Integrated Computer-Aided Engineering 14 (2007) 141–159
IOS Press

Autonomous and cooperative robotic behavior based on fuzzy logic and genetic programming

James F. Smith III* and ThanhVu H. Nguyen
Code 5741, Naval Research Laboratory, Washington, DC, 20375-5320, USA

Abstract. Advances in a fuzzy decision theory that allow automatic cooperation between unmanned aerial vehicles (UAVs) are discussed. The algorithms determine points the UAVs are to sample, flight paths, and the optimal UAVs for the task and related changes during the mission. Human intervention is not required after the mission begins. The algorithms take into account what is known before and during the mission about UAV reliability, fuel, and kinematics as well as the measurement space's meteorological states, terrain, air traffic, threats and related uncertainties. The fuzzy decision tree for path assignment is a significant advance over an older fuzzy decision rule that was previously introduced. Simulations show the ability of the control algorithm to allow UAVs to effectively cooperate to increase the UAV team's likelihood of successfully measuring the atmospheric index of refraction over a large volume. A genetic program (GP) based data mining procedure is discussed for automatically evolving fuzzy decision trees. The GP is used to automatically create the fuzzy decision tree for real-time UAV path assignments. The GP based procedure offers several significant advances over previously introduced GP based data mining procedures. These advances help produce mathematically concise fuzzy decision trees that are consistent with human intuition.

1. Introduction

Autonomous cooperative teams of robots will be used for many applications in the near future. Fundamental to this process will be mission planning prior to the mission and algorithms for automatic control and cooperation of the robots in real-time during the mission.

In the following, algorithms based on fuzzy logic are described that can be used to plan missions for a coordinated team of flying robots. The robotic unmanned aerial vehicles (UAVs) will be rendered autonomous. Human intervention is not required after the mission begins when the algorithms described below are used.

The UAVs' goal is to cooperate to measure the atmospheric index of refraction. The fuzzy mission planning algorithm uses human expertise to determine the points to be sampled, the points to avoid, the best flight paths for sampling and a small number of the best UAVs to conduct the mission.

The planning algorithm takes into account the input properties of each UAV under consideration including: the UAVs risk tolerance, i.e., how much risk the UAVs owner will allow it to experience, expert estimates of the UAV's sensor and non-sensor system reliabilities; and the amount of fuel the UAV carries. The planning algorithm also takes into account what is known about the atmospheric volume where measurements are to be made, i.e., the measurement space prior to the mission. This information includes weather, e.g., rain, turbulence, and icing conditions; physical obstructions such as mountains, and high tension wires; enemy behavior; air traffic such as civilian or military aircraft or animal life. This information is used to form a measure of risk, using a fuzzy risk tree. The risk tree allows a simple mathematical formulation of the mission risk.

Also, the planning algorithm takes into account electromagnetic propagation [4,9] as well as human expertise to determine points in the atmosphere to sample,

*Corresponding author. Tel.: +1 202 767 5358; Fax: +1 202 404 7690; E-mail: james.smith@nrl.navy.mil.

ISSN 1069-2509/07/$17.00 © 2007 – IOS Press and the author(s). All rights reserved

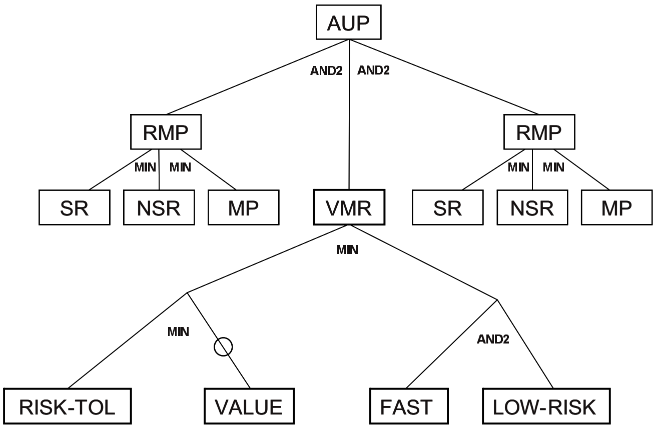

Fig. 1. The AUP subtree.

their priority and a simple mathematical formulation of the mission priority.

A real-time algorithm expressed as a fuzzy decision tree is formulated for assignment of the best UAV to each path. The algorithm that "assigns UAVs to paths" (AUP) is referred to as the AUP fuzzy decision tree and it is depicted in Fig. 1.

The AUP fuzzy decision tree is a simple elegant mathematical formula determining the degree to which each UAV belongs to the path in terms of risk-tolerance, sensor and non-sensor reliability, UAV fuel limitations, mission priority and mission risk.

The predecessor to the AUP fuzzy decision tree was the AUP fuzzy decision rule [20,22,24]. Both the AUP decision rule and AUP decision tree were initially constructed based on human expertise. The AUP fuzzy decision rule had some of the properties of the tree, but was more limited in its decision making ability and subject to certain types of errors that the tree was designed to avoid. The AUP decision tree is a significant advance over the AUP decision rule [22,24].

The AUP fuzzy decision tree is used by both the planning and real-time control algorithm. It can make extremely fast assignments of UAVs to paths, while allowing the various input concepts to remain explicit in the formulation.

Although the AUP decision tree was originally constructed using human expertise, another method for evolving it using a genetic program (GP) [7] as a data mining (DM) function has been created. The GP is capable of recreating the original tree, but has subsequently produced trees that are different but arguably superior in their performance properties. The GP is guided by built in fuzzy logic based on partial expertise and related uncertainty. This procedure results in fuzzy decision trees that are mathematically more concise. The trees also assume a form that is more consistent with human intuition making them easier to understand. By making the tree more concise and easier to understand it is easier to introduce new rules on the trees for the purpose of innovation. Also, concise and intuitive results frequently facilitate validation.

Classical if-then rules have been used in the past to guide GP evolution, notably for the purpose of reverse engineering hardware designs [21,25]. The use of fuzzy rules to guide the GP is a significant advance over using classical if-then rules. Fuzzy logic offers a better way of dealing with the uncertainties associated with partial knowledge.

A GP is a computer program based on the theory of evolution that automatically evolves other computer programs or mathematical expressions [7]. The mathematical expressions evolved here are fuzzy decision trees.

The GP based procedure is a DM technique, a kind of pattern recognition. The GP mines a database of scenarios to produce an optimal tree. An optimal solution is one that maximizes the fitness function. The fitness function is constructed using the database of scenarios. The GP is guided by fuzzy logic to improve the GP's convergence time, to reduce size of the tree and produce elegant mathematical formulations understandable by human beings. Elegance and conciseness of form, and understandability are essential to further innovation. The fuzzy decision trees that are created by the GP based DM procedure are unlike black box algorithms like neural nets where understanding parameter relations is generally out of the question.

Being able to automatically generate decision algorithms using GP based data mining is a significant advance. It is frequently difficult to acquire enough if-then rules from experts to produce a full tree, whereas experts can often provide opinions about the status of a scenario.

Other approaches to creating optimal flight paths with UAV assignments might use a genetic algorithm (GA) or dynamic programming. Both approaches might be suitable for a pre-mission planning algorithm where there is plenty of time for computationally intensive algorithms to run. They are unlikely to be suitable for real-time application, when dealing with slow legacy processors. The AUP fuzzy decision tree requires little CPU time and can produce effective decisions even on slow legacy processors. Also, the one-time decisions made by the GA [8,27] or dynamic programming approaches [1,30] would be represented as numbers, not mathematical expressions. It would be difficult or impossible to extract explicit relationships between input quantities like reliability, risk-tolerance, etc., and output quantities like the ultimate assignment of a UAV to a path.

The planning algorithm makes determinations prior to the mission's execution. Inevitably, during the mission, events will occur that demand changes to the previously determined flight path. The real-time control algorithm allows the UAVs' task to change during the mission. These changes can include alteration in flight path, changing sample points and the need for automatic cooperative behavior between UAVs. As paths change in real-time the AUP fuzzy decision tree is used for reassignment.

A fuzzy decision rule referred to as the priority of helping (PH) decision rule is also provided as part of the control algorithm that allows the UAVs to cooperate automatically in three ways. This automatic cooperation is based on communication; there is no fixed or central command platform: the UAVs automatically self-organize. The information transferred between UAVs is a small number of fuzzy grades of membership, so communication bandwidth requirements are very low.

The PH fuzzy decision rule allows UAVs to collaborate to make atmospheric measurements. It also allows them assist each other when malfunctions are suspected or take over when a malfunction has occurred.

The PH fuzzy decision rule is a simple elegant mathematical relationship between various fuzzy concepts essential to collaboration. It makes explicit relationships that would be impossible to discern using a black box like a neural net [6] or an optimization procedure [1,5, 8,11,12,27,28,30] that delivers only a numerical output like a GA or dynamic programming.

The planning and control algorithms described below could be used for many different cooperative atmospheric measurement processes such as tracking chemical plumes [26], determining properties of atmospheric ducts [4], rain systems, etc. The application that motivated this work was a need to measure the atmospheric index of refraction for the purpose of geo-location [20, 24].

Section 2 discusses the electromagnetic measurement space, UAV risk, and the planning algorithm. Section 2 also discusses the UAV path construction algorithm that determines the minimum number of UAVs required to complete the task, a fuzzy logic based approach for assigning paths to UAVs and which UAVs should be assigned to the overall mission. Finally Section 2 discusses a genetic program based data mining procedure for evolving the fuzzy decision tree for assigning UAVs to paths. Section 3 describes the control algorithm that renders the UAVs autonomous. Section 3 also describes the priority for helping algorithm, a part of the control algorithm based on fuzzy logic that determines which UAV should support another UAV requesting help. The three subclasses of help requests are also discussed in this section. Section 4 discusses experimental results including UAV path determination, UAV path assignment, determination of which UAVs should fly the mission and the result of a request for help during the mission. Section 5 provides conclusions. Finally, Section 6 describes future research directions.

2. Planning, AUP tree and the evolution of logic

The measurement space consists of the electromagnetic propagation environment where UAVs and the IP make their measurements. This environment includes sample points and the desirable neighborhoods that surround them. The sample points or the desirable neighborhoods are where the UAVs will make measurements. The method of determining the sample points and desirable neighborhoods is described below.

The measurement space also includes taboo points and the undesirable neighborhoods that surround them. The taboo points are points of turbulence and other phenomena that could threaten the UAVs. The undesirable neighborhoods surrounding them also represent various degrees of risk. The method of specifying taboo points and quantifying the degree of risk associated with their undesirable neighborhoods employs fuzzy logic and is discussed in this section.

