An Automatic UAV Search, Intercept, and Follow Algorithm for Persistent Surveillance

Gaemus E. Collins

Toyon Research Corporation, 6800 Cortona Dr., Goleta, CA, 93117, USA;

ABSTRACT

Substantial research has addressed the problems of automatic search, routing, and sensor tasking for UAVs, producing many good algorithms for each task. But UAV surveillance missions typically include combinations of these tasks, so an algorithm that can manage and control UAVs through multiple tasks is desired. The algorithm in this paper employs a cooperative graph-based search when target states are unknown. If target states become more localized, the algorithm switches to route UAV(s) for target intercept. If a UAV is close to a target, waypoints and sensor commands are optimized over short horizons to maintain the best sensor-to-target viewing geometry.

Keywords: Unmanned Air Vehicles, Search, Tracking, Sensor Management

1. INTRODUCTION

Substantial research has addressed the problems of automatic search, routing, and sensor tasking for Unmanned Air Vehicles (UAVs), producing many good algorithms for each task. But the path toward complete autonomy for UAVs will encounter missions that involve all of these tasks. A UAV operator may want to launch a UAV to search a region, then once a target is located, the operator would desire that the UAV track and follow ("bird-dog") the target to collect more data about. During both of these "modes" the UAV sensor(s) will need to be logically controlled: for search, perhaps the sensor should zoom out and look all around the UAV to cover the most area; while for target following, the sensor should focus on the target exclusively, and possibly zoom in. A third UAV mode considered in this paper is a target Intercept mode, in which target position and possibly velocity information is supplied from an external source, and the UAV is expected to intercept the target. Toyon's automatic UAV Search, Intercept, and Follow (SIF) algorithm uses switching logic to automatically transition between the Search, Intercept, and Following modes described above.

2. BACKGROUND

In previous work we have developed automatic UAV routing control algorithms for *target Search*, *target Intercept*, and *target Following*. We have also developed automatic UAV sensor tasking and sensor management algorithms that work in conjunction with these routing control algorithms. This section provides an overview of these existing algorithms, which are used as building blocks for the joint algorithm described in Section 3.

2.1 Automatic Target Search

Our Cooperative Graph Based Model Predictive Search (CGBPMS) algorithm jointly optimizes routes and sensor orientations for a team of autonomous agents searching for a mobile target in a closed and bounded region of $\mathbf{R}^2$. By sampling the area of interest (AOI) at locations with high target probability at each time step, the continuous search problem is reduced to a sequence of optimizations on a finite, dynamically updated graph. Paths are computed on this graph using a receding horizon approach, in which the horizon is a fixed number of graph vertices. To facilitate a fair comparison between paths of varying length on non-uniform graphs, the optimization criterion measures the probability of finding the target per unit travel time. Using this algorithm, we have shown that the team discovers the target in finite time with probability one. Simulations verify that the

Further author information:
e-mail: gcollins@toyon.com, Telephone: 1 (805) 968-6787

algorithm makes effective use of additional search agents. We have successfully hardware tested the CGBMPS algorithm in two small UAVs with gimbaled video cameras.

The CGBMPS algorithm is far too detailed to explain in detail here, but details and results for this algorithm can be found in our publications.[1–3]

2.2 Sensor Tasking

A gimbaled UAV sensor is automatically controlled by the SIF algorithm or one of the sub-algorithms. For a given UAV position $p_{\mathrm{UAV}}(t)$ at time t, a set of candidate sensor tasks are formed by partitioning the sensor Field of Regard (FOR) into individual Fields of View (FOVs). Then each candidate task is scored based on what falls within its FOV. Positive scores are assigned for a track in the FOV or some portion of the target probability density function (pdf). The highest task score is selected for execution by the UAV.

This sensor task optimization is performed for each UAV waypoint, and if waypoints are loosely spaced, the process will also be performed at regular intervals between waypoints. In our simulations and hardware tests, a new optimal sensor task is selected about every one second.

2.3 Target Tracking

When a target is detected in sensor data, the position and velocity estimate for the target is maintained in a target *track*. Target tracks can be updated with new sensor information using a data association algorithm and state estimation algorithm. A data association algorithm associates newly received detections with existing target tracks and include the Joint Probabilistic Data Association (JPDA) algorithm, 2D assignment algorithm, and multiple hypothesis tracking algorithm.[4–6] A state estimation algorithm processes the associated detections to estimate the state, such as position and velocity, of the target. Examples of state estimation algorithms include the Kalman filter,[7] particle filter,[8] or Interacting Multiple Model Estimator.[9]

Target tracks are also predicted forward in time using a dynamic motion model that attempts to capture the type of motion that the targets in track undergo. These same models are present in the state estimation algorithms, which must predict a track's future state before incorporating measurements. Examples of motion models include a nearly constant velocity (NCV) motion model that assumes the target moves forward with a constant velocity and small perturbations in acceleration (the unknown accelerations are modeled as a random noise process) and a motion model that assumes a target's motion is constrained to a road network. These prediction algorithms are also used by sensor tasking and platform routing algorithms that must plan routes into the future and, therefore, need estimates of a target's future state.

