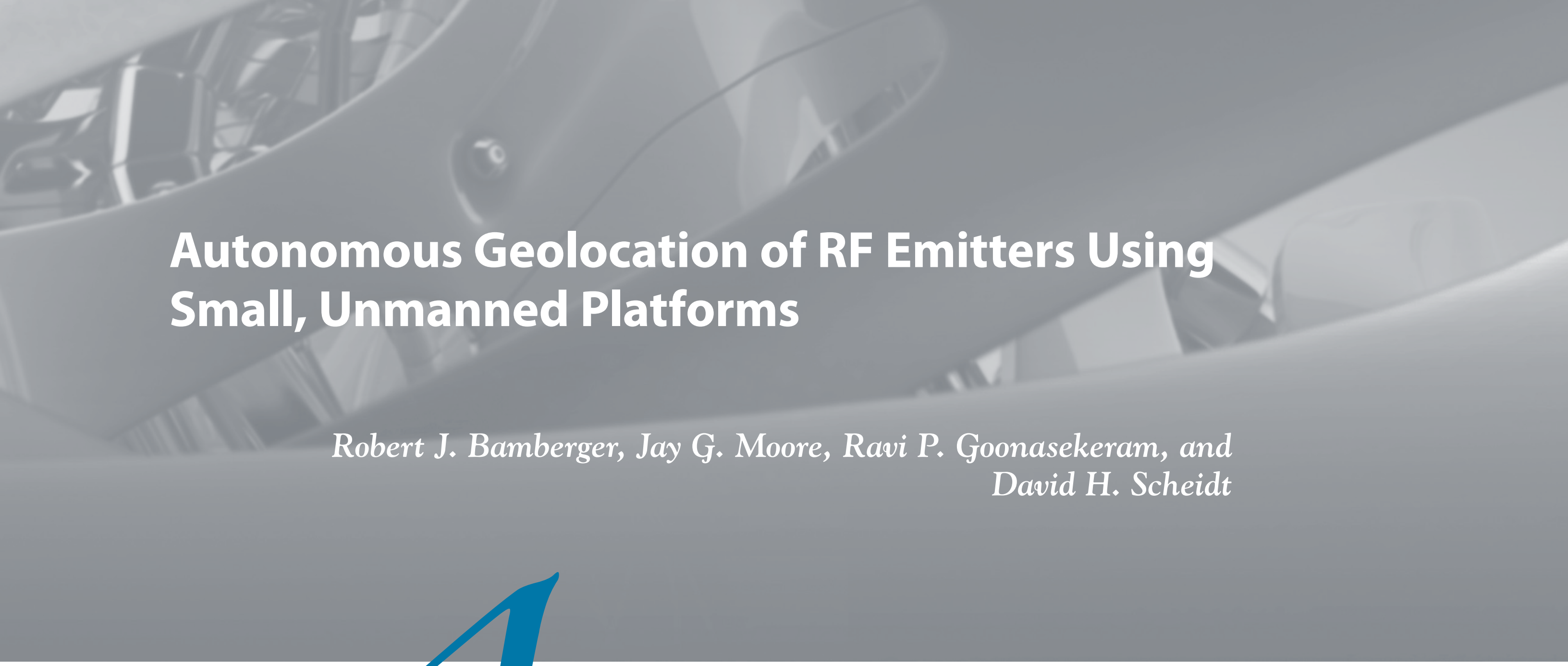

Autonomous Geolocation of RF Emitters Using Small, Unmanned Platforms

Robert J. Bamberger, Jay G. Moore, Ravi P. Goonasekeram, and David H. Scheidt

Autonomous geolocation of RF emitters using small, unmanned systems is a game-changing technology for military, government, and commercial missions. This technique employs a novel application of a common RF direction-finding technique called pseudo-Doppler. Emergent autonomous control concepts are used to control the sensor platform and optimize flight trajectories for efficient and rapid geolocation of the target. The basic components of this concept, from sensor development to unmanned system autonomous behaviors, were tested in simulation and subsequently demonstrated in flight during the Tactical Network Topology experiment.

OVERVIEW

The mission set for geolocation of RF emitters extends across military, governmental, scientific, and commercial domains. Examples include finding adversaries' radios, tracking RF-tagged wildlife, and searching for shipboard emergency beacons. Another application, to be addressed with future research, is the use of this technique for navigation of small unmanned aircraft systems (UASs) in areas where Global Positioning System (GPS) is jammed. Classic methods for RF-emitter geolocation include ground-based direction-finding (DF) techniques and Doppler-based airborne techniques. However, ground-based systems can be labor intensive, often require long-duration sensing, and may not have the ability to access areas that result in a solution. Doppler techniques onboard high-value airborne assets, whether manned or unmanned, require human-in-the-loop control and use sensor platforms from higher echelons that may be too expensive or difficult to schedule and typically lack stealth.

APL has developed a technique using small, inexpensive, autonomous UASs that can be organic to a squad of soldiers, a small team of first responders, or private commercial interests. Furthermore, this approach requires only supervisory control but no human in the loop for vehicle control, sensor processing, or geolocation solutions. Because the sensor platforms are small, they can be more easily made stealthy, and because they are airborne they can access areas from which ground-based systems may be restricted.

