11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference
6 - 8 September 2006, Portsmouth, Virginia

AIAA 2006-6995

THREE-DIMENSIONAL PATH PLANNING OF UNMANNED AERIAL VEHICLES USING PARTICLE SWARM OPTIMIZATION

Jung Leng Foo[1], Jared S. Knutzon[2], James H. Oliver[3], and Eliot H. Winer[4]
Virtual Reality Applications Center, Iowa State University, Ames, IA 50010, USA

Military operations are turning to more complex and advanced automation technology for minimum risk and maximum efficiency. A critical piece to this strategy is unmanned aerial vehicles (UAVs). UAVs require the intelligence to safely maneuver along a path to an intended target, avoiding obstacles such as other aircrafts or enemy threats. Often automated path planning algorithms are employed to specify targets for a UAV to fly to. To date, path-planning algorithms have been limited to two-dimensional problem formulations. This paper presents a unique three-dimensional path planning problem formulation and solution approach using Particle Swarm Optimization (PSO). The problem formulation was designed to minimize risk due to enemy threats while simultaneously minimizing fuel consumption. The initial design point is a straight path between the current position and the desired target. Using PSO, an optimized path is generated through B-spline curves. The resulting paths can be optimized with a preference towards maximum safety, minimum fuel consumption or a combination of the two. The problem formulation and solution implementation is described along with the results from several simulated scenarios.

I. Introduction

Military combat of the future will become highly dependent on the use of unmanned aerial vehicles (UAVs). In recent years, there has been rapid development in UAV technology such as swarm communication, command and control, and developing usable interfaces [1]. The complexity in UAV technology is rapidly growing, and according to the Department of Defense (DOD) Roadmap [2], by the year 2012 it is estimated that F-16 size UAVs will be able to perform a complete range of combat and combat support missions. Thus, the ground control station – the human operator's portal to the UAV – must evolve as UAVs grow in autonomy. The ground control station must facilitate the transformation of the human from pilot, to operator, to supervisor, as the level of interaction with UAVs moves to ever-higher levels. As humans interface with UAVs at more abstract levels, a UAV will be trusted to do more [3]. To develop and maintain that trust, a human must be able to understand a UAV's situation quickly. Future ground control stations will need to provide an operator with situational awareness and quality information at a glance.

To address the many research issues involved in the command and control that the DOD roadmap requires, a "Virtual Battlespace" at Iowa State University was created. In this paper, research into the issue of three-dimensional (3D) path planning for UAVs as part of the Virtual Battlespace project is presented. The method described allows a human operator to focus on selecting an appropriate path from a set of alternate paths produced by the path planner, easing the decision making process. Using a Particle Swarm Optimization (PSO) algorithm, the task of generating alternate paths is formulated into an optimization problem consisting of two main components: 1)

[1] Research Assistant, Department of Mechanical Engineering & Human Computer Interaction, Virtual Reality Applications Center, 2274 Howe Hall, Iowa State University, Ames, IA 50010, USA, Student Member.
[2] Research Assistant, Department of Human Computer Interaction, Virtual Reality Applications Center, 2274 Howe Hall, Iowa State University, Ames, IA 50010, USA.
[3] Professor, Department of Mechanical Engineering & Human Computer Interaction, Virtual Reality Applications Center, 2274 Howe Hall, Iowa State University, Ames, IA 50010, USA.
[4] Assistant Professor, Department of Mechanical Engineering & Human Computer Interaction, Virtual Reality Applications Center, 2274 Howe Hall, Iowa State University, Ames, IA 50010, USA, Member.

for UAVs to avoid obstacles such as threats (e.g., surface to air missile sites, tanks, and aircraft) and 2) maintaining a fuel-efficient path to maximize mission range.

In the following sections of this paper, the background of the Virtual Battlespace project is presented. This is followed by a detailed description of the development and implementation of the 3D path planner using a Particle Swarm Optimization algorithm. Paths generated from the planner from multiple simulated scenarios are then presented and discussed.

II. Background

A. Virtual Battlespace

Development of the Virtual Battlespace originated in 2000 when a research team at Iowa State University's Virtual Reality Applications Center (VRAC) began work with the Air Force Research Lab's Human Effectiveness Directorate and the Iowa National Guard's 133rd Air Control Squadron. The goal of this preliminary version of the Virtual Battlespace was to develop an immersive VR system for distributed mission training. Virtual Battlespace integrates information about tracks, targets, sensors and threats into an interactive virtual reality environment that fuses the available information about the battlespace into a coherent picture that can be viewed from multiple perspectives and scales [4, 5]. Visualizing engagements in this way is useful in a wide variety of contexts including historical mission review, mission planning, prebriefing, post-briefing, and live observation of mission training scenarios. The environment in a large-scale VR environment is shown in Fig. 1.