The planning algorithm allows the determination of the minimum number of UAVs needed for the mission subject to fuel constraints, risk, UAV cost, and importance of various points for sampling. Risk refers to turbulent regions or regions undesirable for other reasons, e.g., the presence of enemy observers or physical obstructions. The planning algorithm automatically establishes the order in which to send the UAVs taking into account the UAV's value; onboard sensor payload; onboard resources such as fuel, computer CPU and memory; etc. The priority of sample points and their desirable neighborhoods are taken into account. The planning algorithm also calculates the optimal path around undesirable regions routing the UAVs to or at least near the points to be sampled.

In the planning phase, the location of the electromagnetic source (EMS) is unknown. Some positions are more likely than others for the EMS's location. When establishing likely positions for the EMS, human experts are consulted. The experts provide subjective probabilities of the EMS being located at a number of positions. These likely EMS locations are referred as *hypothesis positions*. Ray-theoretic electromagnetic propagation [4,20] is conducted from each hypothesis position to each interferometer element on the interferometer platform (IP). The points on the sampling grid nearest the points of each ray's passage are the sample points. The priority of a sample point is related to the subjective probability of the hypothesis position from which the associated ray emerges. Sample points arising from the highest probability hypothesis positions have priority one; sample points associated with lower probability hypothesis positions, priority two; etc.

Each sample point is surrounded by what are referred to as *desirable neighborhoods*. Depending on local weather, topography, etc., the desirable neighborhoods are generally concentric closed balls with a degree of desirability assigned to each ball. The degree of desirability characterizes the anticipated variation in the index of refraction. If for that region of the measurement space, the spatial variation of the index of refraction is slow, the degree of desirability may assume its maximum value of unity for a ball of radius measured in miles. For regions of space where the index of refraction's spatial variation is greater, the degree of desirability may fall off much more rapidly, approaching the minimum value of zero after just a mile or two.

The desirable region need not have spherical geometry. Rotational symmetry may be broken by a variety of processes, e.g., an elevated duct, a radio hole, etc.

The notion of a desirable neighborhood is motivated by the fact that a sample point may also be a taboo point or reside within an undesirable neighborhood. In the case the sample point coincides with or is near a taboo point and at least part of the sample point's desirable neighborhood falls within the taboo point's undesirable neighborhood, the UAV may only sample within a desirable neighborhood that is consistent with its risk tolerance.

A point may be labeled taboo for a variety of reasons. A taboo point and the undesirable neighborhoods containing the point generally represent a threat to the UAV. The threat may take the form of high winds, turbulence, icing conditions, mountains, etc. The undesirable neighborhoods around the taboo point relate to how spatially extensive the threat is. A method of quantifying risk and incorporating it into the path assignment algorithm is presented that offers conceptual improvements over an approach previously developed [20]. This section uses fuzzy logic to quantify how much risk a given neighborhood poses for a UAV. This quantitative risk is then incorporated into the UAV's cost for traveling through the neighborhood as described in this section. Once the cost is established an optimization algorithm is used to determine the best path for the UAV to reach its goal.

When determining the optimal path for the UAVs to follow both the planning algorithm and the control algorithm running on each UAV take into account taboo points and the undesirable neighborhood around each taboo point. The path planning algorithm and control algorithm will not allow a UAV to pass through a taboo point. Depending on the UAV's risk tolerance a UAV may pass through various neighborhoods of the taboo point, subsequently experiencing various degrees of risk. Both the concepts of risk and risk tolerance are based on human expertise and employ rules each of which carry a degree of uncertainty. This uncertainty is born of linguistic imprecision [24,29], the inability of human experts to specify a crisp assignment for risk. Owing to this uncertainty it is very effective to specify risk and risk tolerance in terms of fuzzy logic.

2.1. Risk fuzzy decision tree

Risk is represented as a fuzzy decision tree [3,15–19]. The risk subtree defined below and displayed in Fig. 2 is a subtree of the larger risk tree that was actually used. The risk tree is used to define taboo points and the undesirable neighborhoods surrounding the taboo points.

The root concepts on the risk tree use the membership function defined in Eqs (1)–(3),

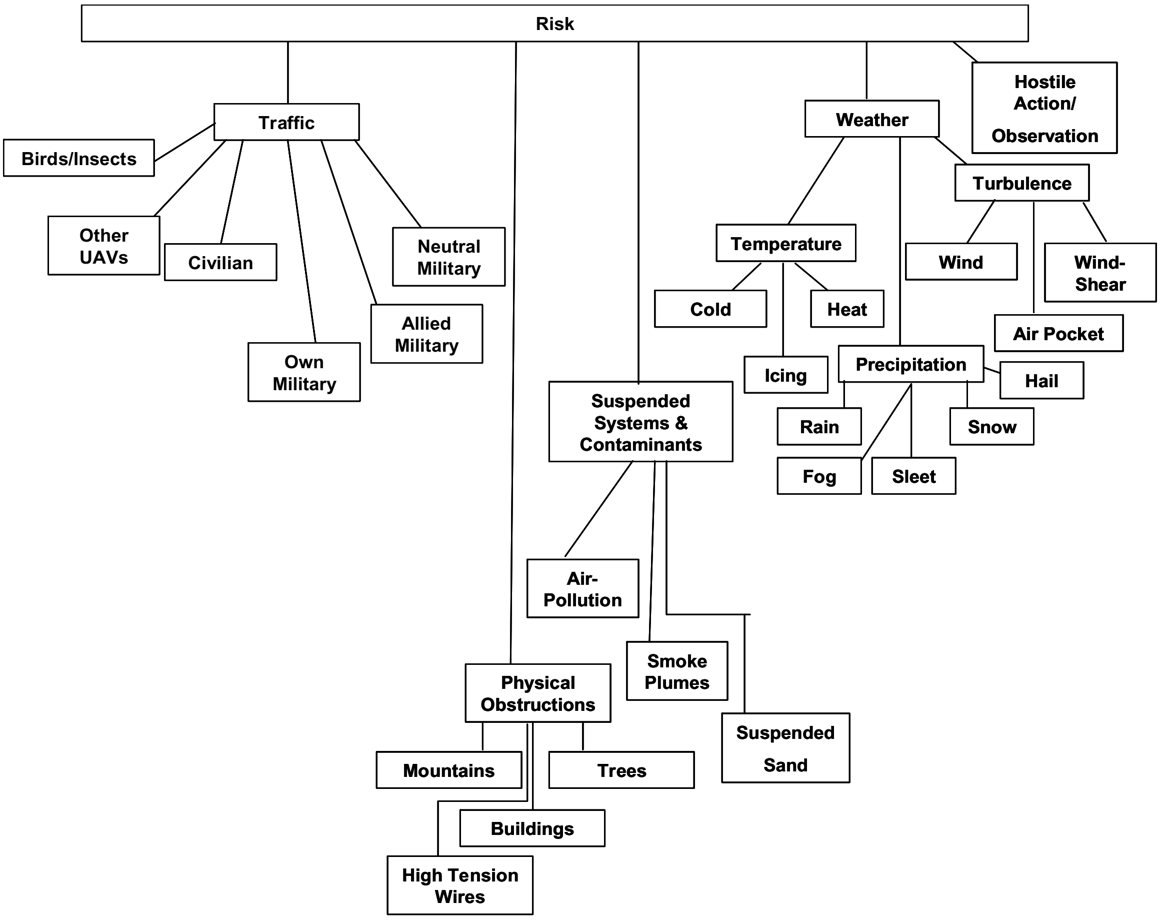

Fig. 2. The fuzzy risk tree and its 25 fuzzy root concepts.

$$\mu_\gamma\left(\vec{q}_{\text{taboo}}, \vec{x}\right) = \begin{cases} 1, if\ r = 0 \\ 3/4, if\ 0 < r \leqslant 1 \cdot \Delta l \\ 1/2, if\ 1 \cdot \Delta l < r \leqslant \sqrt{2} \cdot \Delta l \\ 1/4, if\ \sqrt{2} \cdot \Delta l < r \leqslant \sqrt{3} \cdot \Delta l \\ 0, if\ r > \sqrt{3} \cdot \Delta l \end{cases} \tag{1}$$

$$r = \left\| \vec{x} - \vec{q}_{\text{taboo}} \right\|, \tag{2}$$

$$\vec{q}_{\text{taboo}} = \text{position of taboo point.} \tag{3}$$

where the "taboo point," $\vec{q}_{\text{taboo}}$ is the point at which the risk phenomenon has been observed. The root concepts used on the risk subtree are given in Eq. (4), and the subscript γ is an element of the root concept set, *RC*, i.e.,

$$\gamma \in \mathbf{RC} = \{\text{Mountains, High Tension Wires, Buildings, Trees, Smoke Plumes, Suspended Sand, Birds/Insects, Other UAVs, Air Polution, Civilian, Own Military, Allied Military, Neutral Military, Cold, Heat, Icing, Rain, Fog, Sleet, Snow, Hail, Air Pocket, Wind, Wind Shear, Hostile Action/Observation}\}. \tag{4}$$

The norm in Eq. (2) is typically taken as an Euclidean distance. The quantity Δl for most applications is generally assigned a value of one mile or more. In extreme cases it can be much larger than a mile.

The fuzzy membership function for the composite concept "risk" is defined as

$$\mu_{\text{risk}}\left(\vec{q}_{\text{taboo}}, \vec{x}\right) = \max_{\alpha \in RC} \mu_\alpha\left(\vec{q}_{\text{taboo}}, \vec{x}\right). \tag{5}$$

2.2. Optimal paths and AUP fuzzy decision tree

The best path algorithm is actually an optimization algorithm that attempts to minimize a cost function to determine the optimal trajectory for each UAV to follow, given a priori knowledge. The cost function for the optimization algorithm takes into account various factors associated with the UAV's properties, mission and measurement space. Two significant quantities that contribute to the cost are the effective distance between the initial and final proposed positions of the UAV and the risk associated with travel.

