value in [0, 10] and is divided into five scales, which are "Very Unimportant" (VU), "Medium Unimportant" (MU), "Important" (I), "Medium Important" (MI), and "Very Important" (VI).

Output linguistic variables describing safe distance ranging from 0 to 10 are also distinguished by five labels: "Very Near" (VN), "Near" (N), "Medium" (M), "Far" (F), and "Quite Far" (QF).

Similarly, linguistic words describing RCS of the UAV, importance of the mission, and safe distance are all transformed into their corresponding

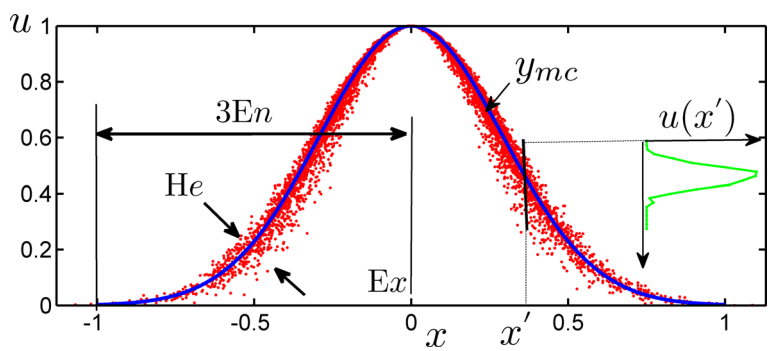

Fig. 7 Membership cloud for linguistic term "about zero". The membership degree at x', $u(x')$,is a random number with a probability distribution illustrated on the right of the figure. $y_{mc}$is the cloud expectation curve

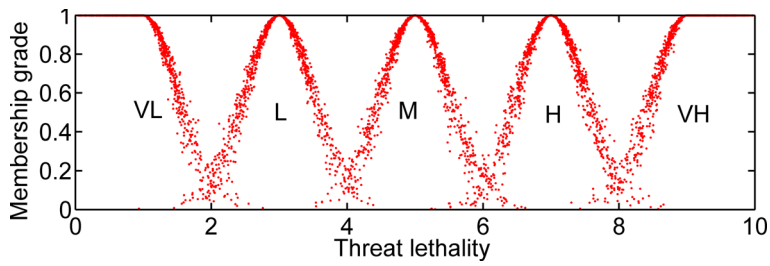

Fig. 8 Normal clouds for describing lethality of the detecting radar

synonyms in Fig. 8. For example, linguistic terms "Large", "Medium Important", and "Far" are all represented by normal cloud (7, 0.5, 0.05).

The typical inference rule that expresses human knowledge is of the form "If DT and DR and DM, then DB". For instance, a specific inference rule is: "If DT_3 and DR_4 and DM_5, then DB_3", interpreted as: if lethality of the threat is moderate, UAV's RCS is small, and the mission is very important, then the guide field is near to threat circle. Based on experience and knowledge of human experts, a knowledge rule base can be made. Partially typical rules are listed in Table 1.

3.4 Online Uncertainty Reasoning Based on Cloud Model

In this part, the cloud theory based uncertainty reasoning is introduced to generate instant guide field, which is more consistency with human thinking than the traditional fuzzy reasoning [20]. In the cooperative path planning process, the uncertainty reasoning engine [36, 40] evaluates the rules, and generates an appropriate conclusion (safe distance d) based on the required input situational battlefield data (x_a, x_b, x_c), described as:

$$d = \Phi(x_a, x_b, x_c) \tag{15}$$

where x_a, x_b, and x_c represent lethality of the threat, RCS of the UAV, and importance of the mission in the situational battlefield environment, respectively. With safe distance d, position of the guide field can be determined according to Fig. 6. Meanwhile, guide radius is provided by backward cloud generator [41], quantitatively revealing uncertainty about location of the guide point, which is a different way from fuzzy reasoning.

Table 1 Fuzzy reasoning rules

Input variable			Output variable
DT1 (VH)	DU_1 (VS)	DM_1 (VU)	DB_1 (VF)
DT_1 (VH)	DU_1 (VS)	DM_2 (MU)	DB_1 (VF)
DT_1 (VH)	DU_1 (VS)	DM_3 (I)	DB_2 (F)
DT_1 (VH)	DU_1 (VS)	DM_4 (MI)	DB_3 (M)
DT_1 (VH)	DU_1 (VS)	DM_5 (VI)	DB_4 (N)
.....			
DT_5 (VL)	DU_5 (VL)	DM_5 (VI)	DB_5 (QN)

4 Experimental Results

In this section, we tested the proposed method in various scenarios to show its feasibility and effectiveness. We also compared performance of the proposed method with automated A* planner, and the human-machine cooperative path planning strategy by using operator-specified guide points.