2.4 Target Intercept

A UAV routing algorithm for target Intercept is relatively straightforward. The algorithm inputs include the current estimated locations of all available UAVs, $\{p_{\mathrm{UAV1}}(t_0), p_{\mathrm{UAV2}}(t_0), \ldots p_{\mathrm{UAV}N}(t_0)\}$, and locations of all target tracks, $\{p_{\mathrm{Trk1}}(t_0), p_{\mathrm{Trk2}}(t_0), \ldots p_{\mathrm{Trk}M}(t_0)\}$.

First, an iterative path-generation algorithm is used to generate the intercept path from each UAV to each target. This path can not typically be computed directly, since the tracks are moving. For each UAV-track pair, the path-generation algorithm makes a guess at the future intercept time t_I, predicts the track location forward to t_I, and generates a straight-line path from the UAV_i to track_j:

$$P_{ij} := p_{\mathrm{Trk}j}(t_I) - p_{\mathrm{UAV}i}(t_0).$$

If the UAV flight time of this path is $t_I - t_0$, the path planner is finished with this UAV-track pair and records the path in an allocation matrix

$$A := [P_{ij}].$$

If the flight time of this path is more than $t_I - t_0$, the path planner generates a new estimate for t_I that is larger than the prior estimate, and repeats the process. (Similarly, if flight time of the path is less than $t_I - t_0$, the path planner generates a new estimate for t_I that is less than the prior estimate, and repeats the process.)

When the path-generation algorithm has converged on a path for each UAV-track pair, the allocation matrix A will be completely filled in. An allocation algorithm them parses the data in A to compute the optimal allocation of each UAV to a track. Optimality is determined by an objective function, which measures the expected reward of UAV$_i$ intercepting track$_j$, offset by the flight time to execute the $i - j$ intercept path.

2.5 Target Following

The seemingly easy problem of staying near a target to image it turns out to be somewhat challenging for very small fixed-wing UAVs. These UAVs typically fly at low speeds and low altitudes, and are dramatically hindered by wind and obstructions (such as buildings and trees) blocking the UAV's clear line of sight to the target. The range of permissible flight speeds for such platforms is often very narrow (e.g. minimum flight speed = 15 m/s, maximum flight speed = 20 m/s). If the target is moving slower than the minimum flight speed for the UAV, the UAV must maneuver with "S"-turns or loops to stay nearby the target.

Sensors onboard small UAVs may have additional limitations. The sensor may be fixed (e.g. side-look only), or the sensor gimbal may have a limited range of motion (e.g. biased to one side of the aircraft, as shown in Figure 1). Zoom capability may also be missing from these systems.

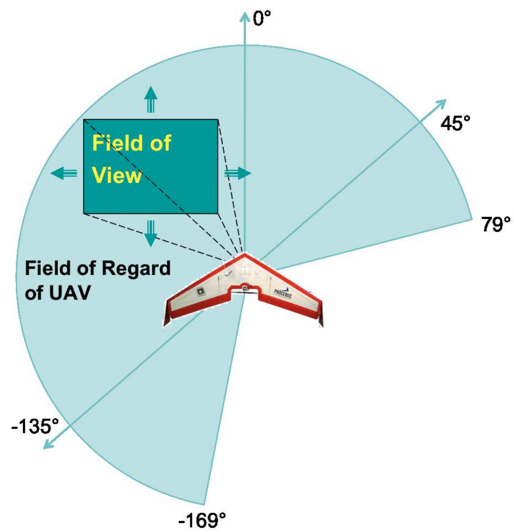

Figure 1. Example sensor field of regard and field of view for a small fixed-wing UAV

We have developed a "Sensor Guided Flight (SGF)" algorithm that optimizes sensor-to-target viewing geometry over possible future actions of the UAV and sensor. SGF is understood as the ability for a UAV's sensing system, primarily an imaging system, to automatically request a platform position and attitude that maximizes sensor performance. It includes the ability to monitor viewing conditions for sensor tasks, assess whether the sensor system parameters and platform position and attitude are optimal for those viewing conditions, and compute preferred parameters and platform state for best quality imagery for those viewing conditions. This is a particularly important capability for small UAVs to mitigate shortcomings of the platform and sensor such as narrow flight speed envelope, limited gimbal range, lack of zoom.