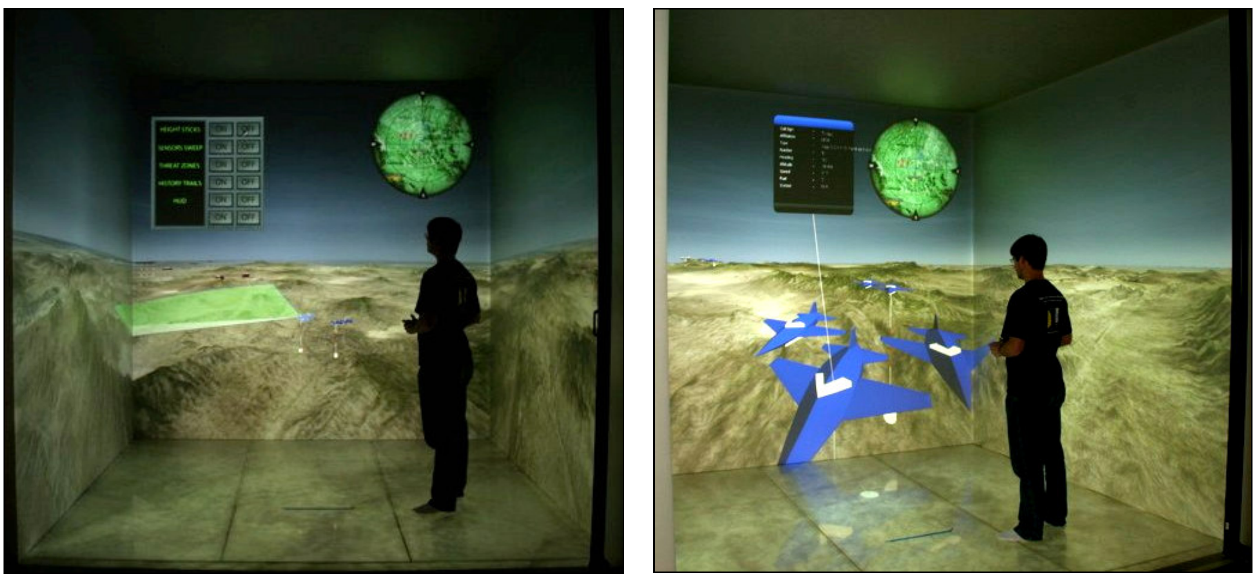

Figure 1. Virtual Battlespace in the C6 six-wall projection system at Iowa State University's Virtual Reality Applications Center.

B. Path Planning for UAVs

Real time dynamic path alteration is needed when a UAV is presented with an unexpected threat. For example, a UAV could encounter an unexpected surface-to-air missile (SAM) site. When this happens the operator must be alerted to this dangerous situation and be able to quickly re-task the UAV to reduce its threat exposure while considering other factors such as fuel usage.

It is important to consider the impact of the immersive environment on this process. In a conventional two-dimensional (2D) interface, the application would have to find some way to convey a 3D path in the 2D interface or restrict the path-planning algorithm to a 2D solution; limiting any alternative paths to changes in direction within the same elevation when in reality an aircraft could also change altitude to avoid threats. This limitation is lifted since the Virtual Battlespace operates in an immersive virtual reality environment, which allows true 3D interaction. As such, there is a need for a path planner application that functions in 3D space. With this tool, the operator can focus on the decision to be made as opposed to inferring the true shape of the path.

There has been extensive research in the area of path planning especially in the artificial intelligence, optimization, and video game communities but most have been restricted to a 2D form [6, 7]. One of the most popular path planning algorithms in the video game/artificial intelligence communities has been the A* path planning algorithm. The A* algorithm strength lies in the ability to heuristically judge or value the best path from point to point. If this cannot be done with reasonable accuracy then the A* method will not be very effective [8]. This is not possible considering the dynamic nature of the battlefield and the variable cost of particular parts of the environment based on differing criteria. Without human intervention, the path planning algorithms must be able to adapt to a variable mission environment. This has lead to research of great interest not only to UAV control but other fields such as robotics [9 – 13].

Without the ability for human intervention of the UAV's path, the Air Controllers and human pilots are solely responsible for maintaining a manageable airspace. Keeping a human in the loop helps prevent catastrophic mistakes by taking advantage of the human's ability to handle and process outside information. The human operators can issue overall objectives and commands to the vehicles under their control. The issuing of objectives as opposed to exact paths can reduce the amount of awareness needed to control an individual UAV. This reduction could result in more UAVs under the control of a single operator.

Because of the variable cost nature of the types of path planning that will be done with UAVs, a particle swarm optimization (PSO) method of path planning was developed. To maintain a human input in the decision making process, several paths are generated by the developed method. The generated alternate paths are represented by B-Spline curves to minimize computation, since a simple curve can be easily defined by as little as only three control points and this method has been successfully used to model constrained curves [14].

C. Particle Swarm Optimization (PSO)

To facilitate the search for optimal paths, a particle swarm optimization (PSO) technique was used to produce a large number of candidate paths for evaluation. PSO is a heuristic optimization method that is based on the movement of insect swarms introduced in the mid 1990s by Kennedy and Eberhart [15].

In PSO, an initial randomly generated population swarm (a collection of particles) propagates towards an optimal point in the design space, and reaches the global optimum over a series of iterations. Each particle in the swarm explores the design space based on the information provided by previous best particles. PSO then uses this information to generate a velocity vector indicating a search direction towards a promising design point, and updates the locations of the particles.

After reviewing the various current methods and research being done for path planning of UAVs, this paper presents a new method of path planning in 3D space using the PSO algorithm to generate alternate optimal paths.

III. Path Planning using Particle Swarm Optimization

The path planning process begins with identifying a target location for a specific UAV. Once the current and target positions are defined, this becomes the initial solution of the problem. From this initial solution, a search space is defined to scan and locate other UAVS within range and identify possible threats. The size of the search space is left open to the user's judgment, setting it too large will incur a longer computation time, while having a search space that is too small might cause some UAVs to be unaccounted for. Fig. 2 shows the process flowchart of the path planning using PSO.