The algorithm was implemented in a C++ and Qt 4.6.3 based software simulation environment on PC with Intel Core2 Duo E7400 CPU running Windows XP. A DTEM with resolution 90m×90m per pixel and different sets of simulative threat data were tested. In the following pictures, flag and triangle represent start position and destination position. Path solutions obtained from the planer were smoothed by using circular arcs. There are three dots on each arc of the final flyable path. The first and third dots represent the start point and the end point of the turning arc. The second dot is the intersection point between two adjacent path segments.

Unless otherwise specifed, the following parameters were adopted. $\alpha_{\max} = 30°$, $r_{\min} = 3.4$ Km, $\sigma = 2.1$ m^2, $\omega_1 = \omega_2 = \omega_3 = \omega_4 = 0.25$. Human's

judgment on the importance of the mission was set to 5, lethality degrees of threats were all set to 6.

4.1 Cooperative Path Planning Process

In this part, we performed an experiment to deeply demonstrate effectiveness of the human-in-the-loop path planning process. In this experiment, we supposed that lethality degrees of threat 1 and threat 3 were 6.6 and 8, respectively.

As shown in Fig. 9a, there was a warning alert and a selection box appeared for the human to specify a coarse pass azimuth relative to threat 1. Based on his global perception of the battlefield situation, the operator made the on-line decision of specifying the coarse pass azimuth.

Assume that the human operator pressed the selection box of "turn left". Under these battlefield circumstances, the cloud theory based uncertainty reasoning automatically generated a reasonable guide field with a distance of 6.10 Km away from threat circle 1 (shown as Fig. 9b). The guide radius is 5.74 Km, quantitatively revealing human's uncertainty about location of the guide point. Then the DGA* planner effectively exploited the heuristic information provided by this guide field to seek path going through it, and therefore, threat 1 was circumvented along a reasonable direction and the problem of getting trapped in dead-ends was avoided.

The DGA* search process continued till threat 3 was passed through by the heuristic path. Using the same strategy, another guide field with a distance of 10.11 Km away from threat circle 3 was produced. The picture in the bottom left of Fig. 9c shows the partial enlarged details of the geometric relationships between threat 1 and its associated guide field, while the picture in the top right of Fig. 9c illustrates the partial enlarged details of the geometric relationships between threat 3 and its associated guide field.

From the two reasoning results, it can be observed that the more dangerous the threat, the further away the guide field from the threat circle, emulating knowledge-based reasoning process of human in specifying a guide field realistically and reflecting the battlefield situation accurately. This process went on until a whole path incorporating human intelligence and reflecting mission situations was planned.

4.2 Experimental Results Comparison and Analysis

To validate the feasibility and effectiveness of our cloud model based DGA* (CDGA*) planner, we compared it with automated A* planner, and the human-machine cooperative path planning strategy recently proposed in [12, 13], where guide points for the DGA* planner were manually set by human operator. 16 participants consisted of 10 graduate students from our research group and 6 researchers from our cooperation partner participated in the experiments, and they were trained for the path planning tasks befrore experiments. The 16 participants consisted of 11 men and 5 women and none of them had served in the military. Ages of the participants range from 22 to 36 with an average of 28 and a stand deviation of 2.48. In the follwing experiments, the time that the operatr spent on the planning tasks is the average time that these participants spent.

Fig. 10a shows the path returned by automated A* planner. At the beginning of the search process, A* planner expanded nodes along a straight line toward the destination, as the search process propagating, it searched large numbers of useless nodes near the first