Toyon's SGF algorithm uses a joint UAV routing and sensor optimization technique that couples the platform and sensor control. First, candidate UAV waypoints are generated at several future time horizons. These waypoints provide a variety of future routing options while conforming to the platform dynamics (Figure 2). Next, candidate sensor tasks are generated for each future waypoint option. The collection of candidate tasks at a given waypoint can be generated by partitioning the azimuth and elevation ranges of the sensor FOR, or by targeting specific points on the ground (like the track locations). Then an expected reward is computed for each waypoint-task pair. The reward for a task comes from viewing a track or part of a target probability density function (pdf). Optimality here is measured in terms of maximizing the number of pixels on target while avoiding visibility obscurations, due to buildings, terrain, etc. The waypoint-task pair with the highest expected reward at each time horizon is selected for execution. However, the SGF algorithm will re-evaluate and possibly modify future waypoints and tasks after each pair is executed.

SAMPLE WAYPTS	Min Speed	Mean Speed	Max Speed
Right turn	Point 1	Point 2	Point 3
Straight	Point 4	Point 5	Point 6
Left turn	Point 7	Point 8	Point 9

Figure 2. A sample candidate waypoint set used by the SGF algorithm. This sample set will be generated over several different time horizons.

3. THE SEARCH, INTERCEPT, AND FOLLOW (SIF) ALGORITHM

The push toward higher levels of autonomy for UAVs drives development of more sophisticated control logic for these assets. The need for multi-modal control in complex missions is obvious. We envision a UAV operator employing an algorithm like the SIF to make UAV operation nearly autonomous. The operator could launch one or more UAVs, turn control over to the algorithm, and focus on the sensor data being collected. The operator would be relieved of teleoperations until it was time to land the UAV(s); and even the landing procedure could be automated.

Toyon's automatic UAV Search, Intercept, and Follow (SIF) algorithm uses switching logic to automatically transition between Search, Intercept, and Following modes based on its estimate of the world state. When UAVs are available to task, but no target tracks have been initiated, the SIF algorithm begins in Search mode. Search paths and sensor tasks are generated by the CGBMPS algorithm in Section 2.1. If a target is discovered by a UAV, the SIF algorithm switches to a Follow mode and generates future waypoints and sensor tasks using the SGF algorithm in Section 2.5 to keep the target in the sensor Field of View (FOV). Alternatively, if a target is discovered by an external sensor or information source, the UAV(s) may be very far away from the target track location. The SIF algorithm will switch to Intercept mode and use the track velocity estimate to plan an intercept path for the UAV using the algorithm in Section 2.4. The mode switching occurs automatically based on the existing or absence of track in the AOI, and based on the proximity of each UAV to each track. The mode and allocation of each UAV are re-evaluated periodically, so a UAV may switch modes several times during a single flight or mission.

The SIF algorithm was coded in C++ within Toyon's Cooperative Decentralized Asset Manager (CDAM) application. CDAM also contains the component Search, Intercept, and Following algorithms (described in Section 2) that compose SIF. CDAM interacts with Toyon's SLAMEM® simulation testbed, a high-fidelity entity-based battlefield simulation tool*.

Several CDAM-SLAMEM test scenarios were developed to test and evaluate the SIF routing algorithm. Both scenarios are set in Camp Roberts, CA, and include two Unicorn UAVs and four mobile ground targets (Figure 3). The Unicorns are launched near the Camp Roberts McMillan Airfield and loiter until they are routed and tasked by the SIF algorithm. The Unicorns can fly between 12 m/s and 18 m/s, and have a minimum turn radius of 50 m. The targets move along a road network (unknown to the SIF router) and speeds varying from 5 to 15 m/s. Typically the targets are moving slower than the minimum UAV airspeed.

In the first scenario, the UAVs are launched and tasked by the SIF algorithm to search a 12 km^2 area of interest (AOI) for the ground targets. The SIF algorithm begins in Search mode because the target locations are unknown to SIF. The UAV sensors are automatically tasked to look at regions of highest target probability.