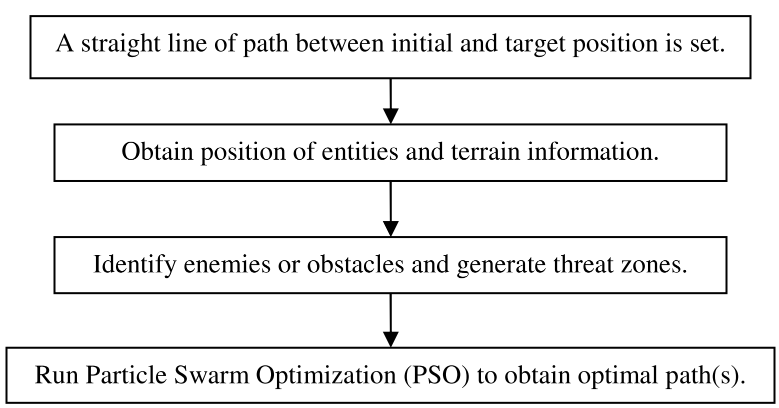

Figure 2. Flowchart of the path planning process using Particle Swarm Optimization.

Once position data of the UAVs within range are obtained, enemy entities are singled out and a 3D threat zone is generated for each of them. A threat zone is defined as a sphere (a hemisphere for ground vehicles) of radius R (user defined) surrounding the obstacle that the path needs to avoid. Threat zones are also generated for non-enemy (friendly) entities to avoid collision, but with a smaller radius. Fig. 3 illustrates this in a 2D manner, where the dot represents a UAV, surrounded by a threat zone.

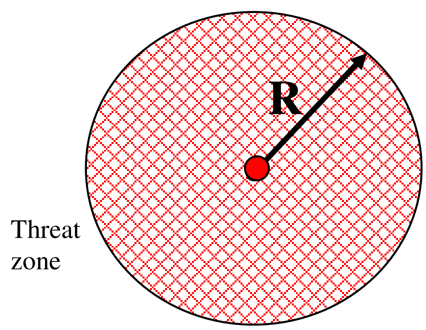

Figure 3: Two-dimensional illustration of a UAV's threat zone.

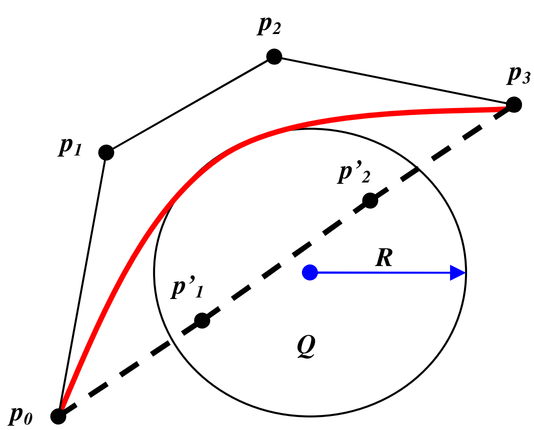

Figure 4: Two-dimensional illustration of a simple threat zone avoidance problem.

Formulation of the optimization cost function begins with the description of a B-spline curve to represent the path of the UAV. Consider the B-spline curve $\boldsymbol{p}(u)$ which interpolates end-points p_0 and p_3 and avoids $\boldsymbol{Q}$, and is illustrated by the red solid curve in Fig. 4.

The cost function components also depend on the number of parametric samples (line segments that form the curve) N that define the resolution of the curve. Here, N is user defined and the value of N brings a trade-off effect between accuracy of the curve and computational efficiency. Both cost function components are summations of the curve characteristics sampled at the regular parametric intervals,

$u_0 = 0$ and

$$u_i = u_{i-1} + \frac{1}{N-1}, \quad i = 1,2,....,N\text{-}1 \tag{1}$$

The initial solution to begin the optimization process assumes a straight line joining the end-points, with control points p'_1 and p'_2 that breaches the threat zone thus violating the constraint, illustrated in Fig. 4 by the black dashed line. By default, the number of interior controls points are defined as twice the number of threat zones. This is to ensure a smooth and continuous representation of the generated path. A new path can be computed by running the PSO such that the interior control points (between the end-points), which are set as the design variables for the optimization problem, satisfy the constraints. To achieve this, the cost function needs to accommodate the avoidance of threats as well as the length of the path, which will eventually lead to better fuel efficiency of the vehicle. The total cost function is represented by the following components:

$$C = K_1 C_T + K_2 C_L \tag{2}$$

where, C_T is the cost due to proximity of enemy entities and violation of the threat zones, and C_L reflects the cost incurred from excessive arc length and deviation from the original path.

The constants K_1 and K_2, in Eq. (2), are component weights that determine the relative emphasis of the various cost components with respect to the overall cost function. Each weight is normalized between zero and one. If a weight is zero then that particular cost function is unimportant for a particular run. All weights add up to 1.0 in total. These weighted cost components are then added together to form the total cost function of a particular path. If the threat component is of equal importance as the length of the path, both constants will be set equal or have a value of one. Table 1 shows an example of generating a set of three different alternate paths, each with its own preference.

Table 1. Example of component weights used to generate a set of alternate paths.