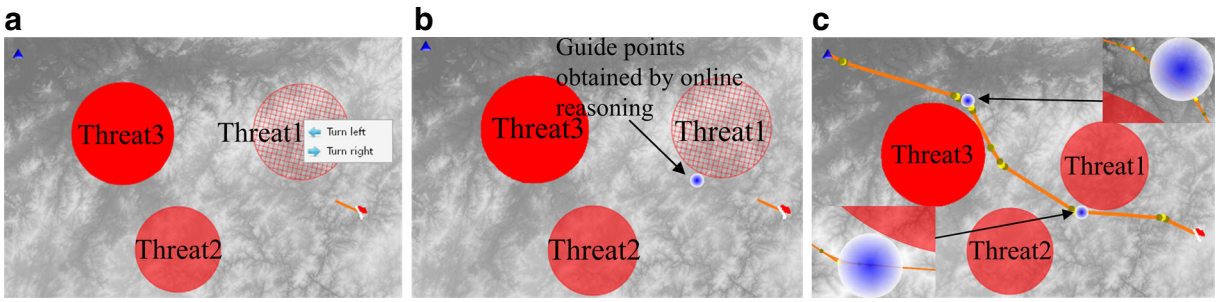

Fig. 9 Human-machine cooperation in the planning process: **a** human specified a coarse pass azimuth; **b** a guide field was provided by cloud model based online reasoning; **c** the final computed path directed by guide points

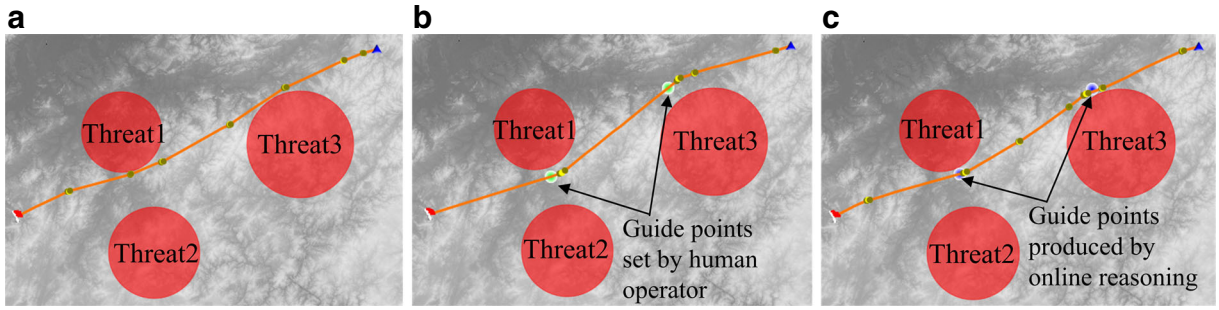

Fig. 10 Comparison of results produced by A*, DGA*, and CDGA* planners: **a** path planned by automated A* planner; **b** path planned by DGA* planner using operator-specified guide points; **c** path planned by CDGA* planner

threat before it found the feasible path avoiding threat 1, taking 110.26 seconds to find the final path. Sizes of Open list and Close list were 10746 entries and 3210 entries, respectively.

As mentioned before, intermediate waypoints were usually specified by human operator or located with a pre-determined distance around threats zones [13, 35]. As shown in Fig. 10b, the two small circles near threat 1 and threat 3 are guide fields specified by human operator during DGA* search process, which took this human operator 6.70 seconds (The time that human operator spent in the path planning tasks depended on his proficiency in using the planning system. Operator's proficiency in using the system is measured by his/her frequency in map use and video-game experiences. Numbers of experiments demonstrated that there were there were no significant differences in time consumptions based on military experiences, and there were differences in time consumptions between experienced and inexperienced users [16, 42]) to specify. With heuristic information provided by guide points, search engine was directed to guide fields, thus, threats were circumvented. However, the resultant path as well as time and memory consumptions depend on locations of guide points, which require human operators' efficiency in map-use and could not implement human knowledge that are associated with uncertainty and imprecision effectively. What is more, specifications of guide points increased operator workload, and in some cases mistakes took place.

On the contrary, as Fig. 10c shows, with the proposed CDGA* planner, human knowledge is embedded into the cooperative strategy, and human operator took the role of mission manager response for strategic guidance. Once a "warning state" arouse, what the human should do was choose a pass azimuth. Based on real-time battlefield situation, a guide field automatically produced by cloud model based uncertainty reasoning was located at the proper position, effectively implementing human knowledge to guide CDGA* planner to avoid map dead-ends and plan a path satisfying mission objectives.

Table 2 shows performance comparisons of these three methods. The second and third columns of Table 2 represent the overall time consumption and the time that the human operator spent in creating the path, respectively. The second line represents the average results of the 16 participants in the path planning tasks. Path cost errors were calculated by normalizing a path cost to the cost of the path returned by automated A* planner. It can be observed that there are no significant differences among total costs of these paths, and the proposed method can significantly reduce time and resource consumptions

Table 2 Performance comparisons of the three methods

Algorithm	Time (s)	HTime (s)	Size (open)	Size (close)	Path cost	Cost error	J_{length} (Km)	J_{smmoth}	Workload (Utilization)
A*	110.26	0.00	10746	3210	126.58	1.00	355.52	0.89	0.00%
DGA*	11.47	6.30	2074	112	129.69	1.02	359.77	1.00	54.93%
CDGA*	7.34	3.25	1756	94	129.07	1.01	357.35	1.02	44.28%

compared to automated A* planner. Compared with the human-machine cooperative strategy by using operator-specified guide points, the CDGA* planner can not only allow humans to express their decisions more freely and adequately, but also reduce operator workload and enhance operator situational awareness.