*More information about SLAMEM appears in Appendix A

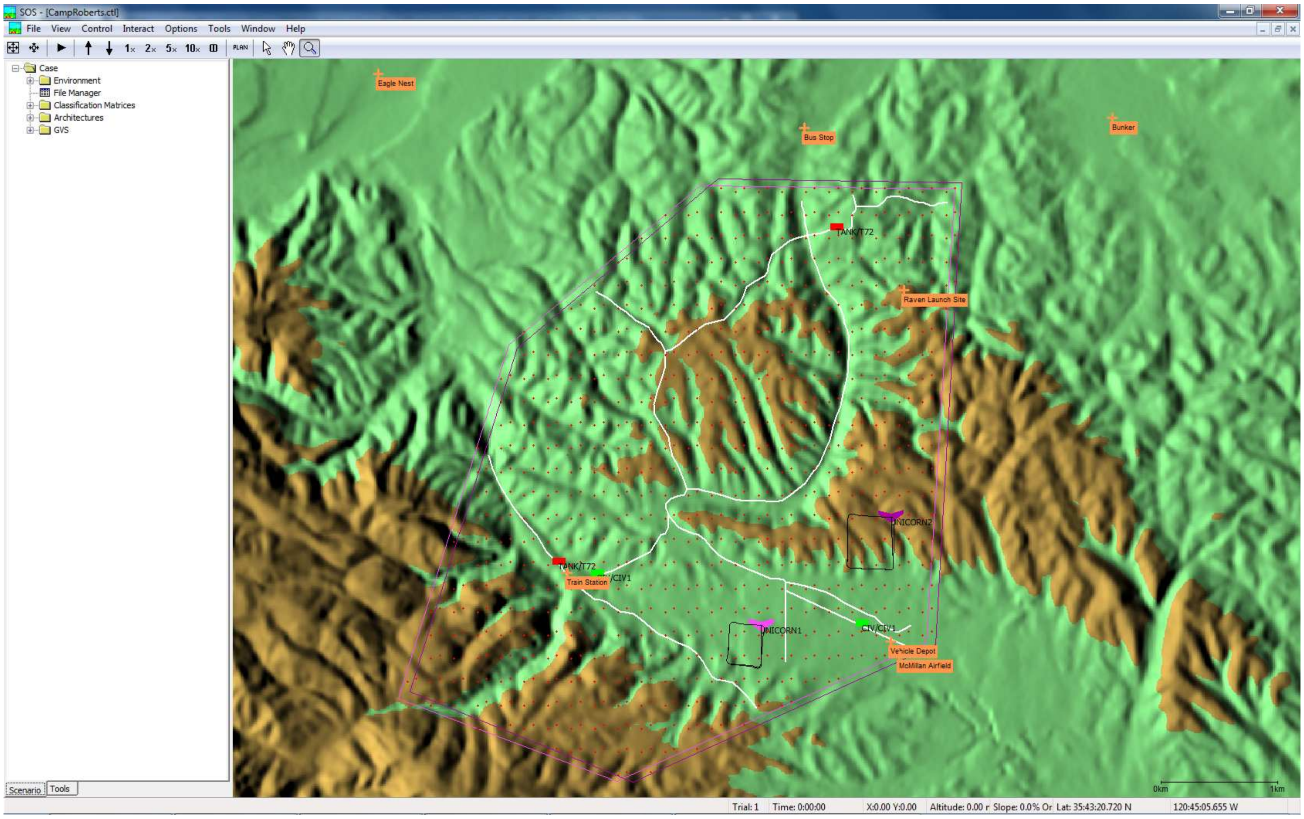

Figure 3. A SLAMEM screen capture showing 2 small UAVs (labeled “UNICORN 1” and “UNICORN 2”) in a default loiter pattern awaiting commands from the SIF algorithm. The future UAV flight paths are shown in black. This simulation scenario is set in Camp Roberts, CA, and includes a real road network (in white), 2 hostile vehicles (red rectangles), and 2 neutral vehicles (green rectangles). Each UAV is assigned its own area of interest (AOI), outlined in purple.

The Search algorithm generates a separate search graph for each UAV, and the graph becomes sparse as regions are searched. Figure 4 shows both UAV search graphs, keyed to the color of the UAV. Terrain drawing has been turned off to highlight the search graphs.

In Figure 5, both UAVs have located targets. The SIF algorithm has switched to Follow mode, and is using Toyon’s Sensor Guided Flight algorithm to plan future waypoints and sensor tasks for the UAVs. The allocation part of the SIF algorithm correctly allocated one UAV to each hostile (red) target.