For purposes of determining the optimal path, the UAV is assumed to follow a rectilinear path consisting of connected lines segments, where the beginning and ending points of each line segment reside on the UAV's sampling lattice. Let A and B be two grid points on the UAV's sampling grid with corresponding position vectors, $\vec{r}_A$ and $\vec{r}_B$, respectively. Denote the Euclidean distance between A and B as $d\left(\vec{r}_A, \vec{r}_B\right)$. Let $v\left(\vec{r}_A, \vec{r}_B\right)$ be the speed at which the UAV travels in going from $\vec{r}_A$ to $\vec{r}_B$. If both $\vec{r}_A$ and $\vec{r}_B$ are sample points then the UAV travels at sampling velocity, otherwise it travels at non-sampling velocity. The path cost is given by

$$\text{path\_}\cos t\left(\vec{r}_A, \vec{r}_B\right) = \frac{d(\vec{r}_A,\vec{r}_B)+\beta\cdot\sum\limits_{i=1}^{n_{\text{taboo}}}\mu_{\text{risk}}\left(\vec{t}_i,\vec{r}_B\right)}{v(\vec{r}_A,\vec{r}_B)}. \tag{6}$$

where n_{taboo} is the number of taboo points, i.e., columns in the taboo point matrix

$$\text{Taboo} \equiv \left[\vec{t}_1, \vec{t}_2, \ldots, \vec{t}_{n_{\text{taboo}}}\right] \tag{7}$$

and $\vec{t}_i, i = 1, 2, \ldots, n_{\text{taboo}}$ are the taboo points determined to exists in the measurement space when $\text{path\_}\cos t\left(\vec{r}_A, \vec{r}_B\right)$ is calculated. The number of points in the measurement space is finite so n_{taboo} is finite. The quantity, β, is an expert assigned parameter. Note that $\text{path\_}\cos t\left(\vec{r}_A, \vec{r}_B\right)$ is an effective time. When risk is not present, i.e., $\beta \cdot \sum\limits_{i=1}^{n_{\text{taboo}}} \mu_{\text{risk}}\left(\vec{t}_i, \vec{r}_B\right)$ is zero, then $\text{path\_}\cos t\left(\vec{r}_A, \vec{r}_B\right)$ is the actual travel time. When risk is present then the travel time is increased. The parameter β helps to penalize risky paths, thus decreasing the probability they are selected.

If the candidate path for the mission consists of the following points on the UAV lattice given by the path matrix in Eq. (8),

$$\text{Path}_i = \left[\vec{r}_1, \vec{r}_2, \ldots, \vec{r}_n\right], \tag{8}$$

then the total path cost is defined to be

$$\text{total\_cost}(\text{Path}_i) \equiv \sum_{j=1}^{n-1} \text{path\_cost}(\vec{r}_j, \vec{r}_{j+1}). \tag{9}$$

Determining the optimal path for the i^{th} UAV consists of minimizing the total path cost given by Eq. (9) such that there is enough fuel left to complete the path.

An A-star algorithm [13] is used to determine the point that will ultimately minimize the cost function. A-star is a heuristic algorithm. It was selected because it is relatively fast and can be polynomial in time [13]. Empirically it is significantly faster than the Dijkstra algorithm [13]. It is very effective for cases where a significant percentage of the atmospheric volume does not change over the course of a mission.

The A-star algorithm as implemented in the planning and control algorithms is an easily replaceable module. It may well be replaced by a more sophisticated algorithm in the future [5,11,12,28].

The planning algorithm determines the path each UAV will pursue, which points will be sampled, the minimum number of UAVs required for sampling the points and makes assignments of UAVs for measurements at particular points. UAVs are assigned as a function of their abilities to sample high priority points first. The planning algorithm determines flight paths by assigning as many high priority points to a path as possible taking into account relative distances including sampling and non-sampling velocity, risk from taboo points, and UAV fuel limitations. Once flight paths are determined, the planning algorithm assigns the best UAV to each path using the fuzzy logic decision tree for path assignment described in this section.

The planning algorithm must assign UAVs to the flight paths determined by the optimization procedure described below in this section. This is referred to as the UAV path assignment problem (UPAP). The planning algorithm makes this assignment using the following fuzzy logic based procedure. To describe the decision tree it is necessary to develop some preliminary concepts and notation.

Each UAV will fly from lattice point to lattice point, i.e., grid point to grid point, let one such route be given by the matrix of points,

$$\text{Path} = \left[\vec{P}_1, \vec{P}_2, \ldots, \vec{P}_{n_{\text{path}}}, \vec{P}_1\right] \tag{10}$$

where the ordering of points gives the direction of the route, i.e., starting at $\vec{P}_1$ and ending at $\vec{P}_1$. Let the taboo points be those given in Eq. (7). Let the degree of undesirability of the neighborhood associated with taboo points, $\vec{t}_i, i = 1, 2, \ldots, n_{\text{taboo}}$ be denoted $\mu_{\text{risk}}\left(\vec{t}_i, \vec{P}_j\right)$ for the route points $\vec{P}_j, j = 1, 2, \ldots, n_{\text{path}}$. The definition of the mission risk (MR) is

$$\text{mission\_risk}\left(\text{Taboo,Path}_k\right) \equiv \sum_{i=1}^{n_{\text{taboo}}} \sum_{j=1}^{n_{\text{path}}} \mu_{\text{risk}}\left(\vec{t}_i, \vec{P}_j\right) \tag{11}$$

The degree to which the k^{th} path belongs to the related fuzzy concept *MR* is given by

$$\mu_{MR}\left(\text{Taboo,Path}_k\right) \equiv \frac{\text{mission\_risk}\left(\text{Taboo,Path}_k\right)}{\max_j \left\{\text{mission\_risk}\left(\text{Taboo,Path}_j\right)\right\}} \tag{12}$$

The "*max*" operation in Eq. (12) is taken over the set of all possible UAVs that can be assigned to the mission.

A fuzzy concept related to "mission risk" is "low risk." The fuzzy membership function for "low risk" denoted as μ_{LR} is defined as

$$\mu_{LR}\left(\text{Taboo,Path}_k\right) \equiv \min\left(1, \alpha + 1 - \mu_{MR}\right) \tag{13}$$

where $\alpha \in (0, 1)$ is an expert defined parameter. The function of α is to make sure that "low risk" does not dominate calculations developed below.

Within the path specified by Eq. (10), let there be the following sample points to be measured, $\vec{S}_j, j = 1, 2, \ldots, n_{sp}$. Let the function $prio$ assign priorities to the sample points, i.e, prio($\vec{S}_j$) is the priority of the j^{th} sample point. The values that prio($\vec{S}_j$) can take are positive integers with one representing the highest priority, two the next highest priority, etc. The mission priority (MP) for the k^{th}Path$_k$ is defined to be

$$\text{mission\_prio}\left(\text{Path}_k\right) \equiv \sum_{i=1}^{n_{sp}} \frac{1}{\text{prio}\left(\vec{S}_i\right)}. \tag{14}$$

The degree to which the k^{th} path belongs to the related fuzzy concept *MP* is given by

$$\mu_{MP}\left(\text{Path}_k\right) \equiv \frac{\text{mission\_prio}\left(\text{Path}_k\right)}{\max_j \left\{\text{mission\_prio}\left(\text{Path}_j\right)\right\}} \tag{15}$$

The fuzzy degree of reliability experts assign to the sensors of UAV(i) is denoted as $\mu_{sr}\left(UAV\left(i\right)\right)$. This is a real number between zero and one with one implying the sensors are very reliable and zero that they are totally unreliable. Likewise, $\mu_{nsr}\left(UAV\left(i\right)\right)$ is the fuzzy degree of reliability of other non-sensor systems onboard the UAV(i). This fuzzy concept relates to any non-sensor system, e.g., propulsion, computers, hard disk, deicing systems, etc. The value of UAV(i) in units of \$1000.00 is denoted as $V\left(UAV\left(i\right)\right)$. The amount of fuel that UAV(i) has at time t is denoted $fuel\left(UAV\left(i\right), t\right)$. All the UAVs participating in a mission are assumed to leave base at time, $t = t_o$.

Let UAV(i)'s fuzzy grade of membership in the fuzzy concept "risk tolerance" be denoted as $\mu_{\text{risk-tol}}$ $\left(UAV\left(i\right)\right)$. The quantity, $\mu_{\text{risk-tol}}\left(UAV\left(i\right)\right)$, is a number between zero and one and will be simply referred to as UAV(i)'s risk tolerance. If the risk tolerance is near zero then the UAV should not be sent on very risky missions. If the UAV's risk tolerance is near one then it can be sent on very risky missions. It seems natural to compare "risk tolerance" to "Value." So the comparison can be carried out on the same footing, a fuzzy concept of value should be defined.

The fuzzy grade of membership of each UAV that can be assigned to the mission in the fuzzy concept "Value" is defined as

$$\mu_V\left(UAV\left(i\right)\right) \equiv \frac{\varepsilon_V \cdot \text{Value}\left(UAV\left(i\right)\right)}{\max_j \left\{\text{Value}\left(UAV\left(j\right)\right)\right\}} \tag{16}$$

The quantity ε_V is an expert assigned value. It is a number between zero and one that insures that the most valuable UAV can still be assigned to a path. The motivation for defining this parameter will be clearer after the AUP subtree is defined below, since it can be observed that if ε_V is zero, then for the most valuable UAV in the team, μ_{AUP} will take the value zero. This would prevent it from being assigned a path which is undesirable.

The advantage of the concept of "risk tolerance" is that it gives the user an extra concept to exploit. If the UAV is not of great relative value, but it still might be needed for a crucial mission after the current one, it might be useful to give it a low risk tolerance so that it is not lost on the current mission. This may allow it to be used on a subsequent mission.

Another fuzzy concept and related fuzzy membership function that will be defined is "fast." A UAV is said to be fast if it permitted to pursue a particular path and the time it would take to complete the path is small. It is not allowed to travel the path unless the UAV's fuel level, reliability, risk-tolerance and mission priority exceed certain tolerances.

Let the $T\left(UAV\left(i\right), \text{Path}\right)$ be the amount of time it will take UAV(i) to fly and make measurements along Path. The fuzzy membership function for the concept "fast" is defined as follows:

$$\mu_{\text{fast}}\left(UAV\left(i\right),Path\right)\equiv\Lambda_{rrtmp}\cdot \tag{17}$$
$$\min\left(1,\alpha+1-\frac{T\left(UAV\left(i\right),\text{Path}\right)}{\max\limits_{j}\left\{T\left(UAV\left(j\right),\text{Path}\right)\right\}}\right)$$

and

$$\begin{aligned}\Lambda_{rrtmp}\equiv\chi&\left[\text{fuel}\left(UAV\left(i\right),t_o\right)+\varepsilon_{\text{fuel}}\right.\\&\left.-T\left(UAV\left(i\right),\text{Path}\right)\right]\cdot\\&\chi\left[\min\left(\mu_{sr},\mu_{nsr}\right)-\varepsilon_{1,rel}\cdot\right.\\&\min(1-\mu_{\text{risk}-\text{tol}},\\&\left.\max(1-\mu_{MP},\varepsilon_{2,MP}))-\varepsilon_{3,rel}\right]\end{aligned}\tag{18}$$

where $\varepsilon_{1,rel},\varepsilon_{2,MP},\varepsilon_{3,rel},\varepsilon_{\text{fuel}}\in(0,1]$ are expert assigned parameters. The Heaviside step functions denoted as χ in Eq. (18) takes the value one when its argument is greater than or equal to zero and is zero otherwise.