To validate the performance of the proposed method in large and complicated environments, three comparative experiments with automated A* planner were conducted. In all these cases representative of most likely scenarios, the map sizes are 4000 × 4000 or bigger.

The first case was to test the influences of human's preferences on the resultant paths, and Fig. 11a displays the path solutions. Path A_2 was planned by automated A* planner. Path A_1 was produced with the preferred turn of left, while path A_3 was computed with the preferred turn of right in the CDGA* planning process. With the CDGA* planner, there can be a certain safe distance determined by real-time battlefield situations away from the threat zone, bringing path A_1 a higher dynamic threat avoidance performance than path A_2. Note that the automated A* planner found the resultant path A_2 in an area populated by high-lethality threat zones, though it's the minimal cost path, in human operator's opinion, it may not be applicable in some specific mission scenarios and is not the satisfying path. Although cost of path A_3 is a little more than A_1, it is in a less hazardous threat environment, thus it can be a better choice than path A_1 in some cases. Therefore, one of the advantages of the proposed method is that it can effectively implement human knowledge and produce good solutions suitable for real applications.

The second test case was of scattered threat zones located complexly between the start and destination positions. The map and paths traced by the two planners are given in Fig. 11b. In this case as well, the CDGA* planner could smoothly find paths towards the destination, and effectively avoid threat zones under the guidance of human's high-level knowledge. In this figure, the three lines A_1, A_2 and A_4 were provided by the CDGA* planner under different human preferences, while path A_3 was produced by the automated A* planner.

Table 3 presents comparisons of the two methods. The third and fourth column of Table 3 represent the overall time consumption and the time human spent, respectively. Similarly, the three lines of the third row depict characteristics of the three paths produced by CDGA* planner of case 2.

Table 3 indicates that CDGA* planner is much faster than automated A* planner and there are no significant differences among costs of the paths returned by these two planners. For the reason that the CDGA* planner can effectively exploit the heuristic information provided by guide points to judge search direction and circumvent threat zones at proper positions, and the trouble that the search engine get trapped in local dead-end areas is avoided. The time human the operator spent accounted for a small part of the overall time consumption of each path, enabling human operator to effectively supervise the search process and promote situation awareness. But what should be noted is that cost of the path planned by automated A* planner is smaller than the path provided by our method, since it's the global optimal solution. In the CDGA* planning method, feasible search space may be pruned by guide fields, and the resultant path is not the global optimal.

It should be noted that in combat environments, definition of optimal is a constantly changing concept.

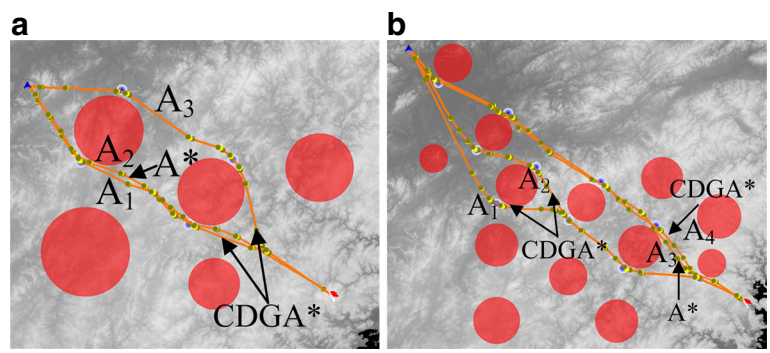

Fig. 11 Comparison of results produced by automated A* planner and CDGA* planner: **a** results on case 1; **b** results on case 2

Table 3 Performance comparisons of two methods on the three cases

Ca-se	Time (A*)	Time (CDGA*)	HTime (CDGA*)	J_{length} (A*)	J_{length} (CDGA*)	Cost (A*)	Cost (CDGA*)	Cost error (CDGA*)	J_{smmoth} (A*)	J_{smmoth} (CDGA*)
1	167.76	17.57	3.01	417.96	415.54	172.63	173.89	1.01	2.59	2.04
		20.37	2.82		431.80		179.65	1.03		2.81
2	259.93	37.39	5.27	494.51	529.38	164.83	173.07	1.05	3.58	3.89
		48.60	6.71		517.47		168.13	1.02		4.10
		39.57	7.68		510.87		169.27	1.03		3.24
3	186.53	14.45	3.41	383.26	384.74	129.89	131.51	1.01	1.53	1.45