The second scenario is a variant on the first in which a standoff Global Hawk was added to the Blue Force architecture. The Global Hawk scans the AOI with GMTI radar and discovers several moving targets. The radar detections are fused into track by a central tracker, and these tracks are sent to the SIF algorithm. The algorithm switches from Search to Intercept mode, allocates each UAV to the closest track, and plans an intercept path for the UAV. These paths are shown in black in Figure 6. Once the UAVs reach the tracks, the SIF algorithm switches to Follow mode.

4. CONCLUSIONS AND FUTURE WORK

We have presented a mode-switching algorithm for multi-mission autonomous UAV control. This algorithm includes 3 sub-algorithms for *target Search*, *target Intercept*, and *target Following*, and uses logic to automatically switch between the modes based on its perceived world state. This algorithm has been coded in C++ and successfully tested against high-fidelity battlefield simulation software.

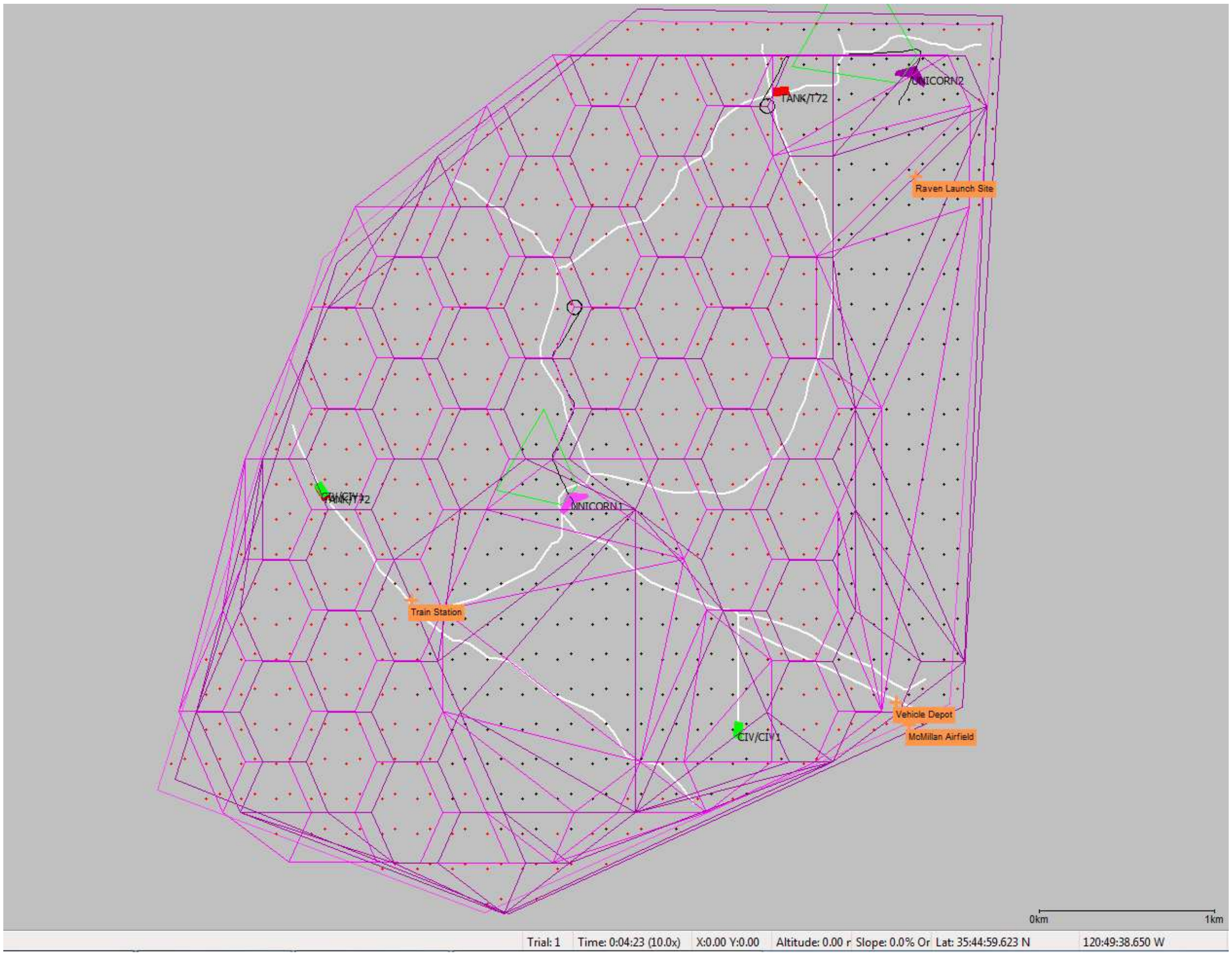

Figure 4. SIF algorithm automatically begins in Search mode when there are no tracks present. The Search algorithm for each UAV generates a search graph and automatically updates the graph based on where the sensor has looked. Regions with no graph vertices have been thoroughly searched. Future UAV flight paths follow the search graph to regions of highest target probability. Terrain has been turned off to highlight the search graphs and flight paths.