	Threat Weight, K_1	**Fuel Weight, K_2**
Threat Avoidance	0.95	0.05
Fuel Conservation	0.05	0.95
Blend	0.50	0.50

The threat component C_T requires a function to determine the distance from a point $\boldsymbol{p}(u_i)$ along the curve $\boldsymbol{p}(u)$ to a UAV inside the threat zone $\boldsymbol{Q}$ and is denoted here as $d(\boldsymbol{p}, \boldsymbol{Q})$. The function will return a positive value if there is a violation of the threat zone, and negative one otherwise. With this, the threat cost is then defined as:

$$d(\boldsymbol{p}(u_i), \boldsymbol{Q}) = (\text{Threat zone radius R}) - (\text{Distance between point } \boldsymbol{p}(u_i) \text{ and threat}) \tag{3}$$

$$C_T = \sum_{i=0}^{N-1} C_i, \quad C_i = \begin{cases} |d(\mathrm{p}(u_i), \mathrm{Q})|, & d(\mathrm{p}(u_i), \mathrm{Q}) \geq 0 \\ 0 & d(\mathrm{p}(u_i), \mathrm{Q}) < 0 \end{cases} \tag{4}$$

A significant violation of the threat zone will result in a significant increase in the threat component of the cost function. Since this simple zone violation constraint allows many possible solution curves with probably unacceptably large length, the second component will simultaneously minimize the curve arc length, thus providing a solution with the best fit possible along the obstacles.

The curve length component of the cost function is computed using a *chordal* approximation of the total curve length, L, relative to the initial solution obtained from a line connecting the endpoints. The curve length component is expressed as follows:

$$C_L = L - |\mathrm{p}_n - \mathrm{p}_0|, \quad L = \sum_{i=0}^{N-2} |\mathrm{p}(u_{i+1}) - \mathrm{p}(u_i)| \tag{5}$$

The curve length component translates as a difference between the generated path and the initial straight path. This represents the additional fuel expense from the alternate path. The goal is to generate a new path for the UAV that avoids a threat, with the lowest additional fuel expense simultaneously.

In the PSO process, each particle in the swarm represents a path defined by its own set of control points. As the interior control points (design variables) are moved around to generate new paths, the total cost for each new path is calculated. The particles propagate to a solution based on the best path (with the lowest cost) generated in that iteration, and the PSO process continues until convergence is reached at a final solution path.

IV. Implementation of PSO Path Planner into Virtual Battlespace

The purpose of an immersive command and control station is to permit the operator to focus on the overall mission status. As the number of aircraft under an operator's control increases, it becomes impossible to constantly monitor and manage every aircraft. To facilitate this, an alert subsystem was developed as part of the Virtual Battlespace to alert the operator of any issues. The alert subsystem plays a vital role in reassuring the operator that when UAVs run into situations that require user input, the operator will be made aware of them.

The alert subsystem, seen in the right image of Fig. 5, notifies the operator of the presence of an alert and when the operator chooses to examine an alert posted by a threatened UAV, the operator will see a variety of automatically generated path options. These path options will appear at a distance corresponding to a default value of 30 seconds ahead of the UAV's current position and reengage with the path when in a safe region. This lead-time can be adjusted by an operator. These points on the old path are used as the start and end points of the path-planning algorithm. All relevant threats and the start and end points are passed to the path-planning algorithm for it to calculate new candidate paths.

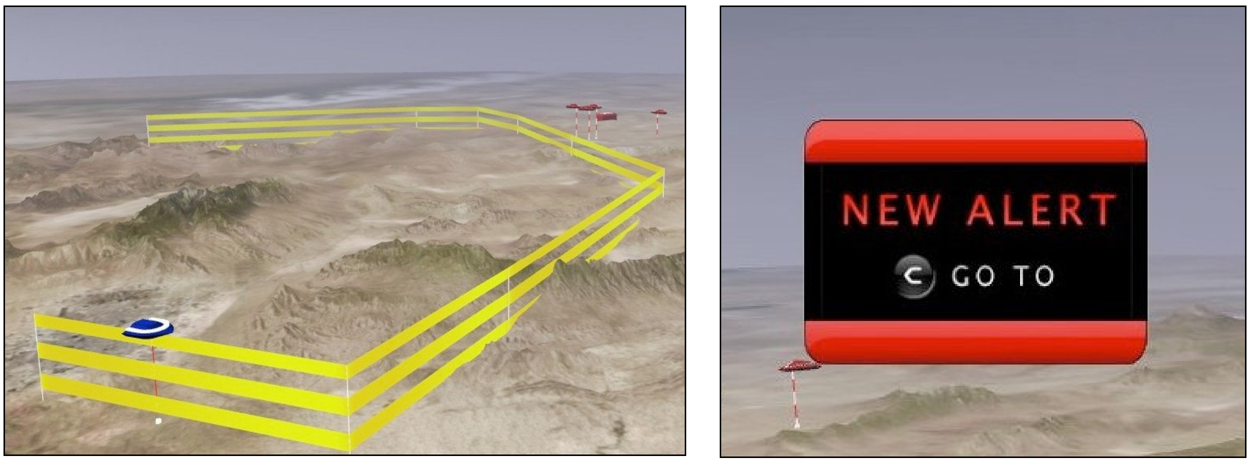

Figure 5. Illustration of a path in the Virtual Battlespace environment (left) and a threat alert display (right).

To generate the set of alternate paths, the PSO path planner is initiated to solve for three different sets of values for the component weights. For the purpose of this paper the parameter settings used were those in Table 1. However, an operator can adjust the available weights if additional paths for review are desired. One path is weighted towards fuel efficiency (minimal fuel expense for an alternate path), the second path makes threat avoidance the preference, and the last path is a middle ground between these options. The operator also has the option to vary these parameters to fit the mission objective. The importance of a UAV's current mission or future mission may demand that the UAV stay on its original path. For this reason the operator is provided with the option to cancel the alternate paths. Additionally, once the UAV is beyond the planned start point the UAV system will assume that it should keep its original path. The generated paths are represented in different colors and labels, as shown in Figs. 6-11. For the results from the simulated scenarios presented in this paper, the alternate paths are represented as follows:

Table 2. Color and label representations of generated alternate paths.