Particularly in situations with uncertainty, having a solution good enough and quickly reached is generally preferable to one that requires extended periods of time. Compared with automated A* planer, the proposed method can significantly reduce computational time and produce solutions better reflecting mission scenarios and personal preferences, and therefore, the new algorithm has better search ability, robustness and proved computational efficiency without compromising path performance.

To validate performance of CDGA* planner in expressing uncertainty and human knowledge implementation, we compared it with the fuzzy logic based DGA* planner (FDGA*). In this experiment, lethality degree of threat 1 is 6.8. Lethality degrees of threat 2, threat 3 and threat 4 are 7.5. We tried to control parameter settings and reduce the effects induced by different parameters. In the FDGA* method, the Gaussian membership functions have the same parameters as the normal clouds in the CDGA* method, and the same rule base is adopted. Fig. 12 shows the linguistic terms (VL, L, M, H, VH) for representing threat degree and their corresponding membership functions.

Compared Fig. 12 with Fig. 8, we can observe that fuzzy sets can not describe randomness associated with membership grade. Normal clouds integrate the fuzziness and randomness of qualitative concepts to represent the knowledge, and the super entropy H*e* of a normal cloud expresses the randomness of the membership degree. In the fuzzy reasoning process of FDGA* method, the multiplication operator is used to obtain strength of the rule [18], the minimum operator is used for implication, the maximum operator is used for aggregation, and the center of gravity defuzzification method is adopted to obtain a safety distance [43]. These operators and defuzzification method are widely used in the literature [43, 44].

According to the different principles of the cloud model and fuzzy theory, safety distance of the fuzzy reasoning corresponding to the same input is always the same without randomness. Therefore, FDGA* planner ignores the randomness existing in the causality of the human reasoning. Safety distance to threat 1 is 6.35 Km, and safety distance to threat 2 is 8.43 Km provided by fuzzy reasoning. As shown in Fig. 13a, path B_1 was produced by the automated A* planner, while the three lines B_2, B_3 and B_4 were provided by

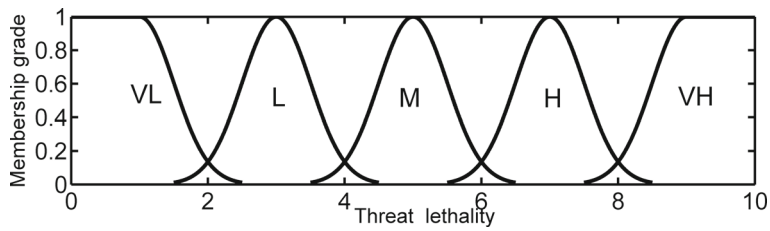

Fig. 12 Linguistic words representing threat degree and their corresponding membership functions

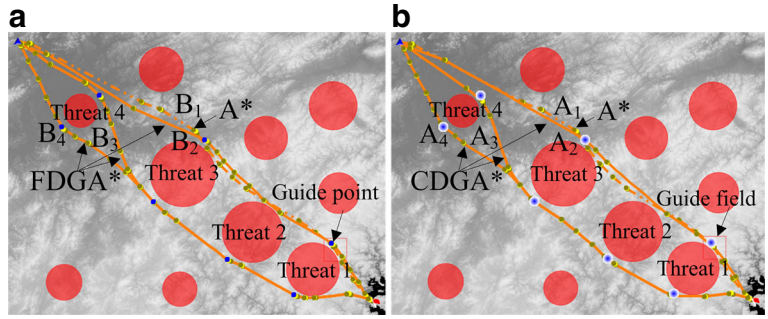

Fig. 13 Comparison of results produced by automated A* planner, FDGA* planner, and CDGA* planner: **a** results produced by automated A* planner and FDGA* planner; **b** results produced by automated A* planner and CDGA* planner