The factor in Eq. (18) determines whether the UAV will be permitted to travel the path at all. If its amount of fuel, sensor and non-sensor reliabilities, risk-tolerance and mission priorities do not exceed certain expert defined tolerances it will not be permitted to travel and as such it will certainly not be "fast."

The term $\varepsilon_{1,rel}\cdot\min(1-\mu_{\text{risk}-\text{tol}},\max(1-\mu_{MP},\varepsilon_{2,MP}))$ in the Heaviside step function's argument in Eq. (18) can result in Λ_{rrtmp}going to zero if $\mu_{\text{risk}-\text{tol}}$ or μ_{MP} are small enough.

The parameter α is selected so that μ_{fast} in Eq. (17) does not go to zero for the UAV that takes the longest time to navigate Path. By preventing μ_{fast} from going to zero in this case, the slowest UAV can be selected if it can complete the path and its grades of membership in the other fuzzy concepts found in Eq. (17) are high enough.

If "Risk tolerance" and "mission priority" take low values then depending on the value of $\varepsilon_{1,rel}$, the membership function for the fuzzy concept "fast" may take the value zero. The parameter $\varepsilon_{2,MP}$ limits the effect of "mission priority." Even if the mission priority is very high, risk tolerance plays an important role. If the UAV has high risk tolerance and the path, high mission priority the UAV must have a minimum reliability given by $\varepsilon_{3,rel}$. Finally, the motivation for the concept "fast" is that a fast UAV experiences a lower relative risk since it is in the field less time and may be exposed to risk for a shorter duration.

A fuzzy concept that combines "Value" and "mission risk" is "VMR" and its membership function denoted as μ_{VMR} is defined as

$$\begin{aligned}\mu_{VMR}\equiv\min&\left[\min\left(\mu_{\text{risk}-\text{tol}},1-\mu_V\right),\right.\\&\left.AND_2\left(\mu_{\text{fast}},\mu_{LR}\right)\right]\end{aligned}\tag{19}$$

The use of AND_2 in Eq. (19) allows distinctions to be made between various values of μ_{fast} and μ_{LR}. If AND_2 were replaced by a $\min$ in Eq. (19) then if μ_{fast} is low enough then $\min\left(\mu_{\text{fast}},\mu_{LR}\right)$ would take the value μ_{fast} independent of the value of μ_{LR} this would not allow fine distinctions to be made.

The logical connective AND_2 is defined as

$$AND_2\left(\mu_A,\mu_B\right)\equiv\mu_A\cdot\mu_B \tag{20}$$

The fuzzy concept "RMP" combines the fuzzy concepts sensor reliability (sr) non-sensor-reliability (nsr) and "MP." The fuzzy membership function for "RMP," denoted as μ_{RMP} is defined as

$$\mu_{RMP}\equiv\min\left(\mu_{sr},\mu_{nsr},\mu_{MP}\right). \tag{21}$$

Table 1 provides an example of the application of Eq. (21) for a three UAV scenario.

Both the membership functions for "VMR" and "RMP" can be represented as fuzzy decision trees.

Finally, the fuzzy membership function for the fuzzy concept "assignment of UAV(i) to the path" (AUP) is defined as

$$\begin{aligned}\mu_{AUP}\equiv{}&AND_2[\mu_{RMP},\\&AND_2\left(\mu_{RMP},\mu_{VMR}\right)]\\={}&\mu_{RMP}^2\cdot\mu_{VMR}\end{aligned}\tag{22}$$

The fuzzy membership function for AUP is a decision tree that combines both "VMR" and "RMP" as subtrees. The use of AND_2 in Eq. (22) in two places renders μ_{AUP} more sensitive to the values of μ_{RMP} and μ_{VMR} than it would be if the membership function for AUP took the value $\min\left(\mu_{RMP},\mu_{VMR}\right)$. If μ_{AUP} were to take the value $\min\left(\mu_{RMP},\mu_{VMR}\right)$ then a small value of μ_{RMP} such that $\mu_{RMP}<\mu_{VMR}$ would cause μ_{AUP} to take the value of μ_{RMP} independent of the value of μ_{VMR}. The use of AND_2 instead of $\min$ allows finer distinctions to be made. The second degree dependence of μ_{RMP} in Eq. (22) results in a small value of μ_{AUP} if μ_{RMP} is small, but μ_{AUP} is still dependent on μ_{VMR}. This is consistent with expertise. If the sensor or non-sensor reliabilities or mission priority are small, μ_{AUP} should be small. Low reliability or priority results in a faster decline in μ_{AUP} than high mission risk, high UAV value, low UAV risk tolerance or the fact that a reliable and risk-tolerant UAV is slow.

It should be observed that the decisions made by AUP are taken over a pool of candidates. So it is reasonable

Table 1
Three UAV example for the composite concept RMP

Three UAV Numerical Example for Use of RMP			
Fuzzy Membership Function	UAV 1	UAV 2	UAV 3
Sensor reliability, μ_{sr}	**0.8**	0.7	**0.8**
Non-sensor reliability, μ_{nsr}	**0.9**	0.4	0.8
Risk Tolerance, $\mu_{\mathrm{risk-tol}}$	0.3	0.4	**0.5**
Mission Priority, μ_{MP}	0.6	**1**	0.4
$\mu_{RMP} \equiv \min(\mu_{sr}, \mu_{nsr}, \mu_{MP})$	0.6	0.4	0.4

those UAVs in the pool that have greater sensor and non-sensor system reliabilities and greater relative mission priority, should have a higher fuzzy grade of membership for assignment to the path under consideration. AUP's second degree dependence on RMP may allow RMP to be too dominant in the calculations. In the genetic program based approach described below versions of RMP were evolved where the power of RMP was between 1.5 and 1.7. These different versions of RMP are still under study.

The fuzzy concept AUP is depicted as a tree in Fig. 1. Leaves of the tree, i.e., those vertices of degree one are labeled by the names of the fuzzy concepts described above. Vertices are labeled by the specific logical connective used, i.e., $\min$ or AND_2. A circle on an edge indicates the fuzzy logic modifier *not*. The fuzzy modifier *not* is defined as the complement of the fuzzy set, i.e., let μ_A be the fuzzy membership function for the fuzzy concept A then membership function for *not A* is given by $1 - \mu_A$. A three UAV example is provided in Table 1 to illustrate the use of the RMP subtree of Fig. 1.

Given the fuzzy grade of membership it is necessary to defuzzify, i.e., make definite UAV-path assignments. Simply assigning the UAV with the highest fuzzy grade of membership for a particular path to that path can give less than desirable results. The approach to defuzzification taken is as follows: if the number of UAVs is denoted as n_{UAV} and likewise, the number of paths is denoted by n_{path}, where $n_{UAV} \geqslant n_{\mathrm{path}}$ then consider the set of all possible permutations of the n_{path} UAVs selected from a total of n_{UAV} UAVs. For each assignment of n_{path} UAVs to the paths, add up the values of μ_{AUP} for that assignment over the paths. This sum is referred to as the assignment benefit (AB). The assignment with the highest AB is the one selected. Finally, a similar procedure is followed if $n_{UAV} < n_{\mathrm{path}}$.

The decision tree for AUP given in Eq. (22) was constructed using expertise provided by human experts. It is a significant improvement over a previously developed fuzzy decision rule for path assignment also constructed from expertise [24]. An alternate method of obtaining Eq. (22) is to evolve it using a genetic program (GP). A GP is a computer program based on the theory of evolution that evolves mathematical expressions or computer programs that can be considered optimal in a sense. The GP has been used as a data mining function to create the decision tree in Eq. (22). The GP data mined a scenario database where each scenario had been labeled by an expert. Expert rules were also incorporated to guide the evolutionary process and improve convergence time. The decision tree in Eq. (22) has been evolved many times. The GP finds the same AUP decision tree, over and over again independent of the seed of the random number generator used to simulate a random evolutionary process. The GP based procedure is described in greater detail in the next subsection and an earlier version in [19].

2.3. GP creation of the AUP tree

The AUP tree given in the previous subsection arose from rules provides by human experts. In this subsection a method of evolving the AUP tree using data mining is discussed. This procedure has been successful in evolving the exactly same tree found in Fig. 1. The fact that two significantly different methods of obtaining the AUP algorithm give the same results provides a significant level of confidence in the AUP tree. Finally, the fact that the data mining approach is an optimization procedure, i.e., it gives results that are optimal in a sense should further strengthen confidence in the AUP tree.

Data mining is the efficient extraction of valuable non-obvious information embedded in a large quantity of data [2]. Data mining consists of three steps: the construction of a database that represents truth; the calling of the data mining function to extract the valuable information, e.g., a clustering algorithm, neural net, genetic algorithm, genetic program, etc; and finally determining the value of the information extracted in the second step, this generally involves visualization.

In a previous paper, a genetic algorithm (GA) was used as a data mining function to determine parameters for fuzzy membership functions [14]. Here, a different data mining function, a genetic program [7] is used. A genetic program is a problem independent method for automatically evolving computer programs or mathematical expressions.

The GP data mines fuzzy decision tree structure, i.e., how vertices and edges are connected and labeled in a fuzzy decision tree. The GP mines the information from a database consisting of scenarios.

The GP evolves a population of decision trees. Each tree is a candidate solution for the AUP tree. To create these candidate solutions the GP requires two input sets. These are the terminal set and function set.

At the end of each generation the GP ultimately rates or determines the fitness of each candidate solution produced during that generation. Its method of rating the candidate solutions is to calculate their fitness using a fitness function. When using a GP as a data mining function the GP requires a third type of input. This third input set is a scenario database where each scenario has been labeled by an expert with a real number between zero and one. This label corresponds to the decision that the optimal decision tree should make. So it is natural when calculating the fitness for a given candidate solution for the AUP tree to compare its final composite concept fuzzy membership value to the value labeling that scenario. So the scenario database is used to construct the final fitness function for ultimately determining the optimal AUP decision tree as determined by the GP.

The terminal set, function set, and fitness functions necessary for the GP to be used as a data mining function to automatically create the AUP tree are described below. The terminal set used to evolve the AUP tree consisted of the root concepts from the AUP tree and their complements. The terminal set, T, is given by

$$\begin{aligned} \mathrm{T} = \{&\text{risk-tol, value, fast, low-risk, sr, nsr,} \\ &\text{MP, not-risk-tol, not-valuable, not-fast,} \\ &\text{not-low-risk, not-sr, not-nsr, not-MP}\}. \end{aligned} \tag{23}$$

Let the corresponding fuzzy membership functions be denoted as

$$\begin{aligned} &\{\mu_{\text{risk}-\text{tol}}, \mu_{\text{value}}, \mu_{\text{fast}}, \mu_{\text{low}-\text{risk}}, \mu_{sr}, \\ &\mu_{nsr}, \mu_{MP}, \mu_{\text{not}-\text{risk}-\text{tol}}, \\ &\mu_{\text{not}-\text{valuable}}, \mu_{\text{not}-\text{fast}}, \\ &\mu_{\text{not}-\text{low}-\text{risk}}, \mu_{not-sr}, \mu_{not-nsr}, \mu_{not-MP}\}. \end{aligned} \tag{24}$$

When mathematical expressions are constructed by a GP that reproduce the entries in a database within some tolerance, the process is referred to as symbolic regression [10]. It is found in symbolic regression that candidate solutions are frequently not in algebraic simplest form and this is the major source of their excess length. When candidate solutions are too long this is referred to as bloat [10].