	Color	Label
Blend	Blue	A
Threat Avoidance	Green	X
Fuel Conserving	White	Y

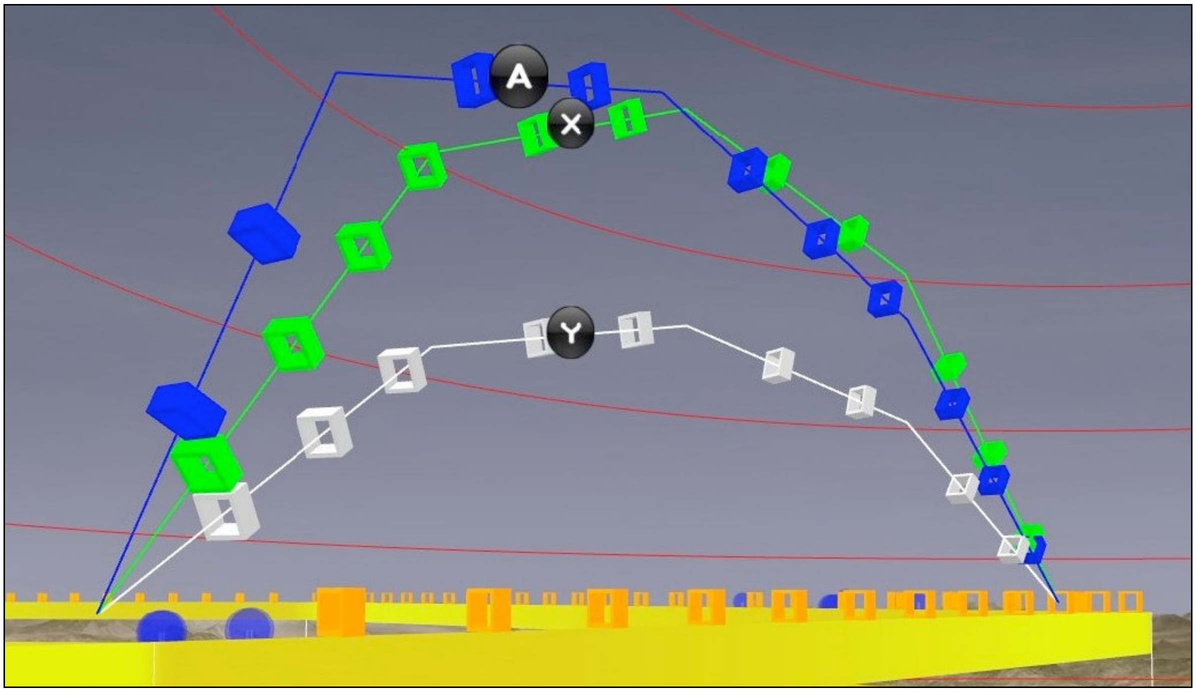

Figure 6: "Pilot" view of the three resulting alternate paths for a single threat zone.

In the "Pilot" view of the generated alternate paths in Fig. 6, the paths generated were successful in satisfying its objective. Threat zone hemispheres are marked as a set red rings. The blue path "A" represents the middle group blend where equal preference is placed on safety and fuel usage. Path "X" (in green) is the path that puts safety as a top priority but still tries to minimize fuel usage as a low priority, and Path "Y" (in white) is the exact opposite which minimizes fuel usage but with a low objective for safety. There is a distinct difference in paths "X" and "Y", as "X" plans a path that is of significantly higher altitude (safer) than "Y" which thrives on flying a shorter deviation from the original path to conserve fuel.

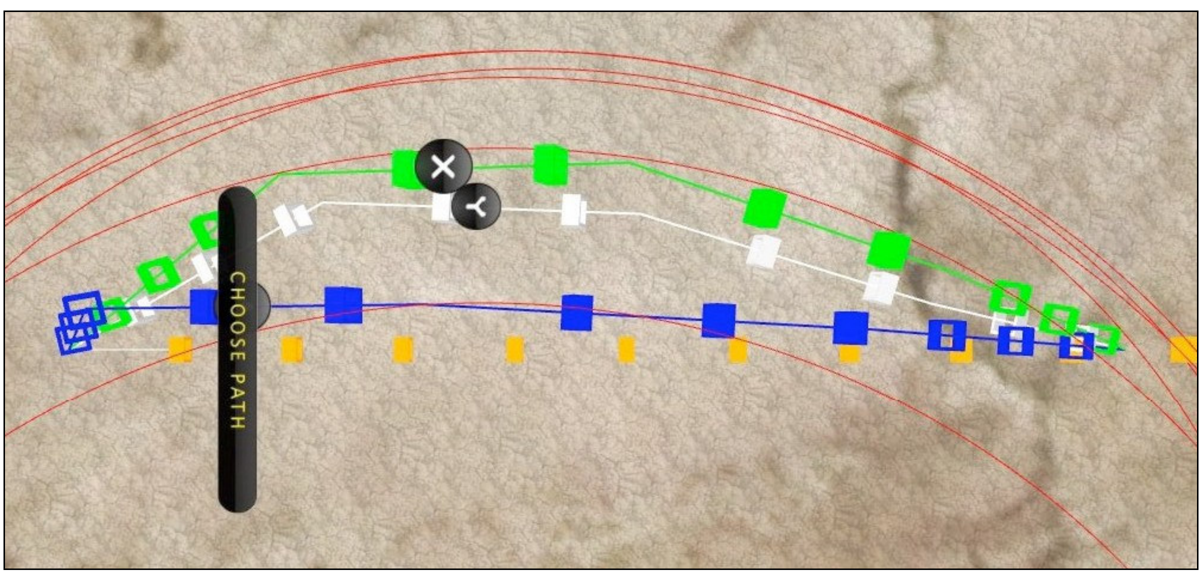

Figure 7. Top-down view of the three resulting alternate paths for a single threat zone.