Small UASs are true force multipliers, providing soldiers or first responders organic mobile sensor platforms that can provide close-in sensing while remaining stealthy. These next-generation UASs work individually or as autonomous, multi-vehicle collaborative units and can operate as "fire and forget" resources requiring very little human intervention for control (usually only takeoff and landing). APL's unique physics-based approach to UAS autonomy has been successfully demonstrated in dozens of flight test experiments, with mission objectives ranging from unattended ground sensor (UGS) data exfiltration and relay, to vision-based ground vehicle tracking, to chemical-biological agent plume characterization.[1, 2]

However, small UASs do not fly at velocities sufficient to produce Doppler shifts that result in accurate geolocation solutions. One method that does not require a high-velocity platform is known as pseudo-Doppler, also called synthetic Doppler, which is employed by both amateur radio enthusiasts and law enforcement.[3] In the pseudo-Doppler approach, the receive signal is switched rapidly between a constellation of antennas, and the phase difference is measured to determine line of bearing (LOB).[4, 5]

For this effort, multiple receive antennas and a sensor payload were integrated onboard each UAS. COTS hardware was adapted to receive the signals, and custom hardware was developed to switch between antennas and to measure the phase shift that produced LOBs. Miniaturization of the hardware enabled implementation onboard a small UAS platform (less than 160 cm wingspan and less than 3.2 kg gross vehicle weight).

APL's autonomy algorithms enable multiple UASs, each deriving its own LOB solution, to optimize their own flight trajectories. Kalman filters implemented on each aircraft were used in the derivation of the geolocation solution and error ellipse based on the LOBs of the UAS team. Thus, each UAS was equipped with its own processing module to implement the autonomy and derive the geolocation solution, as well as a communication module to exchange data with the other UASs and send the solution information to the user on the ground. Each UAS also had an onboard autopilot that supported true unmanned flight of the UAS solely on the basis of the direction provided to it by the onboard autonomy software; no human in the loop was required to direct the UAS flight.

ONBOARD KALMAN FILTER AND GEOLOCATION SOLUTION PROCESSING

To obtain a target geolocation solution, it is necessary to combine two or more LOB measurements from different look angles to the target RF emitter.[6] This can be accomplished with a single sensor platform, but the solution is arrived at more quickly, and the error smaller, using two or more platforms (balancing resource constraints with the geolocation solution error, three vehicles seem to be optimal for this application). This is especially true for moving RF emitters.

For this effort, two UAS vehicles were used as the sensor platforms (although not optimal, two were considered adequate for proof of concept, although one can also provide a solution). Each vehicle implemented an onboard Kalman filter for fusing LOB measurements into a geolocation solution. Each vehicle broadcast its measured LOB values, along with the time and location of the measurements, to the wireless network. This allowed each vehicle to independently fuse its own measurements with the measurements of the other UASs. No terrain elevation information was available to the vehicles, so a flat-earth assumption was made, and the target RF emitter geolocation was computed in the horizontal plane only. Because the flight altitudes were low relative to standoff distance, altitude was assumed to be the same as the terrain elevation at the vehicle launch site. Each filter was initialized using the first available LOB measurement and an assumption about the maximum range at which the target might be detected. This information was used to construct an uncertainty region oriented along the LOB, with a 95% confidence interval in the range direction reaching from the vehicle position to the assumed maximum range.

Because the process by which the vehicle and target emitter positions define a LOB (i.e., the observation model) involves nonlinear trigonometric functions, a linear Kalman filter was not suitable for the task of fusing LOBs. In early simulation, an extended Kalman filter, which linearizes a nonlinear process model about some operating point, was implemented and tested but was found to diverge egregiously under conditions of high noise because of the degree of nonlinearity. An unscented Kalman filter, which uses a limited form of sampling to approximate nonlinearities to second-order accuracy, was found to converge reliably even when the LOB measurement noise reached a standard deviation of 90°.

A constant process model was used in the Kalman filter because, for these experiments, the target was nominally stationary. However, a non-zero process noise matrix was used. This served two purposes: first, it allowed the filter to better recover from unmodeled biases in the LOB measurements by discounting old measurements in favor of new ones; and second, it allowed for the tracking of moving targets, even with no prior assumptions about the direction or pattern of motion. Tests in simulation demonstrated success in tracking a moving target both with a constant speed and direction and with randomly changing direction.

LOB measurements were retained in a buffer in chronological order for a short time before they were fused in the filter. This allowed time for the data from all the UASs to arrive and be inserted into the buffer in the correct order before being fused, eliminating the need to "roll back" the filter to an earlier time to incorporate older data. Fused measurements were retained in the buffer for another short period so they could be broadcast to the network multiple times, reducing the chances of lost data and increasing the degree of agreement between the vehicle target estimates. Each UAS uses its own LOB measurements, as well as the LOB measurements of the other UAS, to estimate target location.