By including in the terminal set a terminal and its complement, e.g., "risk-tol," and "not-risk-tol"; "value" and "not-valuable"; etc., it is found that bloat is less and convergence of the GP is accelerated. This is a recent innovation which was not used when another resource manager, i.e., the electronic attack resource manager (EARM) was evolved using GP based data mining (DM) [17]. Additional bloat control procedures are described below and in [23] which provides much less detail than found here.

The mathematical form of the complement whether it appears in the terminal set or is prefixed with a "NOT" logical modifier from the function set is one minus the membership function. To make this more explicit

$$\mu_{NOT(A)} = \mu_{not-A} = 1 - \mu_A, \tag{25}$$

where $NOT(A)$ refers to the application of the logical modifier NOT from the function set to the fuzzy concept A from the terminal set. The notation, *not-A* refers to the terminal which is the complement of the terminal A.

The function set, denoted as F, consists of

$$F = \{AND1, OR1, AND2, OR2, NOT\} \tag{26}$$

where the elements of Eq. (26) are defined in Eqs (20), (27)–(30). Let A and B represent fuzzy membership functions then elements of the function set are defined as

$$AND1\,(A, B) = \min\,(A, B)\,; \tag{27}$$

$$OR1\,(A, B) = \max\,(A, B)\,; \tag{28}$$

$$OR2\,(A, B) = A + B - A \cdot B; \tag{29}$$

and

$$NOT\,(A) = 1 - A. \tag{30}$$

The database to be data mined is a scenario database kindred to the scenario database used for evolving the EARM [17]. In this instance scenarios are characterized by values of the fuzzy membership functions for the elements of the terminal set plus a number from zero to one indicating the experts' opinion about the value of the fuzzy membership function for AUP for that scenario.

GPs require a fitness function [7]. As its name implies the fitness function measures the merit or fitness of each candidate solution represented as a chromosome. The fitness used for data mining is referred to as the input-output fitness.

The input-output fitness for mining the scenario database takes the form

$$f_{IO}(i, n_{db}) \equiv \frac{1}{1 + 2 \cdot \sum_{j=1}^{n_{db}} \left| \mu_{gp}\left(i, e_j\right) - \mu_{\text{expert}}\left(e_j\right) \right|}. \tag{31}$$

where e_j is the j^{th} element of the database; n_{db} is the number of elements in the database; $\mu_{gp}(e_j)$ is the output of the fuzzy decision tree created by the GP for the i^{th} element of the population for database element e_j; and $\mu_{\text{expert}}(e_j)$ is an expert's estimate as to what the fuzzy decision tree should yield as output for database element e_j.

The AUP tree is evolved in three steps. The first step involves evolving the VMR subtree; the second step, the RMP subtree and the final step, the full AUP tree. In the second and third steps, i.e., evolving the RMP subtree and full AUP tree from the RMP and VMR subtrees, only the input-output (IO) fitness in Eq. (31) is calculated, i.e., the rule-fitness described below is not used.

When evolving the VMR subtree a rule-fitness is calculated for each candidate solution. Only when the candidate's rule fitness is sufficiently high is its input-output fitness calculated. The use of the rule-fitness helps guide the GP toward a solution that will be consistent with expert rules. Also the use of the rule fitness reduces the number of times the IO fitness is calculated reducing the run time of the GP. After some preliminary definitions of crisp and fuzzy relations, a set of crisp and fuzzy rules that were used to help accelerate the GP's creation of the VMR subtree are given. The rules are combined to formulate the rule fitness.

Let T be a fuzzy decision tree that represents a version of the VMR subtree, that is to be evolved by a genetic program. Let A and B be fuzzy concepts. Then let $\gamma_{share}(T, A, B) = 1$ if A and B share a logical connective denoted as C and $\gamma_{share}(T, A, B) = 0$, otherwise.

Furthermore, define the fuzzy relation

$$\mu_{com}(T, A, B, C) = \begin{cases} 0.4 \text{ if } C = AND1 \text{ or } AND2 \\ 0.1 \text{ if } C = OR1 \text{ or } OR2 \\ 0, \text{ otherwise} \end{cases}. \tag{32}$$

The following is a subset of the rules used to accelerate the GP's convergence and to help produce a result consistent with human expertise.

R1. "not-valuable" and "risk-tol" must share a logical connective, denoted as C_1, i.e., it is desired that γ_{share} (T,not-valuable,risk-tol) $= 1$.

R2. "not-valuable" and "risk-tol" strongly influence each other, so they should be connected by AND1 or AND2. So it is desired that

$$\mu_{com}(\text{T,not-valuable,risk-tol},C_1) = 0.4.$$

R3. "fast" and "low-risk" have an affinity for each other. They should share a logical connective, denoted asC_2, i.e., it is desired that γ_{share} (T,fast,low-risk) $= 1$.

R4. The fuzzy root concepts "fast" and "low-risk" strongly influence each other, so they should be connected by AND1 or AND2. So it is desired that μ_{com} (T,fast,low-risk,C_2) $= 0.4$.

R5. There is an affinity between the fuzzy root concepts C_1 (not-valuable,risk-tol) and C_2 (fast,low-risk), they are connected by a logical connective denoted as C_3, i.e., it is desired that,

$$\begin{array}{l}\gamma_{\text{share}}(T, C_1\,(\text{not-valuable,risk-tol}), \\ C_2\,(\text{fast,low-risk})) = 1\end{array}. \tag{33}$$

When the EARM was evolved by GP based data mining [17] bloat was controlled using adhoc procedures based on tree depth and parsimony pressure. Most of the bloat in evolving mathematical expressions with a GP arises from the expressions not being in algebraic simplest form [10]. With that observation in mind, computer algebra routines have been introduced that allow the GP to simplify expressions. The following is a partial list of algebraic simplification techniques used during the evolution of the EARM and the AUP tree. The simplification routines used when evolving AUP are more sophisticated than those applied to the creation of EARM [17].

One routine simplifies expressions of the form N*OT*(*NOT*(*A*)) = *A*. This can be more complicated than it initially appears, since the NOT logical modifiers can be separated on the fuzzy decision tree.

Another simplification procedure consists of eliminating redundant terminals connected by an AND1 logical connective. An example of this is *AND1*(*A*,*A*) = *A*. Like the case with the logical modifier NOT there can be a separation between the AND1s and the terminals that add complexity to the simplification operation.

The third algebraic simplification example is like the second. It involves simplifying terminals connected by OR1s. Like AND1, separation between terminals and OR1 can increase the complexity of the operation.

Other types of algebraic simplification use DeMorgan's theorems in combination with the above procedures. This can significantly reduce the length of an expression.

Another algebraic procedure that reduces the length of expressions includes replacement of forms like AND2(A,A) by the square of "A," i.e., A^2. Still another length reducing simplification includes replacing NOT(A) with not-A, its complement from the terminal set listed in Eq. (23).

There is always a question of how much algebraic simplification should be conducted from generation to generation as such the simplification algorithm allows levels of simplification. If a low level of simplification is selected then some parts of an expression remain that might be eliminated during full simplification. This has two advantages: it leaves chromosome subcomponents that may prove useful during mutation or crossover and it takes less CPU time.

Algebraic simplification produces candidate solutions in simpler form making it easier for human observers to understand what is being evolved. Having candidate solutions that are easier to understand can be an important feature for improving the evolution of GPs.

3. Control algorithm

Each UAV has a real-time algorithm onboard it that allows recalculation of paths during flight due to changes in environmental conditions or mission priorities. These changes typically become apparent after the planning algorithm has run during the pre-flight stage. As in the case of the planning algorithm the control algorithm uses an A-star algorithm [13] to do the best path calculation, employs fuzzy logic and solves a constrained optimization problem. A-star's and fuzzy logic's CPU time requirements have been quite satisfactory for this application.

The control algorithms' recalculation of flight paths can be triggered by a number of events such as weather broadcasts that indicate new taboo regions or changes of priority of sample points. For those changes that do not require UAVs supporting each other, the control algorithm does not differ from the planning algorithm. The control algorithm is faster by virtue that it only need process those parts of the measurement space where there have been changes relative to sample or taboo points. The A-star algorithm is particularly effective in an environment where change is confined to small regions.

A UAV may requests help if it discovers a potential elevated system like a radio hole, malfunctions or suspected malfunctions. All of these conditions can result in help messages being transmitted between the UAVs. These help messages can result in interactions between the UAVs based on transmission of the results of priority calculations for rendering support to the requesting UAVs.

Currently in the control stage, when a UAV discovers an interesting physical phenomenon, is malfunctioning, or suspects due to internal readings that it is malfunctioning, it sends out a request for help. Each UAV receiving this message calculates its priorities for providing assistance to the UAV in need. This priority calculation gives rise to a number between zero and one, inclusive, which is subsequently transmitted to the original UAV desiring support. The requesting UAV sends out an omni-directional message with the ID of the UAV with highest priority for contributing support. The high priority UAV then flies into the necessary neighborhood of the requesting UAV to provide help.

There are three classes of help request. The first occurs when a UAV, the requester, determines it may have discovered an interesting physical phenomenon. This phenomenon may be an elevated duct, radio hole, rain system or some other type of system with physical extent. The requester desires to determine if the phenomenon has significant extent. It will request that a helping UAV or UAVs sample likely distant points within this phenomenon.

The second class of help request relates to a UAV that according to internal diagnostics may be experiencing a sensor malfunction. This UAV will requests that another UAV or UAVs measure some of the points that the requesting UAV measured. This will help determine if the UAV is actually malfunctioning. If the requesting UAV is determined to be malfunctioning, then it will fly back to base, if it is capable. The determination of whether it is actually malfunctioning requires some consideration. Since the second UAV will probably be measuring a distant point at a time different than the original requesting UAV made its measurements, potential variation in the index of refraction over time must be taken into account.