The top-down view in Fig. 7 illustrates another view of the three alternate paths. When the alternate paths are ready to be inspected by a human operator, a “Choose Path” option prompt appears in the Virtual Battlespace menu system, and the operator can cycle between three preset views to inspect these paths. The operator also has the freedom to break away from the presets and move around the virtual environment to view these paths. This is an advantage that helps the operator in the decision making process. A human operator can evaluate these paths from different angles and select an appropriate path for a particular mission.

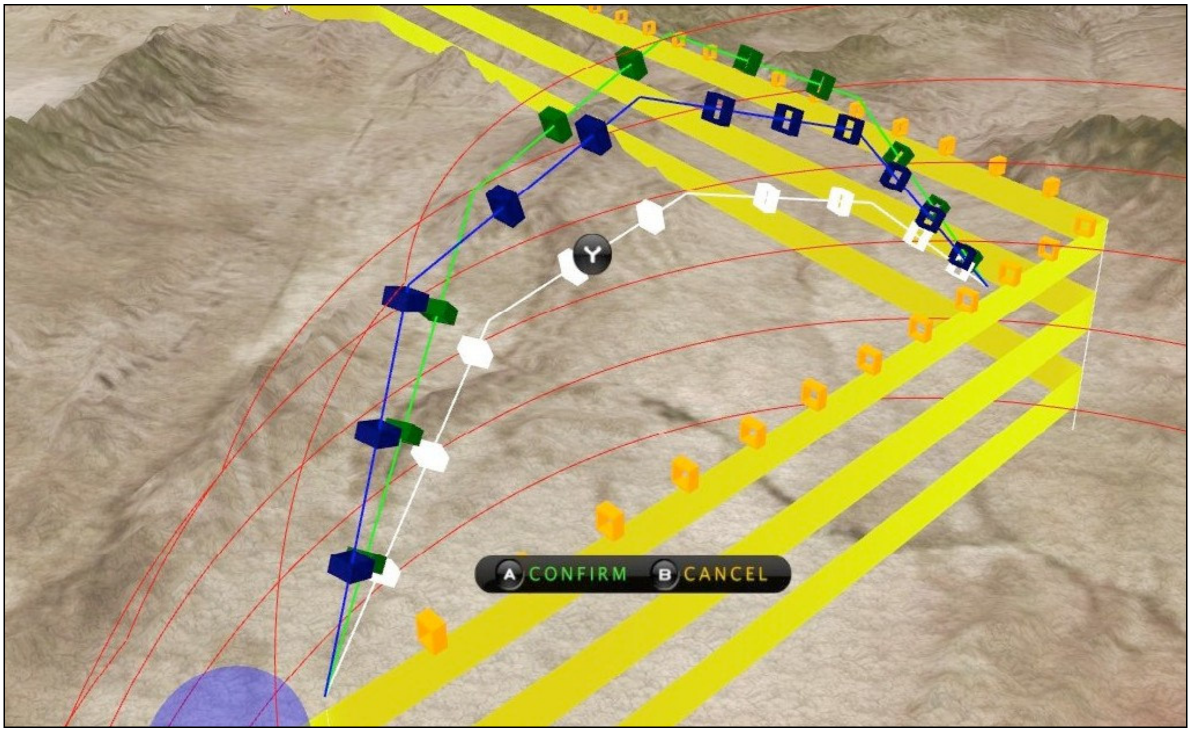

Figure 8. Highlight of selected path “Y”, with the two other alternate paths dimmed. Original path is also shown in yellow.