MISSION-BASED AUTONOMY CONCEPT AND SOFTWARE DESIGN

Autonomy Concept

Previous APL research efforts pioneered the use of potential field theory to achieve an effects-based control of multiple UASs.[7] This approach is based on insect models of cooperation and coordination. Problems are solved by heterarchically, rather than hierarchically, organized swarms that alter and react to the environment.[8,9] Instead of centralized control, decision making is decentralized, occurring with each member of the swarm.

This sharing of knowledge is the cornerstone of the potential fields concept. The transmitted data packets contain information such as the vehicle's situational awareness (e.g., sensor data), operational status (e.g., location in space), or mission parameters (e.g., user commands). These data packets are communicated over a wireless local area network (WLAN) and are referred to as "beliefs." A communications framework was developed for belief transfer that employs a modular multilayered architecture. This framework was designed to facilitate distributed collaboration over any mobile ad hoc network (MANET).

The potential fields are generated as a result of the worldview of the UAS, which is itself the totality of the UAS beliefs. These fields are used to influence vehicle action, most notably movement. The forces associated with these fields, which are illustrated in Fig. 1, may be attractive (directing the vehicle toward a point), repulsive (directing the vehicle away from a point), or complex (a combination of attractive and repulsive fields). At any given time, the total force on a vehicle is the summation of all attractive, repulsive, and complex forces due to all known influences.

Software Design

The Java-based software implementation of the UAS autonomy was developed from a system of related subsystems, including the agent system and the belief network interface.

Agent System

At the center of the implementation is the agent system. This subsystem has interfaces to the sensor interface, the autopilot, the Kalman filer, and the belief network. It acts as a data conduit and processing system. The UAS behaviors are also implemented in this subsystem.

Belief Network interface

The agent system interfaces with a virtual blackboard known as the belief network. This blackboard is made up of all the belief managers spread across the network. The belief managers attempt to automatically and efficiently synchronize and update the beliefs held in the blackboard. For this effort, two sensor beliefs were added to the legacy belief network. They represent the LOB output from the onboard sensor package and the uncertain target geolocations. These are called "RangeBearingSensorBelief" and "UncertainTargetBelief," respectively.

RangeBearingSensorBelief represents a time-indexed list of all sensor readings performed by any agent. For efficiency, the belief drops any sensor reading older than a certain decay time. This decay time is configurable at run-time. UncertainTargetBelief holds the results of the sensor data beliefs and geolocation uncertainties of each individual agent. The geolocation uncertainty is represented by an error ellipse about the derived geolocation solution. This belief is used to display ellipses on a modified version of the standard display tool.

Custom Optimization Behaviors

Two custom behaviors were created for this effort: GhostCircularFormationBehavior and AngularDiversityTrackBehavior. Each behavior attempts to guide the aircraft in a certain direction. The agent system combines the results of these behaviors into a single command sent to the autopilot. At certain times, the system weighs some behaviors more than others, on the basis

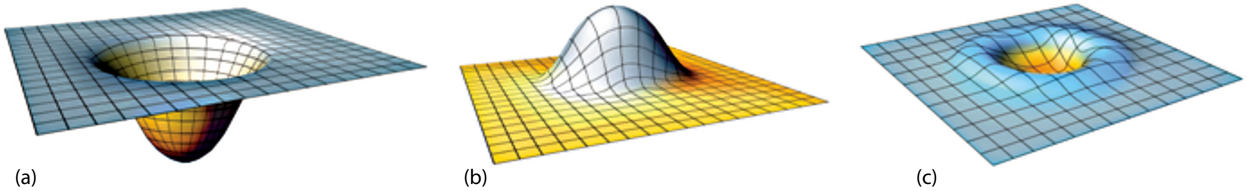

Figure 1. Fields that are (a) attractive, (b) repulsive, and (c) complex.

of position and objective. For this effort, two modes were developed, one corresponding to each behavior: if the UAS was within a certain distance of its own target, it would enter the orbit mode; if the UAS was farther away, it would enter into the homing mode. The orbit mode weighted the output of GhostCircularFormationBehavior fully, whereas homing mode weighted AngularDiversityTrackBehavior.

The GhostCircularFormationBehavior behavior attempts to orbit a target while maintaining ideal angular separation of the orbiters. Simulation runs resulted in an ideal orbiting angle of 90° phase difference for two UASs, 120° phase difference for three planes, etc. One way to implement this is to have the "ghost" of each vehicle reflected through the target and equally space all the planes including the ghosts. This method extends a behavior known as CircularFormationBehavior, which attempts to have a set of planes orbit a target with equal separation.

The AngularDiversityTrackBehavior is intended to have two or more planes approach a target and maintain a balance between maximum angular diversity and target proximity. This behavior detects other planes that are also in this mode and steering a course that depends on the angular separation between the two planes. The relationship between the angular separation and each vehicle's steering command is developed from a simplified representation of the geolocation geometry. Within the simplifying assumptions, the steering commands maximize the instantaneous improvement in geolocation accuracy for two cooperating vehicles, and the resulting behavior generalizes in an intuitive way to more than two vehicles. The simplified geometry for two-vehicle geolocation is depicted in Fig. 2.