We believe this algorithm will help push the envelope of autonomy for small, fixed-wing UAVs, and enable these systems to maintain persistent ISR. We will continue to test this algorithm in simulation using more complex and challenging scenarios. An iterative refinement process will help prepare for flight testing this algorithm at Camp Roberts later in 2010.

APPENDIX A. SLAMEM

SLAMEM is Toyon's high-fidelity battlefield simulation tool, developed to analyze the performance of coordinated C4ISR and targeting systems against time-critical mobile targets. SLAMEM contains 6-DOF flight models for certain surveillance platforms, attack aircraft, and unmanned air vehicles (UAVs). These models enable us to evaluate how aircraft interact with other entities and the environment while executing a flightpath. SLAMEM models ground vehicles and targets (TELs, SAMs, force hierarchy, boats, background traffic); surveillance platforms (airborne, ground-based, space-based); sensor payloads (optical, radar, SIGINT); attack aircraft (fighters, helicopters, UCAV); and unmanned vehicles (Predator, Global Hawk, etc.). These entities are characterized by their system-level parameters. GMTI radar, for example, is modeled with a field-of-regard, beam scan rate,

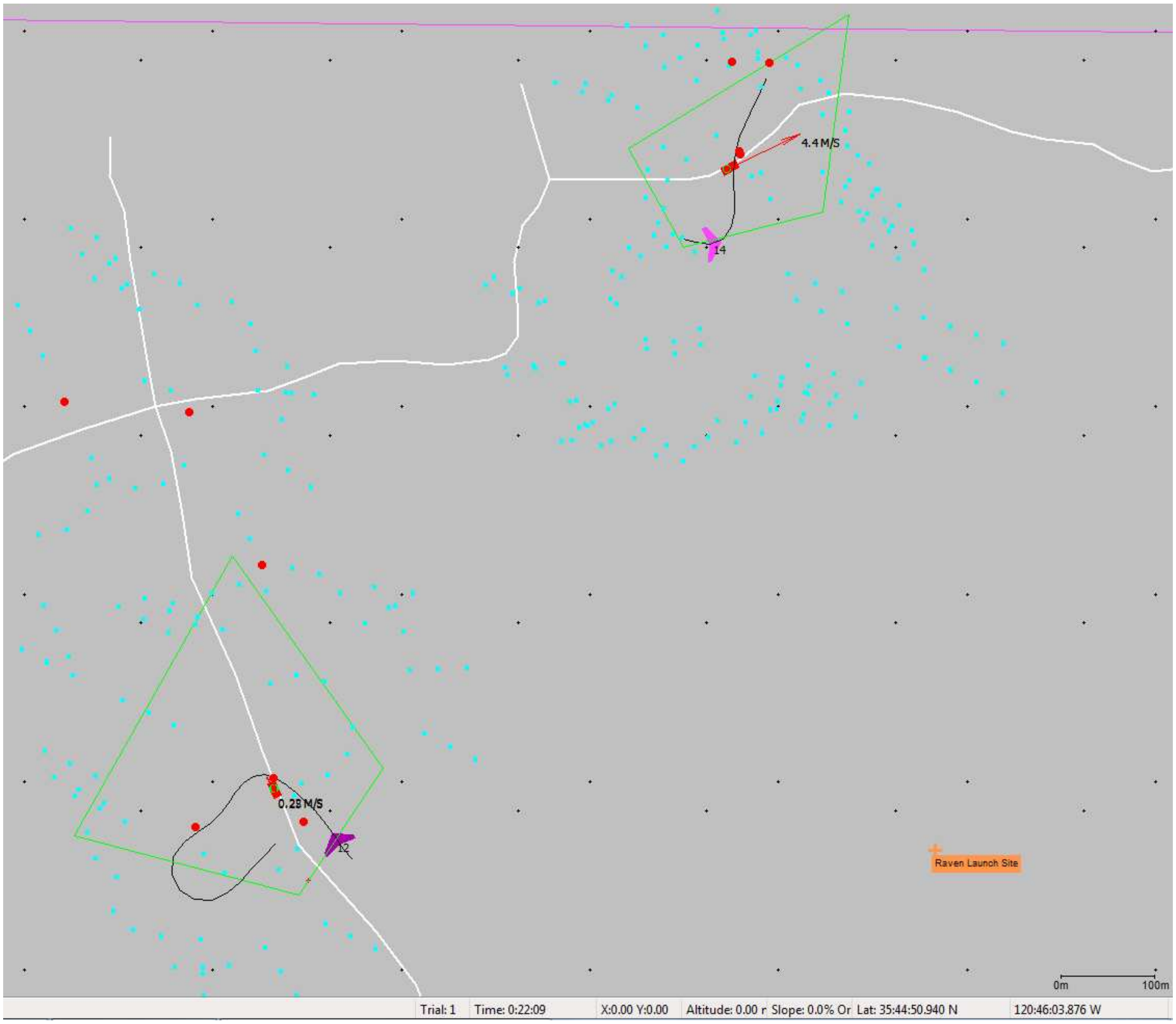

Figure 5. Both hostile targets have been discovered by the UAVs. The allocation algorithm has allotted one UAV per target. The SGF router is planning future waypoints (red dots) and sensor footprints (green quadrilaterals) for the UAVs to keep maintain best viewing geometry relative to the targets. The UAV speeds are drawn in black next to the UAV icons, and are much greater than the track speed estimates (displayed next to the target icons), so the UAVs must turn and loop to avoid outrunning the tracks.

processing time, and minimum detectable velocity. SLAMEM also models the command, control, and communication (C3) process of passing data for exploitation (detection and classification), multi-source fusion, sensor retasking, and attack nomination.

Specific geography is modeled in SLAMEM via Digital Terrain Elevation Data (DTED), road networks, foliage cover, and buildings. Buildings and foliage are modeled in 3D and used to perform clear-line-of-sight checks: a ray trace from sensor position to target that returns hard blockage (terrain, building, or vehicles) and soft blockage (foliage, clouds, fog, smoke). SLAMEM also incorporates realistic environmental factors and weather conditions.

SLAMEM has supported systems analysis for the US Army (ATEC, 10th Mountain Division), US Navy