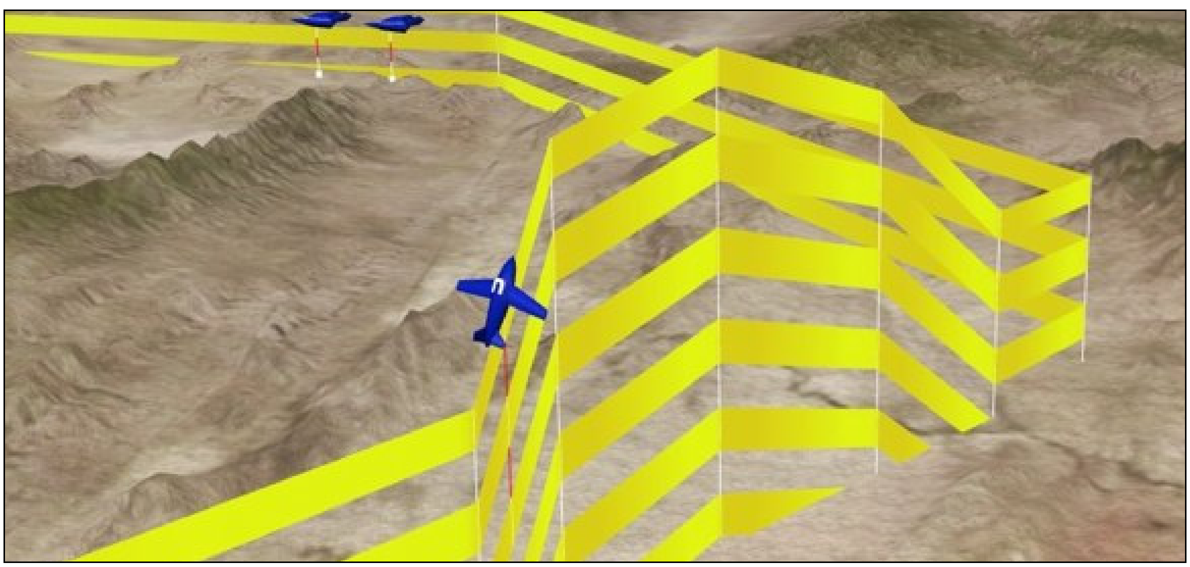

Figure 9. The selected alternate path replaces part of the original path.

Once the operator has selected a path, the path is highlighted and a prompt appears in the menu system asking for a confirmation as shown in Fig. 8. This two-step process for path selection is done as a preventive measure in case the operator makes a mistake in selecting the wrong path or decides to go with a different alternate path. When the selected alternate path is confirmed, the path planner replaces part of the original path of the UAV with the alternate path, as shown in Fig. 9.

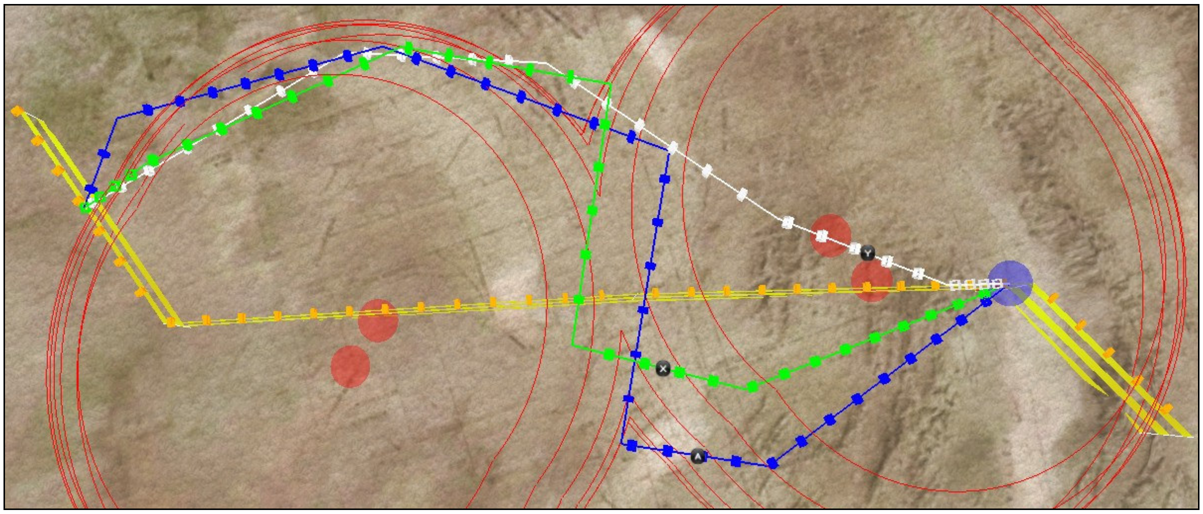

Figure 10. Top-down view of generated alternate paths for a simulated scenario with two distinct threat zones.

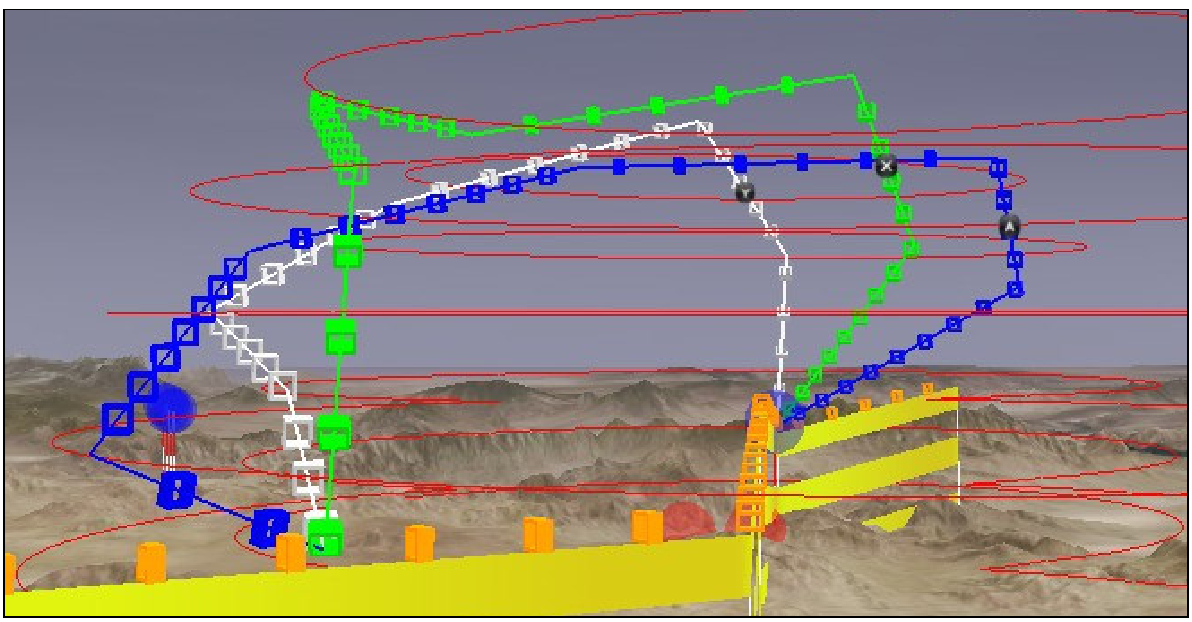

Figure 11. "Pilot" view of alternate paths for a simulated scenario with two distinct threat zones.