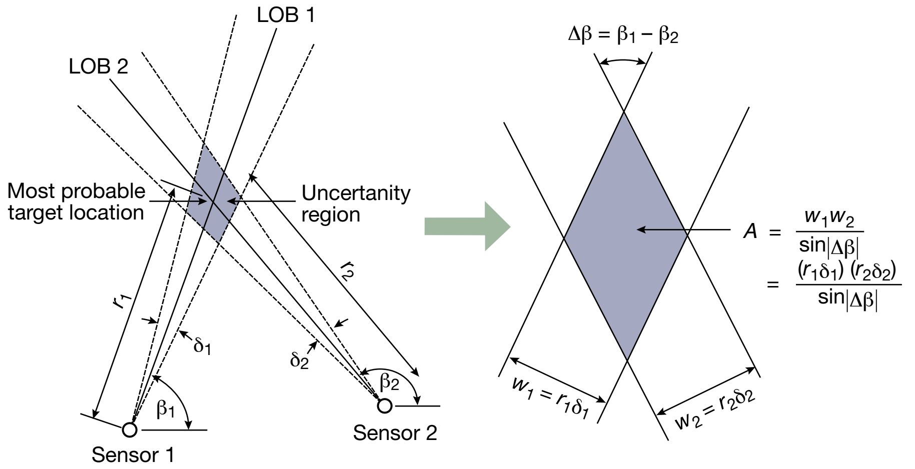
LOB 1
LOB 2
Most probable target location
Uncertainty region
r_1
r_2
δ_1
δ_2
β_1
β_2
Sensor 1
Sensor 2
$\Delta\beta = \beta_1 - \beta_2$
$A = \frac{w_1 w_2}{\sin|\Delta\beta|} = \frac{(r_1\delta_1)(r_2\delta_2)}{\sin|\Delta\beta|}$
$w_1 = r_1\delta_1$
$w_2 = r_2\delta_2$

Figure 2. LOB from two airborne sensors.

Two UASs labeled Sensor 1 and Sensor 2 are shown on the left-hand side of the figure independently measuring LOB to the target with some error, resulting in the uncertainty area, shaded blue, within which the target is most likely to be found. This geometry has been simplified in the right-hand side of the figure to approximate the area of uncertainty as a trapezoid, greatly simplifying the expression for the area of the uncertainty region, A:

$$A = \frac{w_1 w_2}{\sin|\Delta\beta|} = \frac{(r_1\delta_1)(r_2\delta_2)}{\sin|\Delta\beta|} . \qquad (1)$$

The control law will choose the direction (course) that each UAS will travel, denoted by ψ_1 and ψ_2; the UAS velocities V_1 and V_2 are assumed fixed (and will be shown later not to impact the computed course). The difference between ψ and β is defined as θ. This geometry is illustrated in Fig. 3.

If it is the case that the signal might be lost at any time, the best policy is, at each point in time, to choose to steer in the direction that maximizes the rate at which

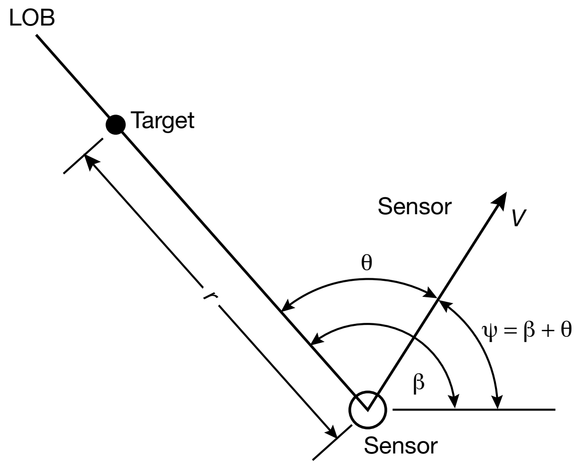

Figure 3. Steering geometry definitions.

the uncertainty shrinks, because the next sensor reading may be the last. At each iteration of the control law, the courses will be chosen so as to maximize the rate at which A shrinks, that is, we wish to minimize $\dot{A}$, the time rate of change of A, with respect to ψ_1 and ψ_2. First we must derive an expression for $\dot{A}$:

$$\begin{aligned}\dot{A} &= \delta_1\delta_2\frac{(r_1\dot{r}_2 + r_2\dot{r}_1)\sin|\Delta\beta| - r_1r_2\cos|\Delta\beta||\Delta\dot{\beta}|}{\sin^2|\Delta\beta|}\\ &= \frac{\delta_1\delta_2}{\sin^2|\Delta\beta|}[(r_1V_2\cos\theta_2 + r_2V_1\cos\theta_1)\sin|\Delta\beta| \mp r_1r_2\cos|\Delta\beta|(\dot{\beta}_1 - \dot{\beta}_2)]\\ &= \frac{\delta_1\delta_2}{\sin^2|\Delta\beta|}\left[(r_1V_2\cos\theta_2 + r_2V_1\cos\theta_1)\sin|\Delta\beta| \mp r_1r_2\cos|\Delta\beta|\left(\frac{V_1\sin\theta_1}{r_1} - \frac{V_2\sin\theta_2}{r_2}\right)\right]\end{aligned} \quad . \qquad (2)$$