When a UAV sends out an omni-directional request for help, those UAVs receiving the message will calculate their fuzzy priority for helping, denoted as "PH." The UAV that will ultimately help the requester is the one with the highest fuzzy priority for helping. The fuzzy priority for helping takes into account a variety of properties of the potential helper. The set of UAVs that receive the request for help from UAV(i) at time t is denoted as help(i,t). If UAV(i) request help at time t and UAV(j) receives the message then UAV(j) will take into account the amount of time, denoted, help_time$(UAV(j))$, it will take it to fly from the point where it received the request to the point where it would provide support. It also takes into account the amount of fuel UAV(j) has left at the time of the request, denoted fuel$(UAV(j),t)$; UAV(j)'s fuzzy concept of price denoted as *"price"*, and UAV(j)'s fuzzy concept of "*mission priority*" at time, t. Let the set of relevant UAV properties be denoted as UAV_prop and be defined as

$$UAV\_prop = \{\text{help\_time},\text{fuel},\text{mission\_prio},\text{price}\} \tag{34}$$

The fuzzy priority for helping denoted as μ_{PH} takes the form

$$\mu_{PH}(UAV(i),UAV(j)) = \sum_{\delta \in UAV\_prop} w_\delta \cdot \mu_\delta(UAV(j)) \tag{35}$$

The quantities w_δ and μ_δ for $\delta \in UAV\_prop$ are expert defined weights and fuzzy membership functions, respectively. The fuzzy membership functions are defined in Eqs (36)–(39) and given below,

$$\mu_{\text{help\_time}}(UAV(i),UAV(j)) = \left[\frac{\text{help\_time}(UAV(j))}{\max\limits_{k\in \text{help}(i,t)}\{\text{help\_time}(UAV(k))\}} + 1\right]^{-1} \tag{36}$$

$$\mu_{\text{fuel}}(UAV(i),UAV(j)) = \frac{\text{fuel}(UAV(j))}{\max\limits_{k\in \text{help}(i,t)}\{\text{fuel}(UAV(k))\}} \tag{37}$$

$$\mu_{\text{mission\_prio}}(UAV(i),UAV(j)) = \left[\frac{\text{mission\_prio}(UAV(j))}{\max\limits_{k\in \text{help}(i,t)}\{\text{mission\_prio}(UAV(k))\}} + 1\right]^{-1} \tag{38}$$

$$\mu_{\text{price}}(UAV(i),UAV(j)) = \left[\frac{\text{Value}(UAV(j))}{\max\limits_{k\in \text{help}(i,t)}\{\text{Value}(UAV(k))\}} + 1\right]^{-1} \tag{39}$$

It is assumed that all evaluations are processed at time, t, so time dependence is suppressed in Eqs (35)–(39) for notational convenience. A more sophisticated version of the control logic that takes path risk, changes in risk, UAV reliability, UAV risk-tolerance and missed sample points into account will be the subject of a future publication.

4. Computational experiments

The planning and control algorithms described in the previous sections have been the subject of a large number of experiments. This section provides a description of a small subset of these experiments. They serve to illustrate how the algorithms were tested.

UAV experiments using only one UAV demonstrate how the planning and control algorithm will determine the route the UAV flies so that it is successful in making measurements at sample points in space, while the UAV avoids taboo points, that is points in space that could damage or destroy the UAV. Experiments using two UAVs illustrate how the control algorithm allows the UAVs to automatically support each other to increase the probability their joint mission is successful.

Figures 2–6 use the same labeling conventions. Sample points are labeled by concentric circular regions colored in different shades of gray. The lighter the shade of gray used to color a point, the lower the point's grade of membership in the fuzzy concept "desirable neighborhood." The legend provides numerical values for the fuzzy grade of membership in the fuzzy concept "desirable neighborhoods". If the fuzzy degree of desirability is high then the index of refraction is considered to be close to the index of refraction of the sample point at the center of the desirable neighborhood. This allows the UAV to make significant measurements while avoiding undesirable neighborhoods.

Each sample point is labeled with an ordered pair. The first member of the ordered pair provides the index of the sample point. The second member of the ordered pair provides the point's priority. For example, if there are n_{sp} sample points and the q^{th} sample point is of priority p, then that point will be labeled with the ordered pair (q,p).

Points surrounded by star-shaped neighborhoods varying from dark grey to white in color are taboo points. As with the sample points, neighborhoods with darker shades of gray have a higher grade of membership in the fuzzy concept "undesirable neighborhood." The legend provides numerical values for the fuzzy

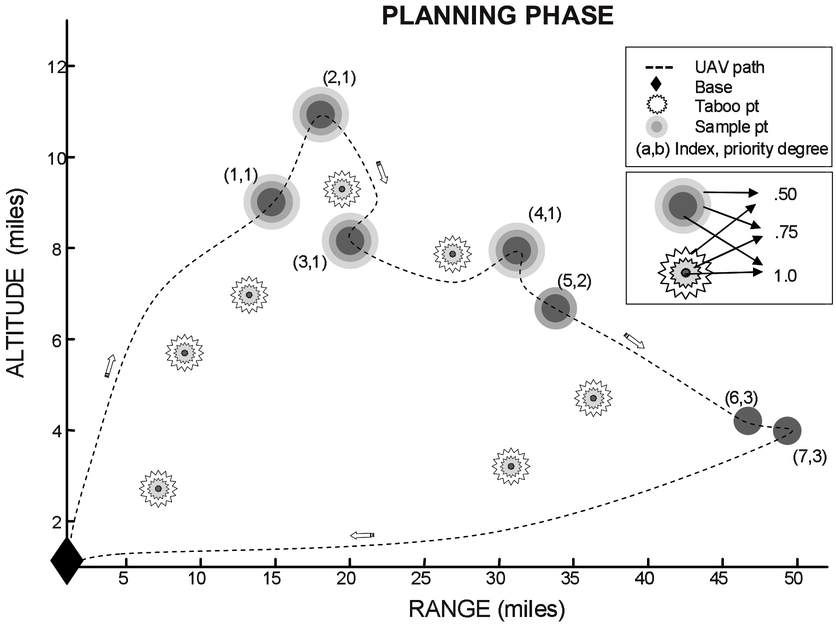

Fig. 3. One UAV trajectory as determined by planning algorithm.

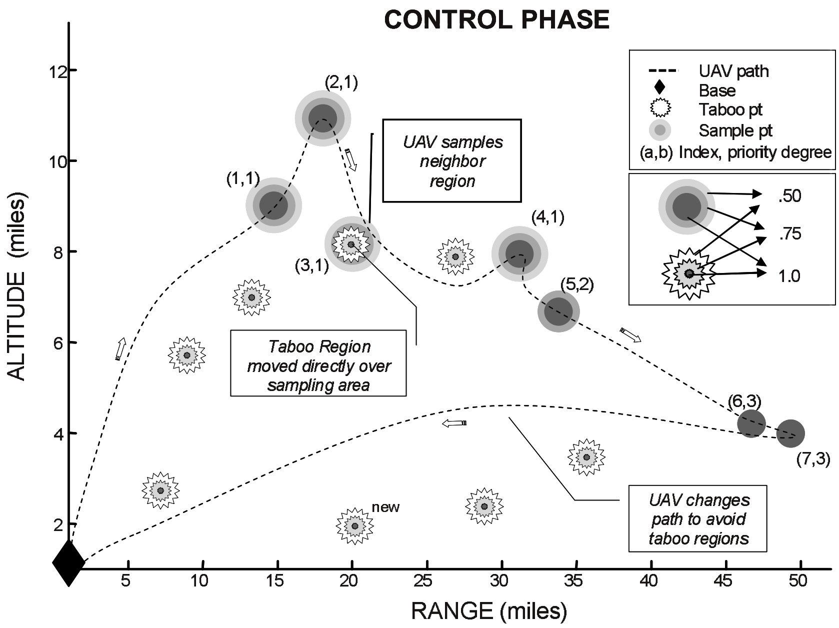

Fig. 4. One UAV trajectory as determined by real-time control algorithm.

grade of membership in the fuzzy concept "undesirable neighborhood." UAVs with high risk tolerance may fly through darker grey regions than those with low risk tolerance. When comparing planning and associated control pictures, if a point ceases to be taboo, the neighborhood where it resides is marked by a very dim gray star as well as being labeled by a dialog box as being an "old taboo point." New taboo points and their associated undesirable neighborhoods are labeled with dialog boxes indicating that they are "new."

Each UAV has three states of motion, it can be at rest, traveling at sampling speed or non-sampling speed.

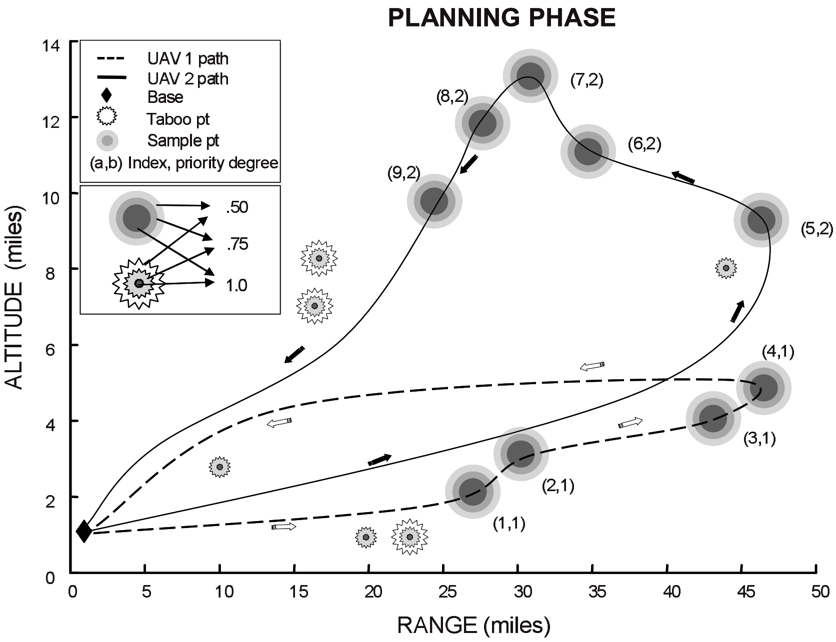

Fig. 5. Trajectory of two UAVs as determined by planning algorithm.