Fig. 10 presents the results from another simulated scenario with two distinct threat zones. Paths "A" in blue (blend) and "X" in green (threat avoidance) appear to be similar, as they "wrap" around the threat zones. However as seen in Fig. 11, "X" takes on a path of higher altitude as an extra precaution for threat avoidance. Path "Y" in white (fuel conserving) is one of a shorter distance compared to the other two as seen in Fig. 10, and is also of a lower altitude as seen in Fig. 11.

V. Conclusion

A three-dimensional path planner was developed to intelligently generate a set of alternate paths to be selected by an operator of a UAV. From the test cases using different simulated scenarios, the three-dimensional PSO path planner successfully generated alternate paths satisfying its respective objective as set in the component weight parameters. Most importantly, these paths were generated in real time.

By performing the path planning in three-dimensional space, the solution paths presented are more realistic to what UAVs are actually capable of performing. The option of selecting a particular path from a set of solutions ensures that the human factor is still part of the decision making process. With multiple views to evaluate the generated alternated paths allows the operator to make informed decisions based on the current mission objective. The feedback that has been received from experts within the field of UAV control seems to indicate that this is a relevant and interesting concept that warrants further investigation.

Pending implementations to the existing path planner involve the addition of a third component to the cost function, a component for target attraction. This will enable the path planner to account for certain locations that are required for the UAV to pass (e.g., a reconnaissance target). Another functionality to be added is to develop a path planner that is dynamic in nature to incorporate time as a variable when alternate paths are being generated, since the position of threats could change in the future of the alternate path.

Acknowledgments

This research was supported by the Air Force Office of Special Research.

References

[1]Barry, C.L., and Zimet, E. – UCAVs Technological, Policy, and Operational Challenges, Defense Horizons, *Center for Technology and National Security Policy, National Defense University,* No. 3, October 2001.

[2]US Department of Defense, Unmanned Aerial Vehicles Roadmap 2002-2027, December 2002, Available: http://www.acq.osd.mil/usd/uav_roadmap.pdf.

[3]Miller, C. – Trust in Adaptive Automation: The Role of Etiquette In Tuning Trust via Analogic and Affective Method. 1st International Conf. on Augmented Cognition, Las Vegas, NV; July 22-27, 2004.

[4]Walter, B.E., Knutzon, J.S., Sannier, A.V., and Oliver, J.H. – Virtual UAV Ground Control Station. *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, Chicago, IL, September 2004.

[5]Ruff, H.A., Calhoun, G.L., Draper, M.H., Fontejon, J.V., and Guilfoos, B.J. – Exploring automation issues in supervisory control of multiple UAVs. *Human performance, situation awareness, and automation: Current research and trends, Mahwah, NJ, Lawrence Erlbaum Associates, Inc*, Vol. II, pp. 218-222, 2004.

[6]Davis, L. – Warp Speed: Path Planning for *Star Trek:* Armada. *AAAI Spring Symposium AAAI Press*, Menlo Park, CA 2000.

[7]Frazzoli, E., Dahleh, M., and Feron, E. – Real-time Motion Planning for Agile Autonomous Vehicles. *Journal of Guidance, Control and Dynamics*, Vol. 25 no. 1, 2002.

[8]Stout, B. – Smart Moves: Intelligent Path Planning. *Game Developer Magazine*, July 1997.

[9]Li, S., Boskovic, J.D., Seereeram, S., Prasanth, R., Amin, J., Mehra, R.K., Beard, R.W., and McLain, T. – Autonomous hierarchical control of multiple unmanned combat air vehicles (UCAVs). *American Control Conference*, 2002.

[10]Beard, R.W., McLain, T.W. – Multiple UAV cooperative search under collision avoidance and limited range communication. *42nd IEEE Conference on Decision and Control*, 2003.

[11]Schouwenaars, T., De Moor, B., Feron, E., and How, J. – Mixed Integer Programming for Multi-Vehicle Path Planning. *European Control Conference*, 2001.

[12]Mitchell, I., Shankar Sastry, S. – Continuous Path Planning with Multiple Constraints. Technical Report 2003, Available: www.eecs.berkeley.edu/Pubs/TechRpts/2003/4157.html.

[13]Yang, H., Zhao, Y. – Trajectory Planning for Autonomous Aerospace Vehicles Amid Known Obstacles and Conflicts. *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004.

[14]Malhotra, A., Oliver, J.H., and Tu, W. – Synthesis of Spatially and Intrinsically Constrained Curves Using Simulated Annealing. *Journal of Mechanical Design,* March 1996.

[15]Eberhart, R.C., and Kennedy, J. – A New Optimizer Using Particle Swarm Theory. *6th International Symposium on Micro Machine and Human Science, Inst. of Electrical and Electronics Engineers,* Piscataway, NJ, 1995, pp. 39-43.