To choose ψ_1, we set $\partial\dot{A}/\partial\theta_1 = 0$ and solve for θ_1:

$$\begin{aligned}\frac{\partial\dot{A}}{\partial\theta_1} &= \frac{\delta_1\delta_2}{\sin^2|\Delta\beta|}\left[r_2V_1\sin\theta_1\sin|\Delta\beta| \pm r_1r_2\cos|\Delta\beta|\frac{V_1\cos\theta_1}{r_1}\right]\\ &= 0\\ r_2V_1\sin\theta_1|\Delta\beta| &= \pm r_1r_2\cos|\Delta\beta|\frac{V_1\cos\theta_1}{r_1}\\ \tan\theta_1 &= \pm\cot|\Delta\beta|\\ \theta_1 &= \begin{Bmatrix}\Delta\beta + \frac{\pi}{2} & \Delta\beta > 0\\ \Delta\beta - \frac{\pi}{2} & \Delta\beta < 0\end{Bmatrix}\end{aligned} \quad . \qquad (3)$$

The steering command is then:

$$\begin{aligned}\psi_1 &= \beta_1 + \theta_1\\ &= \beta_1 + \Delta\beta \pm \frac{\pi}{2}\end{aligned} . \qquad (4)$$

An analogous process is used to find ψ_2. When $\Delta\beta = 0$, the vehicles will each choose at random whether to use $\theta_i = \Delta\beta + \pi/2$ or $\theta_i = \Delta\beta - \pi/2$. This situation is extremely unlikely to arise in real-world applications, however, and is really a concern only in simulation where the vehicles can be initialized to exactly the same position.

Notice that θ_1 depends only on $\Delta\beta$, so the velocities, distances from the target, and uncertainty in the LOBs do not affect the course command.

This control policy for two vehicles can easily be generalized to more than two vehicles by using a potential field approach. In concept, the procedure is to express the influence of each vehicle on every other as a potential field, sum the fields, and steer each vehicle down the gradient of the potential field. In practice, however, it is easier not to explicitly construct the potential field but to work directly with the gradients from the start; because gradient is a linear operator, the gradients resulting from the influence of each vehicle on a given teammate can be found individually and summed.

If the potential field gradient due to the influence of vehicle j on vehicle i is $\nabla\phi_{i,j}$, then the potential field equivalent of the above control law is:

$$\nabla\phi_{i,j} = \cos\theta_{i,j}\hat{x} + \sin\theta_{i,j}\hat{y}, \qquad (5)$$

where $\theta_{i,j} = \beta_i - \beta_j$.

The influence of all $N - 1$ other vehicles on vehicle i is:

$$\begin{aligned}\nabla\phi_i &= \sum_{j=1}^{N} w_{i,j}\nabla\phi_{i,j}\\ &= \sum_{j=1}^{N} w_{i,j}(\cos\theta_{i,j}\hat{x} + \sin\theta_{i,j}\hat{y})\end{aligned} , \qquad (6)$$

where $w_{i,j} = \begin{cases} 1 & i \neq j \\ 0 & i = j \end{cases}$.

The steering command is then found from:

$$\begin{aligned} \theta_i &= \tan^{-1}\left(\frac{\nabla\phi_i \cdot \hat{y}}{\nabla\phi_i \cdot \hat{x}}\right) \\ &= \tan^{-1}\left(\frac{\sum_{j=1}^{N} w_{i,j}\sin\theta_{i,j}}{\sum_{j=1}^{N} w_{i,j}\cos\theta_{i,j}}\right). \qquad (7) \\ \psi_i &= \beta_1 + \theta_1 \end{aligned}$$

HARDWARE DEVELOPMENT AND SYSTEMS INTEGRATION

Aerial Vehicle and Autopilot

The UAS platforms that were employed for this effort were Unicorns with 153-cm wingspans from Procerus Technologies (see Fig. 4). The small size, low power, and light weight of the sensor and control payload developed for these platforms demonstrate that this technique could be implemented on a fielded military or commercial UAS of similar, or even smaller, size. The UAS autopilot is a Kestrel Autopilot v.2.22, also from Procerus Technologies. These autopilots contain three-axis angular rate and acceleration sensors, a three-axis magnetometer, a barometric altimeter, 20-point sensor temperature compensation, GPS, wind estimation, and a dead-reckoning filter for GPS-denied operation. For this effort, the autopilot was controlled over a serial interface by the onboard autonomy module. The ground station was used only to collect position data for posttest analysis, as well as to provide flight safety.

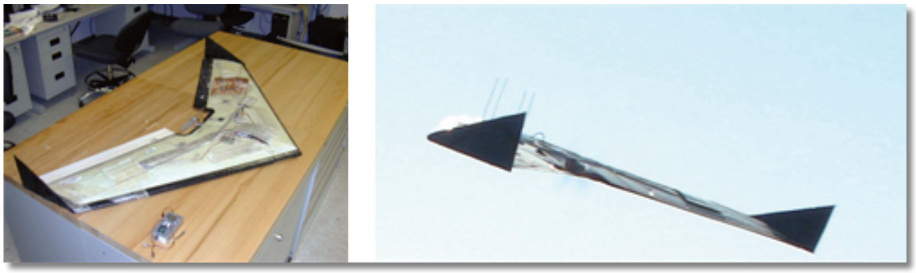

Figure 4. Procerus Unicorn research airplane.