the FDGA* planner (in this experiment, guide radius associated with each guide point produced by fuzzy reasoning was set to 90 m). As shown in Fig. 13b, safety distance to threat 1 is 6.50 Km, and safety distance to threat 2 is 8.60 Km produced by cloud model based uncertainty reasoning. Path A_1 was produced by the automated A* planner, while the three lines A_2, A_3 and A_4 were provided by CDGA* planner. It can be observed that there are few differences between these safety distances provided by these two different reasoning strategies. However, uncertainty reasoning based on cloud model considers both fuzziness of the traditional fuzzy logic and randomness of probability reasoning. The parameter H*e* can reflect the uncertain reasoning process of the driving factors to the location of guide point. Reasoning results of the uncertainty reasoning strategy are random, effectively reflecting human's uncertainty about locations of the guide points, and therefore, they are more reasonable and consistency with human thinking. The fuzzy reasoning strategy could be treated as an approximation to the uncertainty reasoning strategy, and the latter can achieve high levels of simplicity and robustness [20]. Table 4 presents comparisons of A* planner, FDGA* planner and CDGA* planner.

From Table 4, we can observe that costs of the paths provided by CDGA* planner are smaller than the corresponding paths provided by the FDGA* planner, and the CDGA* planner can be faster than FDGA* planner. The CDGA* planner can provide a bigger search space and costs of its resultant path solution could be lower. With respect to the reason for the CDGA* planner could be faster than FDGA*, one reason may be that generating a path passing through a predefined midcourse waypoints in obstacle rich environments may requires more time than generating a path passing through guide fields due to turn angle constraint. Therefore, the CDGA* method can produce a more robust search space. It provides a better way in implementing human knowledge and is more effective than the FDGA* method.

4.3 Discussions

The proposed method can also work well in the presence of static, pop-up, and dynamic obstacles and threats. It has been pointed out that SAS path planner works very well in dynamic environments [4, 27]. The DGA* planner is based on SAS path planner, and dynamic interactions with human operator is well

Table 4 Performance comparisons of three methods

Algorithm	Time(s)	HTime(s)	Size (open)	Size (close)	Path cost	Cost error	J_{length} (Km)	J_{smmoth}
A*	145.45	0.00	13454	4327	228.62	1.00	523.28	2.46
FDGA*	35.17	3.10	9952	496	233.21	1.02	524.73	2.64
	58.89	06.24	12710	725	240.07	1.05	553.37	3.44
	42.56	6.10	10537	533	238.72	1.05	548.62	3.46
CDGA*	29.20	3.04	5162	339	230.28	1.01	522.05	2.30
	44.02	6.26	11401	634	238.81	1.05	544.27	3.01
	33.54	6.15	8198	518	236.93	1.04	543.59	3.27

admissible. Therefore, the proposed method is suitable for real time path planning in dynamic environments.

5 Conclusion

This paper develops an effective human-in-the-loop path planning strategy combining human's intelligence with optimization algorithm's computational speed, and proposes a DGA* path planner based on cloud model. The developed DGA* planner can effectively utilize the heuristic information provided by guide points to judge search direction, and return the optimal path passing through guide fields. Using online reasoning based on cloud model to generate instant guide fields, human's judgments, experience, and high level knowledge are effectively implemented to guide the DGA* planner to avoid map dead-ends, fulfilling the real-time requirements of path planning and providing a new idea in the research of path planning.

As comparative experiments demonstrate, the proposed method is much more effective than the automated A* planner, and it can obtain good solutions reflecting mission scenarios. Compared with the human-machine cooperative strategy by using operator-specified guide points, the proposed method can implement human knowledge more effectively, and require less human workload. Most crucially, paths generated by the proposed method under different complex battlefield circumstances can realize dynamic threat avoidance and reflect personal preferences.

As for future research, a possible direction could be further study of the analytic form of the hybrid process. Design of the human-machine cooperative path planning system also has many issues to resolve: how much human-in-the-loop interactions should be required or permitted, and how intelligently to introduce human intelligence to reduce risks caused by human mistakes. Moreover, how to make computer algorithms have intuitive decision making ability as humans do should also be further studied, may be some research can be done on machine learning or so to solve such problems.

Acknowledgments The authors would like to thank Ding, M. for his assistance and invaluable insights. The authors would also like to thank Zhou, C., Xiao, J., Fu, Y., Li, S., and Yang, X. for their help.