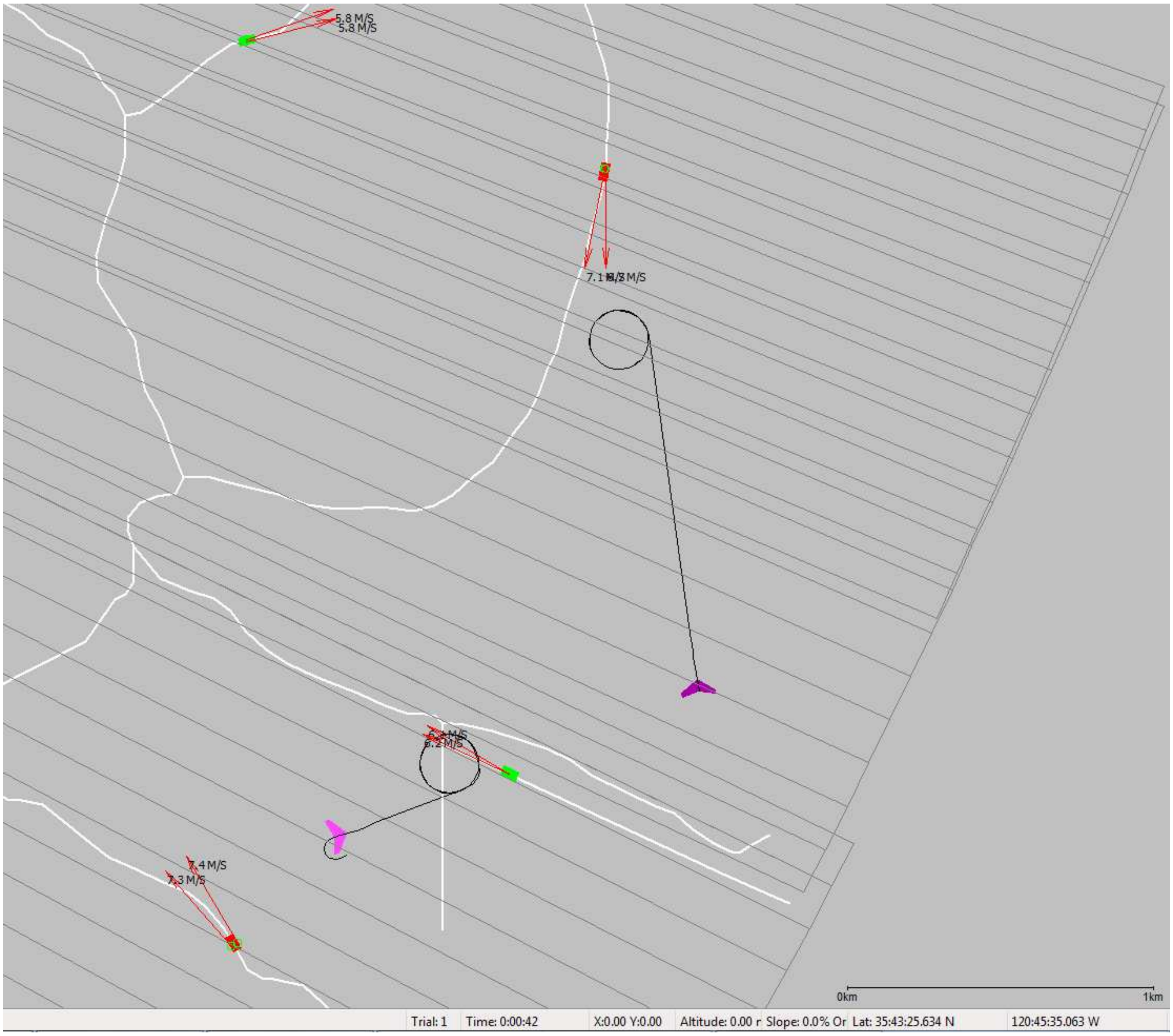

Figure 6. This variation of the SLAMEM Camp Roberts scenario includes a standoff Global Hawk with GMTI radar. Radar scan beams are shown in dark grey. Targets are detected by the radar and the target tracks are initiated by a centralized tracker. The track locations are sent to the SIF algorithm, which switches to Intercept mode. The Intercept algorithm considers all possible UAV-to-track allocations and computes Intercept paths for the most efficient allocation. The hostile and neutral targets are considered equal by the Intercept algorithm because the radar detected then with no classification. The optimal paths are sent to the UAVs.

(NRO, NRO DDSE, N81, SPAWAR), US Air Force (AFSPACE, AFRL, AFRL TUT, AFRL/SV), DARPA (IXO, IXO/Nettrack, IXO/VIVID, IXO/FOPEN), and USJFCOM (JUO, J9). Slamem has also supported interactive experimentation and war-gaming for Urban Resolve (UR) Joint Experimentation, USJFCOM Joint Experimentation Directorate (J9), National Reconnaissance Office (NRO), Army Mounted Maneuver Battle Lab (MMBL), Space Missile Defense Command, Naval Warfare Development Center, Synthetic Theater of War (STOW), DMSO Pegasus Federation, IDA Joint Advanced Warfare Program (JAWP), and OSD JTMD JTF (1998).

ACKNOWLEDGMENTS

Support for this work was provided by

a. Army Aviation Applied Technology Directorate (AATD) Phase II SBIR contract W911W6-10-C-0001, and

b. Army Institute for Collaborative Biotechnologies (ICB) 6.2 research contract W911NF-09-D-0001.