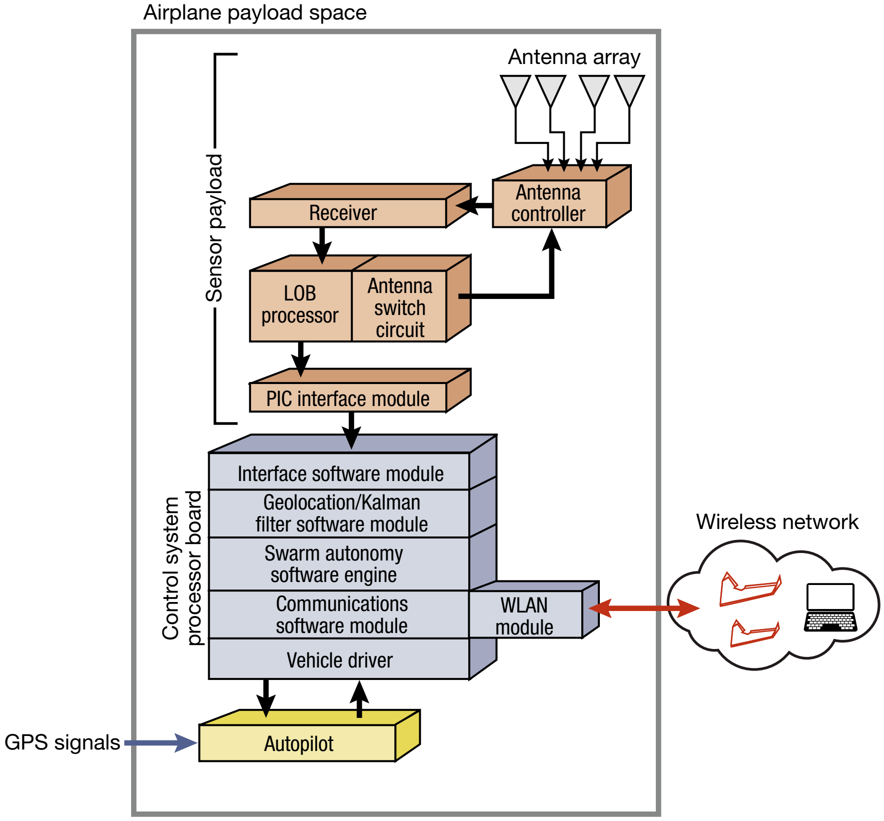

Figure 5. Onboard hardware, software, and communications architecture.

Sensor Payload and Control System

The sensor payload consists of the antenna system, the RF receiver, the antenna switch circuit integrated with the LOB processor module, and a PIC-based interface board. The control system consists of a COTS XScale Reduced Instruction Set Computer (RISC) processor board with integrated WLAN plug-in card. The onboard architecture is shown in Fig. 5.

Sensor Payload

The sensor payload consists of an array of four custom-built antennas, a COTS radio receiver, and custom processing and switching electronics.[10] The radio is a multiband, 900-channel handheld Yaesu VX-6 transceiver that was operated in receive mode only.

In the UHF band that was used to test this geolocation concept, the radio has a sensitivity of 0.2–0.5 μV for 12 dB signal-to-noise and distortion ratio.

The radio receiver output is characterized using a custom circuit that compares the phase of the incoming signal from the four antennas and, from that phase comparison, generates an LOB value. This measurement provides the LOB in discrete steps of 0.3927 radians (22.5°). A separate PIC processor-based board converts this output to the proper format and sends the data over a serial interface to the main control system processor board. The LOB processor board also contains the antenna switch circuit that sends the antenna switching signals to a very small antenna controller board that is collocated with the antennas.

The antenna system is an array of four wire antennas, each with a small ground plane, arranged in a square pattern. The greatest challenges in the payload systems integration effort were the spacing, tuning, ground plane fabrication, impedance matching, and orientation of the antennas. Most of these challenges are a result of the small size of the platform. The size constraints on the antenna system ground plane resulted in the most significant effects to system performance.

Although a COTS transceiver was used for this proof-of-concept effort, future enhancements may include the development of a much smaller receive-only module with similar receive performance to the Yaesu radio. Four of these modules could be used to receive the RF emissions from each antenna simultaneously. Other possible future enhancements include the use of digital signal processing technology in the LOB processor and including the PIC processor data interface on the same board. These enhancements will not only reduce the size, weight, and power budget of the sensor payload but will also significantly improve performance.

Control and Communications Payload

All autonomy control software, Kalman filtering, geolocation algorithms, and communications control are implemented on a Wave Relay Quad Radio 2.4-GHz router.[11] The Wave Relay board was developed by Persistent Systems, LLC. The Wave Relay system is designed specifically for both high scalability and high mobility. The Wave Relay system functions at the data link layer, allowing seamless integration with Ethernet-based networks and devices. The onboard protocols use distributed online learning algorithms to continuously track the highest throughput paths in the network in order to maximize capacity and minimize interference.