References

1. Zheng, C., Li, L., Xu, F., Sun, F., Ding, M.: Evolutionary route planner for unmanned Air vehicles. IEEE Trans. Robot. **21**, 609–620 (2005)
2. Goerzen, C., Kong, Z., Mettler, B.: A survey of motion planning algorithms from the perspective of autonomous UAV guidance. J. Intell. Robot. Syst. **57**, 65–100 (2010)
3. Kalra, N., Ferguson, D., Stentz, A.: Incremental reconstruction of generalized voronoi diagrams on grids. Robot. Auton. Syst. **57**, 123–128 (2009)
4. Szczerba, R.J., Galkowski, P., Glicktein, I., Ternullo, N.: Robust algorithm for real-time route planning. IEEE Trans. Aerosp. Electron. Syst. **36**, 869–878 (2000)
5. Lee, J., Nam, Y., Hong, S., Cho, W.: New potential functions with random force algorithms using potential field method. J. Intell. Robot. Syst. **66**, 303–319 (2012)
6. Petřek, M., Otyepka, M., Banáš, P., Košinová, P., Koča, J., Damborský, J.: CAVER: a new tool to explore routes from protein clefts, pockets and cavities. BMC Bioinforma. **316**, 1–9 (2006)
7. Lu, Y., Huo, X., Arslan, O., Tsiotras, P.: Incremental multiscale search algorithm for dynamic path planning with low worst-case complexity. IEEE Trans. Syst. Man, Cybern. B, Cybern. **41**, 1556–1570 (2011)
8. Kott, A., Budd, R., Ground, L., Rebbapragada, L., Langston, J.: Building a tool for battle planning: challenges, tradeoffs, and experimental findings. Appl. Intell. **23**(3), 165–189 (2005)
9. Gorodetsky, V., Karsaev, O., Samoylov, V., Serebryakov, S.: Agent-based distributed decision-making in dynamic operational environments. Intell. Dec. Technol. **3**(1), 35–57 (2009)
10. Maza, I., Kondak, K., Bernard, M., Ollero, A.: Multi-UAV cooperation and control for load transportation and deployment. J. Intell. Robot. Syst. **57**(1–4), 417–449 (2010)
11. Stouch, D.W., Zeidman, E., Callahan, W., McGraw, K.: Dynamic Replanning on Demand of UAS Constellations Performing ISR Missions. In: SPIE Defense, Security, and Sensing 2011
12. Griner, A.: Human-RRT collaboration in Unmanned Aerial Vehicle mission path planning. Master of Engineering Thesis, Massachusetts Institute of Technology, Cambridge, MA (2012)
13. Cummings, M., Marquez, J., Roy, N.: Human-automated Path Planning Optimization and Decision Support. Int. J. Hum-comput., 116–128 (2011)
14. Boškovic, J.D., Knoebel, N., Moshtagh, N., Amin, J., Larson, G.L.: Collaborative Mission Planning & Autonomous Control Technology (CoMPACT) System Employing Swarms of UAVs. In: AIAA Guidance, Navigation, and Control Conference, Chicago (2009)
15. Clare, A.S., Maere, P.C., Cummings, M.L.: Assessing operator strategies for real-time replanning of multiple unmanned vehicles. Intell. Dec. Technol. **6**(3), 221–231 (2012)
16. Clare, A.S., Cummings, M.L., How, J.P., Whitten, A.K., Toupet, O.: Operator object function guidance for a real-time unmanned vehicle scheduling algorithm. J. Aeros. Comp. Inf. Com. **9**, 161–173 (2012)