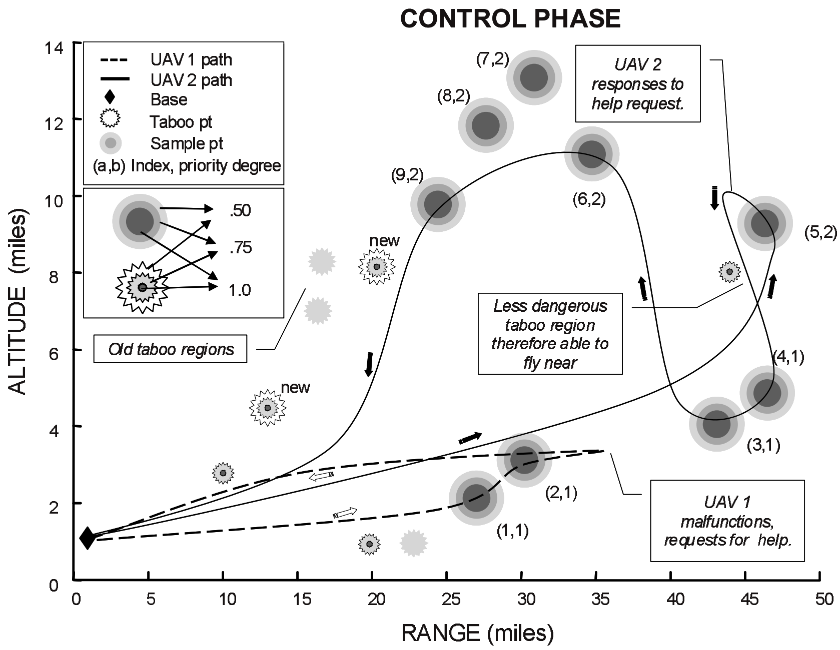

Fig. 6. During flight, updates about changes cause the real-time control algorithms on the two UAVs to change their trajectories.

Within the simulation the UAV properties other than speed are its risk-tolerance, cost, fuel, sensor reliability and non-sensor reliability. The values associated with each property can vary over the pool of UAVs in use.

UAVs start their mission at the UAV base which is labeled with a diamond-shaped marker. They fly in the direction of the arrows labeling the various curves in Figs 2–5.

In Figs 2–6, flight paths that can be represented in two dimensions have been selected for easy comprehension.

In the general case the flight paths are three dimensional spatial curves, which can be difficult to visualize.

Figure 3 provides the sample points, taboo points and sample path for one UAV as determined by the planning algorithm. It is important to notice that the UAV's path passes directly through each sample point, i.e., through the center of the concentric circular regions representing the fuzzy degree of desirability of neighborhoods. Fortuitously, the taboo points and their neighborhoods are so positioned that they do not interfere with the UAV's measurement process or its return to base.

Figure 4 depicts the actual path the UAV flies as determined by the UAV's real-time control algorithm. The path determined by the control algorithm differs from the one created by the planning algorithm due to real-time changes in taboo points. After leaving the UAV base new weather data was acquired informing the UAVs that the exact position of the third sample point, i.e., the one labeled (3,1) actually resides within an undesirable neighborhood. Due to the high priority of the sample point and the UAV's risk-tolerance, the UAV flies into the taboo points' undesirable neighborhood as indicated in Fig. 4.

In both the planning and control algorithms the UAV measures sample points of two different priorities, with the direction of the flight path selected so that the higher priority points are measured first. By measuring high priority points first, the likelihood of an important measurement not being made is diminished, if the UAV can not complete its mission due to a malfunction, change in weather, etc.

Also, due to movement of old taboo points or the emergence of new taboo points which are marked "New," the path determined for the UAV using the control algorithm is significantly different than the one created by the planning algorithm. The path change represents the control algorithm's ability to reduce UAV risk.

Figure 5 depicts the sampling path determined by the planning algorithm for an experiment involving two UAVs. The first, UAV(1) follows the dashed curve; the second, UAV(2), the solid curve. The UAVs were assigned to the different paths by the fuzzy path assignment decision tree described in Section 2. UAV(1) is assigned to sample all the highest priority points, i.e., the priority one points. UAV(2) samples the lower priority points, i.e.; those with priority two. Due to the greedy nature of the point-path assignment algorithm, the highest priority points are assigned for sampling first.

Figure 6 depicts the actual flight path the UAVs take during real-time. Initially, UAV(1) is successful in measuring sample points one and two as assigned it by the planning algorithm. Just beyond sample point two, UAV(1) experiences a malfunction. UAV(1)'s real-time control algorithm subsequently sends out a help request informing the only other UAV in the field, UAV(2) of the malfunction. UAV(2)'s control algorithm determines a new path for UAV(2) to fly so that the priority one points, labeled (3,1) and (4,1), that UAV(1) was not able to sample are subsequently measured. After UAV(2) measures sample point five, its new flight path allows it to measure sample points three and four. UAV(2)'s control algorithm determined it was very important that these priority one points be measured. Unfortunately, due to the extra fuel expended in reassigning sample points three and four to UAV(2), UAV(2) did not have enough fuel to measure sample points seven and eight which were of priority two. UAV(2)'s real-time control algorithm determined the best possible solution in the face of changing circumstances and limited resources.

It is important to note that the control algorithms running on UAV(1) and UAV(2) direct both UAVs to alter their return paths to the base due to the emergence of new taboo points making the planning algorithm determined flight paths too dangerous. The control algorithm uses each UAV's fuzzy risk-tolerance to determine how near each UAV may approach a taboo point.

Figure 7 provides an example of the AUP decision tree's assignment of three UAVs to three paths. The highest priority locations are assigned to UAV(1) as it has the greatest fuel capacity, i.e., 90 minutes. UAV(1) however does not have enough fuel to handle the high priority points located at positions six and seven and therefore UAV(2) is assigned these points along with the second degree high priority locations.

Table 2 provides numerical details of the tasks depicted in Fig. 7. The column labels have the following interpretation: "Location," the UAV coordinates on the map; "Fly mode," whether the UAV sampled from its previous location to its current position. If the UAV sampled then a "S" was entered. "NS" was entered if sampling did not occur. "Fuel Time" refers to how much fuel remained by the time the UAV reached the associated location.

5. Conclusions and summary

Fuzzy logic algorithms for planning and control of a coordinated team of robots making atmospheric mea-

Table 2
Details of three UAV mission depicted in Fig. 7

Three UAV Mission								
UAV 1 MISSION			UAV 2 MISSION			UAV 3 MISSION		
Locations	Fly Mode	Fuel Time Remain (minutes)	Locations	Fly Mode	Fuel Time Remain (minutes)	Locations	Fly Mode	Fuel Time Remain (minutes)
Base		90.0	Base		85.0	Base		85.0
(1,1)	NS	76.5088	(6,1)	NS	67.9691	(11,3)	NS	64.2839
(2,1)	S	61.5088	(7,2)	S	55.2412	(12,3)	S	51.0412
(3,1)	S	54.2662	(8,2)	S	47.9986	(13,3)	S	39.5559
(4,1)	S	42.7809	(9,2)	S	39.5133	(14,3)	S	31.0706
(5,1)	S	28.2956	(10,2)	S	22.028	Base	NS	6.2574
Base	NS	6.7113	Base	NS	11.7854			

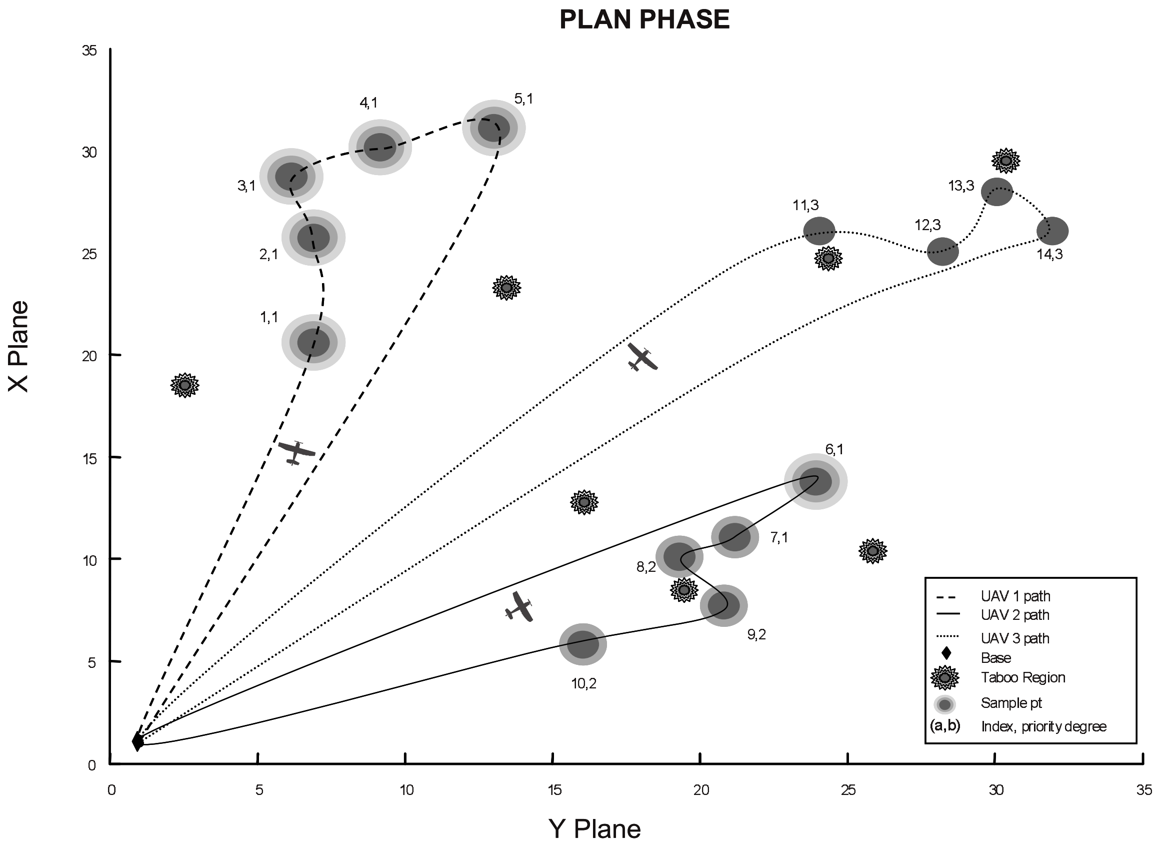

Fig. 7. Three UAV mission described in Table 2, an example of the AUP decision tree's assignment.

surement have been developed. The algorithms exhibited excellent performance under extensive testing in digital simulation.

The fuzzy logic algorithms provide explicit, concise, elegant mathematical relationships between input root concepts like risk-tolerance, sensor reliability, non-sensor system reliability, mission-risk, mission-priority and output composite fuzzy concepts like "assign UAV to path." Such mathematical relationships are valuable for subsequent innovation. Only having an implicit knowledge of the relationship between concepts as frequently occurs if only an optimization approach is used is less desirable. Such implicit relationships can make innovation considerably more difficult.

The fuzzy logic based algorithms require little CPU time and can function in real-time even on many slow legacy processors. They also require very little memory storage.

The fuzzy logic algorithms allow a group of flying robots to collaborate without a central or fixed commander. Decisions are made based on communication. The team of robots automatically self-organize. This is valuable since the loss of anyone robot will not destabilize the team. Late arriving robots may join the group and contribute to the team's well-being without difficulty.