Because the Wave Relay routing algorithms take up only a fraction of the processing power and onboard memory, APL used the remaining processing and memory space for its autonomy and geolocation software. The Wave Relay board communicated with the sensor payload and autopilot through the board's serial ports. Communications with other UAS nodes and with the ground node were accomplished using the Wave Relay wireless network capability.

PROOF-OF-CONCEPT SIMULATIONS

Models of two core concept pieces were developed, and simulations were performed to establish proof of concept. One model represented the autonomy and was used both to test and to help develop the autonomy software. The other model represented the geolocation sensor payload, the Kalman filter, and the geolocation algorithms.

The autonomy model demonstrated the theoretical trajectories flown by each vehicle given a stable geolocation solution process. Figure 6 shows convergence to the target for different numbers of UASs and UAS starting position. Once in proximity to the target, they set up circular orbits around the target at fixed phase differences depending on the team size. For instance, for two airplanes, the phase difference is 90°, and for three airplanes it is 120°. (An example of a simulated two-vehicle team orbit is shown in Fig. 11b.)

The geolocation and Kalman filter algorithms were simulated using the second model. A single UAS was used, with the payload sensor resolution and errors included in the model. Figure 7 shows four snapshots of the simulation. At t_1, data collection has only recently begun, and there is little angular diversity in the data, so the error ellipse is quite large. Nevertheless, the center of the ellipse is in close proximity to the actual target. At

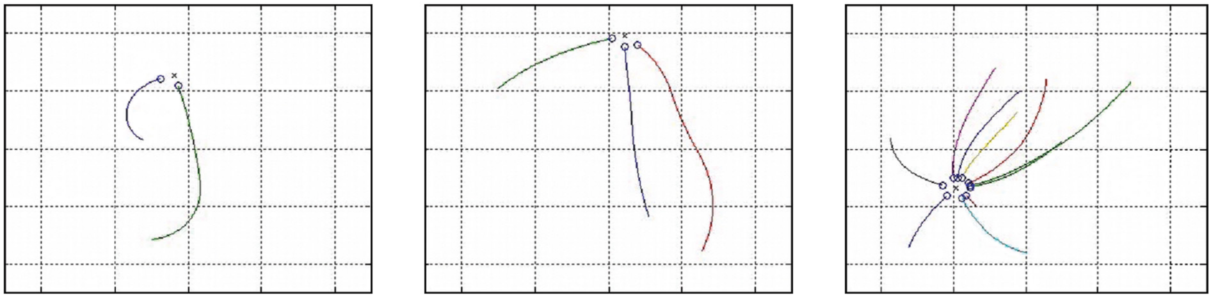

Figure 6. Two, three, and 10 UASs converge to target in simulation.

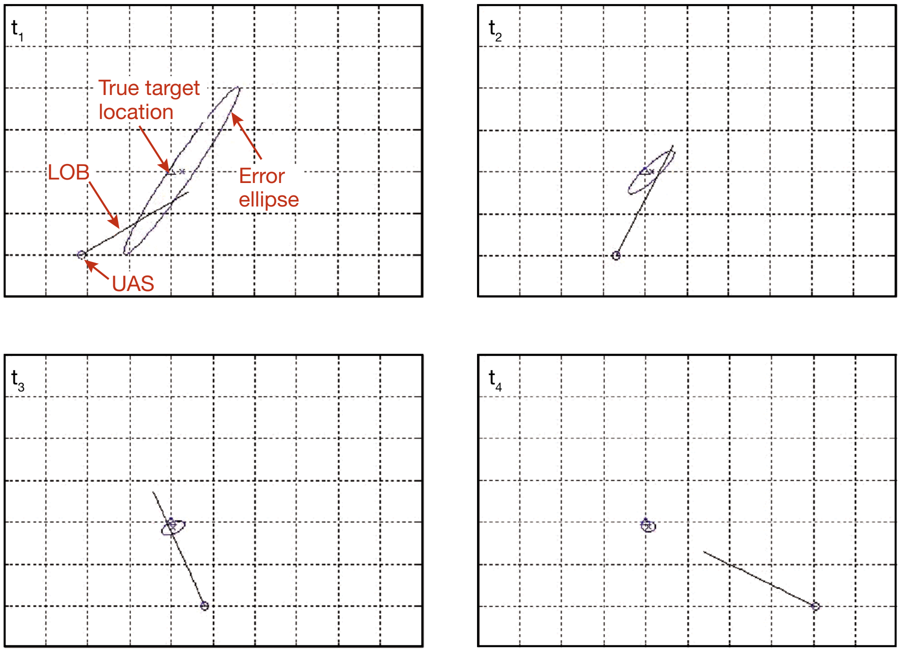

Figure 7. Snapshot of geolocation solution from a single UAS.

t_2 and t_3, more data and more angular diversity result in a shrinking of the error ellipse. Finally, at t_4, the ellipse has collapsed to a very small error centered very near to the true target location.