17. Badillo, A.R., Ruiz, J.J., Cotta, C., Fernández-Leiva, A.J.: On user-centric memetic algorithms. Soft. Comput. **17**, 285–300 (2013)
18. Sabo, C., Cohen, K.: Fuzzy logic unmanned air vehicle motion planning. Adv. Fuzzy. Syst. **989051**, 1–14 (2012)
19. Fu, C., Olivares-Mendez, M., Suarez-Fernandez, R., Campoy, P.: Monocular visual-inertial SLAM-Based collision avoidance strategy for fail-safe UAV using fuzzy logic controllers. J. Intell. Robot. Syst., 1–21 (2013)
20. Li, D., Cheung, X., Shi, V.N.g.: Uncertainty reasoning based on cloud models in controllers. Comput. Math. Appl. **35**, 99–123 (1998)
21. Jiang, Y., Jiang, J., Zhang, Y.: A novel fuzzy multiobjective model using adaptive genetic algorithm based on cloud theory for service restoration of shipboard power systems. IEEE Trans. Power Syst. **27**, 612–620 (2012)
22. Zhang, L., Wu, X., Ding, L., Skibniewski, M.J.: A novel model for risk assessment of adjacent buildings in tunneling environments. Build. Environ. **65**, 185–194 (2013)
23. Zhou, Z.: Cognition and removal of impulse noise with uncertainty. IEEE Trans. Image Process. **21**, 3157–3167 (2012)
24. Koenig, S., Likhachev, M.: Fast replanning for navigation in unknown terrain. IEEE Trans. Robot. **21**(3), 354–363 (2005)
25. Breimyer, P., Wurman, P.R.: PBA*: using proactive search to make A* robust to unplanned deviations. In: Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (2008)
26. Cummings, M., Bruni, S.: Collaborative human-computer decision making in network centric warfare. Int. J. Aviat. Psychol. **1**, 17 (2005)
27. Zheng, C., Xu, F., Hu, X., Sun, F., Yan, P.: Online route planner for unmanned air vehicle navigation in unknown battlefield environment. In: IMACS Multiconference on Computational Engineering in Systems Applications, IEEE, pp. 814–818 (2006)
28. Duan, H., Yu, Y., Zhang, X., Shao, S.: Three-dimension path planning for UCAV using hybrid meta-heuristic ACO-DE algorithm. Simul. Model. Pract. **18**, 1104–1115 (2010)
29. Xu, C., Duan, H., Liu, F.: Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path planning. Aerosol. Sci. Tech. **14**, 535–541 (2010)
30. Besada-Portas, E., de la Torre, L., de la Cruz, J.M., de Andrés-Toro, B.: Evolutionary trajectory planner for multiple UAVs in realistic scenarios. IEEE Trans. Robot. **26**, 619–634 (2010)
31. Dechter, R., Pearl, J.: Generalized best-first search strategies and the optimality of A*. J. ACM (JACM) **32**, 505–536 (1985)
32. Cummings, M.L., Mitchell, P.J.: Predicting Controller Capacity in Supervisory Control of Multiple UAVs. Systems and Humans. IEEE Trans. Syst. Man, Cybern. A, Syst. Hum. **38**, 451–460 (2008)
33. Marler, R.T., Arora, J.S.: The Weighted Sum Method for Multi-objective Optimization: New Insights. Struct. Multidiscip. Optim. **41**, 853–862 (2010)
34. Hanson, M.L., Roth, E., Hopkins, C.M., Mancuso, V.: Developing mixed-initiative interaction with intelligent systems: Lessons Learned from Supervising Multiple UAVs. In: AIAA 1st Intelligent Systems Technical Conference, Chicago (2004)
35. Shanmugavel, M., Tsourdos, A., White, B., Zbikowski, R.: Co-operative path planning of multiple UAVs using Dubins paths with clothoid arcs. Control Eng. Pract. **18**, 1084–1092 (2010)
36. Sun, X., Cai, C.: Human-machine cooperation in unmanned aerial vehicle path planning based on cloud model, MIPPR 2011: Automatic Target Recognition and Image Analysis, 4-6 Nov. 2011. SPIE - The International Society for Optical Engineering, USA, 1–8 (2011)
37. Liang, Q., Cheng, X.: KUPS: knowledge-based ubiquitous and persistent sensor networks for threat assessment. IEEE Trans. Aero. Elec. Sys. **44**(3), 1060–1069 (2008)
38. Lee, D.S., Gonzalez, L.F., Srinivas, K., Periaux, J.: Robust evolutionary algorithms for UAV/UCAV aerodynamic and rcs design optimisation. Comput. Fluids. **37**, 547–564 (2008)
39. Lingxiao, W., Deyun, Z.: Effective path planning method for low detectable aircraft. J. Syst. Eng. Electron. **20**, 784–789 (2009)
40. Wang, H., He, S., Liu, X., Dai, L., Pan, P., Hong, S., Zhang, W.: Simulating urban expansion using a cloud-based cellular automata model: a case study of Jiangxia, Wuhan. China. Landsc. Urban. Plan **110**, 99–112 (2012)
41. Li, H., Guo, C.: Piecewise cloud approximation for time series mining. Knowl.-based. Syst. **24**, 492–500 (2011)
42. Cummings, M.L., Clare, A., Hart, C.: The role of human-automation consensus in multiple unmanned vehicle scheduling. Hum. Factors: J. Hum. Factors. Ergon. Soc. **52**, 17–27 (2010)
43. Esposito, M., De Pietro, G.: An ontology-based fuzzy decision support system for multiple sclerosis. Eng. Appl. Artif. Intel. **24**, 1340–1354 (2011)
44. Xu, Z., Gao, K., Khoshgoftaar, T.M., Seliya, N.: System regression test planning with a fuzzy expert system. Inform. Sci. **259**, 532–543 (2014)