Self-organization arises by the UAV's transmitting only small amounts of information. The UAVs only transmit processed information, fuzzy grades of membership, resulting in very low bandwidth requirements. Since the bandwidth requirements are low, no expensive data compression is required. This means physical power and time are saved. This is a valuable feature for future inexpensive, disposable, low powered systems.

A method of creating fuzzy decision logic using an algorithm related to the theory of evolution, a genetic program (GP) is discussed. This algorithm's evolution is guided by expert provided scenarios in a data base and expert rules in the form of fuzzy logic. Ultimately, the GP evolves a fuzzy decision tree that is optimal, with respect to the expertise provided for the desired task. The GP has been successful in reproducing known results as well as creating new distinct algorithms.

The GP's output, the fuzzy decision logic is concise, elegant and understandable by humans. This is largely due to the use of partial expertise embedded in the GP in the form of fuzzy rules and innovative bloat control mechanisms like computer algebra. All the bloat control mechanisms contribute to shorter solutions, but the use of fuzzy rules within the GP to guide evolution produces results closer to human intuition.

In the case where the decision logic is the same as that found by interviewing experts, it is easy to argue the GP's results are understandable by humans, after all, the GP has reproduced results handed down by human experts. In the case where the results are different from those obtained from expertise, they can be related to the expert results and the differences understood.

6. Future directions

The various techniques for controlling GP based bloat, i.e., generating more concise results must be examined to determine their assets and liabilities. Also, various techniques for accelerating the convergence of the GP will be explored.

Additional fuzzy logic algorithms will be added to give the UAVs greater flexibility. This will require consulting with experts to obtain rules. In the absence of a good set of rules it should be possible to construct a scenario data base for mining by the GP. A partial set of rules about the anticipated shape of fuzzy decision trees can be used to help accelerate the GP's convergence and produce simpler, more concise results understandable by human beings as in the AUP decision tree case.

From the practical perspective of modeling physical systems, the fidelity of the underlying UAV models, the electromagnetic propagation model, and environmental models will be increased.

Increases in fidelity of the UAV model will include specifics of the UAV engine model, e.g., how does its efficiency change with altitude. The ray-theoretic electromagnetic propagation model will be partially replaced with a ray-wave-theoretic hybrid [9]. This will allow greater fidelity in modeling rough surface scattering from terrain and also various types of ducting phenomena. By using a hybrid model increases in CPU time and memory requirements can be kept in check as the fidelity of the modeling of underlying physics increases.

Changes in modeling fidelity will also result in alterations of the cost function related to flight path determination. It will probably also prove useful, eventually to explore more sophisticated algorithms for path determination [5,11,12,28] that will replace A-star. An algorithm of greater sophistication may be useful for rapidly changing environments or when there are more obstructions or threats within the environment.

Acknowledgements

This work was sponsored by the Office of Naval Research. The authors would also like to acknowledge Mr. Alan Schultz, Dr. Lawrence Schuette, Dr. Jeffrey Heyer, Dr. Francis Klemm, and Dr. Gregory Cowart.

References

[1] M. Azizoglu, S. Subrarnaniam and A.K. Somani, Converter placement on wavelength-routed network paths, in: *Converter placement on wavelength-routed network paths III*, John Senior, ed., San Diego, SPIE Proceedings, October 1997, pp. 265–275.

[2] J.P. Bigus, *Data Mining with Neural Nets*, New York, McGraw-Hill, 1996.

[3] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Chapter 11, Boston, Artech House, 1999.

[4] L.V. Blake, *Radar Range-Performance Analysis*, Artech House, Boston, 1986.

[5] K.R. Buck, R. Gassner, A.B. Poore and X. Yan, Visibility, Constrained Routing of Unmanned Aerial Vehicles, in: *Signal Processing, Sensor Fusion, and Target Recognition VIII,* (Vol. 3720), I. Kadar, ed., SPIE Proceedings, Orlando, Florida, April 1999, pp. 256–266.

[6] C. Jeffries, Tracking, code recognition, and memory management with high order neural networks, in: *Applications and Science of Artificial Neural Networks*, Vol. 2492, S. Rogers, ed., SPIE Proceedings, Orlando, Florida, April 1995, pp. 364–373.

[7] J.R. Koza, F.H. Bennett III, D. Andre and M.A. Keane, *Genetic Programming III*: *Darwinian Invention and Problem Solving,* Morgan Kaufmann Publishers, San Francisco, 1999.

[8] J. Latourell, B. Wallet and B. Copeland, Genetic Algorithm to Solve Constrained Routing Problem with Applications for Cruise Missile Routing, in: *Applications and Science of Computational Intelligence*, Vol. 3390, S. Rogers, ed., SPIE Proceedings, Orlando, Florida, April 1998, pp. 490–500.

[9] M. Levy, *Parabolic equation methods for electromagnetic wave propagation*, Institution of Electrical Engineers, Herts, United Kingdom, 2000.

[10] S. Luke and L. Panait, Fighting Bloat with Nonparametric Parsimony Pressure, in: *Parallel Problem Solving from Nature – PPSN VII,* 7^{th} *International Conference Proceedings,* LNCS Vol. 2439, J.J.M. Guervos, ed., Berlin, Springer-Verlag, 2002, pp. 411–421.

[11] R. Mittu and J.K. Uhlmann, Routing and Advanced Display Technologies within STOMPM, in: *Robotic and Semi-Robotic Ground Vehicle Technology Vol. 3366*, G. Gerhart, ed., Orlando, Florida, SPIE Proceedings, April 1998, pp. 177–188.

[12] H.G. Nguyen, H.R. Everett, N. Manouk and A. Verma, Autonomous mobile communication relays, in: *Unmanned Ground Vehicle Technology IV Vol. 4715,* G.R. Gerhart, C.M. Shoemaker and D.W. Gage, eds, Orlando, Florida, SPIE Proceedings, April 2002, pp. 50–57.

[13] S.J. Russel and P. Norvig, *Artificial Intelligence: A Modern Approach*, (2nd ed.), Englewood Cliffs, Prentice-Hall, 2002.

[14] J.F. Smith, III and R. Rhyne, II, A Resource Manager for Distributed Resources: Fuzzy Decision Trees and Genetic Optimization, in: *Proceeding of the International Conference on Artificial Intelligence, IC-AI'99*, Vol. II, H. Arabnia, ed., Las Vegas CSREA Press, 1999, pp. 669–675.

[15] J.F. Smith, Co-evolutionary Data Mining to Discover Rules for Fuzzy Resource Management, in: *Proceedings of the International Conference for Intelligent Data Engineering and Automated Learning*, H. Yin, ed., Manchester, Springer-Verlag, August, 2002, pp. 19–24.

[16] J.F. Smith, Data Mining for Fuzzy Decision Tree Structure with a Genetic Program, in: *Proceedings of the International Conference for Intelligent Data Engineering and Automated Learning*, H. Yin, ed., Manchester, Springer-Verlag, August 2002, pp. 13–18.

[17] J.F. Smith, Fuzzy logic resource manager: decision tree topology, combined admissible regions and the self-morphing property, in: *Signal Processing, Sensor Fusion, and Target Recognition XII*, Vol. 5096, I. Kadar, ed., Orlando, SPIE Proceedings, April 2003, pp. 104–114.

[18] J.F. Smith, Fuzzy logic resource manager: real-time adaptation and self organization, in: *Signal Processing, Sensor Fusion, and Target Recognition XIII*, Vol. 5429, I. Kadar, ed., Orlando, SPIE Proceedings, April 2004, pp. 77–88.

[19] J.F. Smith, Genetic Program Based Data Mining for Fuzzy Decision Trees, in: *Proceedings of the International Conference for Intelligent Data Engineering and Automated Learning*, H. Yin, ed., Exeter, Springer-Verlag, August 2004, pp. 464–470.

[20] J.F. Smith and T.H. Nguyen, Distributed autonomous systems: resource management, planning, and control algorithms, in: *Signal Processing, Sensor Fusion, and Target Recognition XIV*, Vol. 5809, I. Kadar, ed., Orlando, SPIE Proceedings, April 2005, pp. 65–76.

[21] J.F. Smith, III and T.H. Nguyen, Data mining based automated reverse engineering and defect discovery, in: *Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2005*, B. Dasarathy, Vol. 5812, April 2005, Orlando, SPIE Proceedings, 2005, pp. 232–242.

[22] J.F. Smith and T.H. Nguyen, Fuzzy Logic Based UAV Allocation and Coordination, in: *Proceedings of the Third International Conference on Informatics in Control, Automation and Robotics: Intelligent Control Systems and Optimization,* J.A. Cetto, ed., Setubal, Portugal, ICINCO 2006 Proceedings, August 2006, pp. 9–18.

[23] J.F. Smith III and T.H. Nguyen, Guiding Genetic Program Based Data Mining Using Fuzzy Rules, in: *Proceedings of the International Conference for Intelligent Data Engineering and Automated Learning*, H. Yin, ed., Burgos, Spain, Springer-Verlag, September 20–23, 2006.

[24] J.F. Smith and T.H. Nguyen: Resource manager for an autonomous coordinated team of UAVs, in: *Signal Processing, Sensor Fusion, and Target Recognition XV: 62350C,* I. Kadar, ed., Orlando, SPIE Proceedings, April 2006, 104–114.

[25] J.F. Smith, III and T.H. Nguyen, Genetic program based data mining to reverse engineer digital logic, in: *Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2006*, Vol. 6241, B. Dasarathy, Orlando, SPIE Proceedings, April 2006.

[26] D. Spears and D. Zarzhitsky, Multi-Robot Chemical Plume Tracing, in: *Multi-Robot Systems: From Swarms to Intelligent Automata,* Vol. III, A. Schultz, ed., New York, Springer, May 2005, pp. 211–221.

[27] S.R. Thangiah and K.E. Nygard, School bus routing using genetic algorithms, in: *Applications of Artificial Intelligence X: Knowledge-Based Systems,* Vol. 1707, G. Biswas, Orlando, Florida, SPIE Proceedings, April 1992, pp. 387–398.

[28] K. Thyagarajan, R. Batta and M.H. Karwan, Planning Dissimilar Paths for Military Units, Progress Report Department of Industrial Engineering, University of Buffalo SUNY, Buffalo, NY 14260, USA, 2004.

[29] L.H. Tsoukalas and R.E. Uhrig, *Fuzzy and Neural Approaches in Engineering*, Chapter 5, New York, John Wiley and Sons, 1997.

[30] S. Verma and A.K. Saxena, Polynomial Time Algorithm for Hop Constrained QoS Routing, in: *Internet Quality of Service*, Vol. 5245, M. Atiquzzaman and M. Hassan, eds Orlando, Florida, SPIE Proceedings, April 2003, pp. 131–135.