REFERENCES

1. G. E. Collins, J. R. Riehl, and P. S. Vegdahl, “A UAV routing and sensor control optimization algorithm for target search,” in *Unmanned Systems Technology IX*, G. R. Gerhart, D. W. Gage, and C. M. Shoemaker, eds., **6561**, p. 65610D, SPIE, 2007.
2. J. Riehl, G. Collins, and J. Hespanha, “Cooperative graph-based model predictive search,” in *Proceedings of the 46th IEEE Conference on Decision and Control*, December 2007.
3. J. R. Riehl, G. E. Collins, and J. P. Hespanha, “Cooperative graph-based model predictive search,” *Transactions on Aerospace and Electronic Systems* , 2010. Accepted for publication.
4. R. Singer, R. Sea, and K. Housewright, “Derivation and evaluation of improved tracking filters for use in dense multitarget environments,” *IEEE Transactions on Information Theory* **IT-20**, pp. 423–432, July 1974.
5. Y. Bar-Shalom and W. Blair, eds., *Multitarget-Multisensor Tracking: Applications and Advances, Volume III*, Artech House, Norwood, MA, 2000.
6. Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, Boston, MA, 1988.
7. Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, McGraw-Hill, New York, NY, 2001.
8. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Transactions on Signal Processing* **SP-50**, pp. 174–188, Feb. 2002.
9. H. Blom and Y. Bar-Shalom, “The interacting multiple model algorithm for systems with markovian switching coefficient,” *IEEE Transactions on Automatic Control* **AC-33**, pp. 780–783, August 1988.