A series of geolocation solution simulations using multiple vehicles were also conducted. These showed that the accuracy of the solution increased, and the time required for a good solution decreased, with the number of swarm members. However, greater than four vehicles produced diminishing returns, and two vehicles were shown to be adequate for a reasonably fast and accurate solution.

FLIGHT DEMONSTRATIONS

To test this geolocation concept, a series of bench tests, hardware-in-the-loop tests, and field tests were conducted at APL facilities, a nearby leased field, and the U.S. Army's Aberdeen Proving Ground. A final concept demonstration was conducted at Camp Roberts, California, during a Tactical Network Topology exercise; Tactical Network Topology is a quarterly series of experiments hosted by the Naval Postgraduate School and U.S. Special Operations Command.

Two airplanes were used during these tests. As shown in simulation, two airplanes provide a good geolocation solution in a reasonable time period. Each vehicle was outfitted with the full sensor and control payload described previously. Also as previously described, derivation of a geolocation solution and determination of flight behaviors were achieved separately on each airplane on the basis of self-knowledge and communications from the other UAS. That is, all geolocation and autonomy algorithms were performed independently of ground processing.

The experiment that was demonstrated at Camp Roberts consisted of two parts: (*i*) geolocation of an actual RF emitter with two airplanes flying preset loiter patterns and (*ii*) demonstration of the converge and orbit behaviors using a virtual RF emitter (location hard-coded in the onboard autonomy code). Ultimately, it is the combination of these parts that results in realization of the full geolocation concept—that is, geolocation of the actual RF emitter, and vehicle convergence to and orbit of, that emitter. However, this requires real-time calibration of the LOB with the vehicle bearing, and the constrained demonstration schedule did not allow for this calibration step. A posttest calibration was performed for one of the UASs, which resulted in the red line shown in Fig. 8. This calibration was derived by plotting true LOB versus measured LOB collected over two and a half loiter orbits; the loiter point was approximately

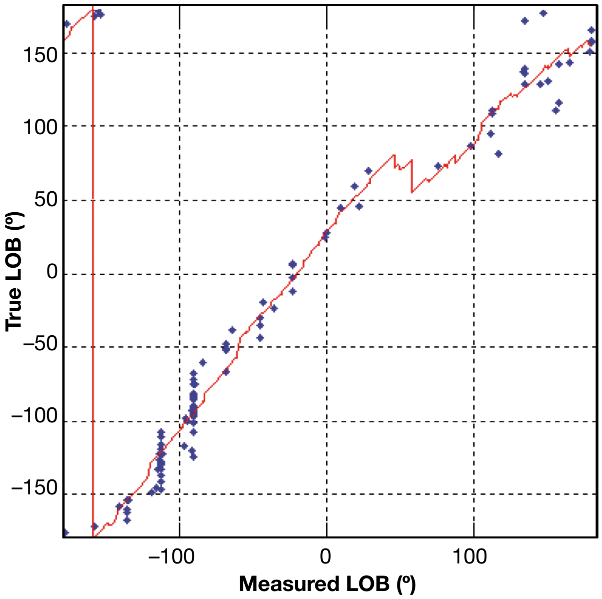

Figure 8. Calibration line (red) derived by plotting true LOB versus measured LOB.

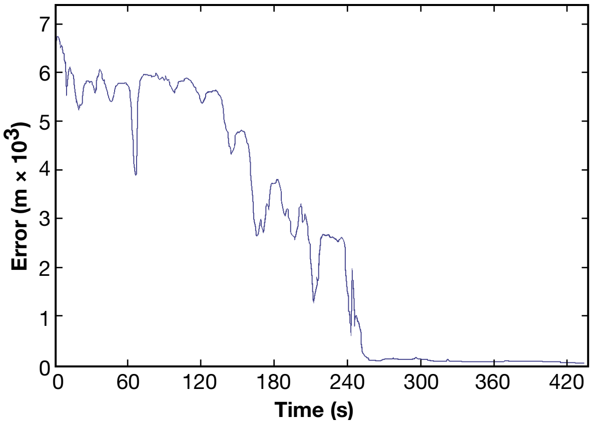

Figure 9. Reduction in solution error as a function of time.

550 m from the calibration source, and the orbit radius was approximately 50 m. Future enhancements to the system are planned to quickly and efficiently provide this calibration in real time.

For the first part of the experiment, an RF emitter transmitting a 350-MHz continuous wave signal was located near the vehicle launch area. Both airplanes were launched with waypoints preset for loiter points approximately 550 m from the emitter.

Error with respect to time was evaluated using the data from one of the UASs. As shown in the plot in Fig. 9, the error drops significantly as a function of time, with the error settling down to a reasonable value approximately 260 s after the start of data collection.

Figure 10 illustrates the reduction in error with respect to the error ellipse. The plots in Fig. 10 show the error ellipse from one airplane at t_0, t_1, t_2, and t_3. The snapshots are roughly equally spaced throughout the data collection time period, so t_0 is the initial reading at 0 s, t_1 is at t_0 + 145 s, t_2 is at t_0 + 290 s, and t_3 is the final error ellipse at t_0 + 435 s.

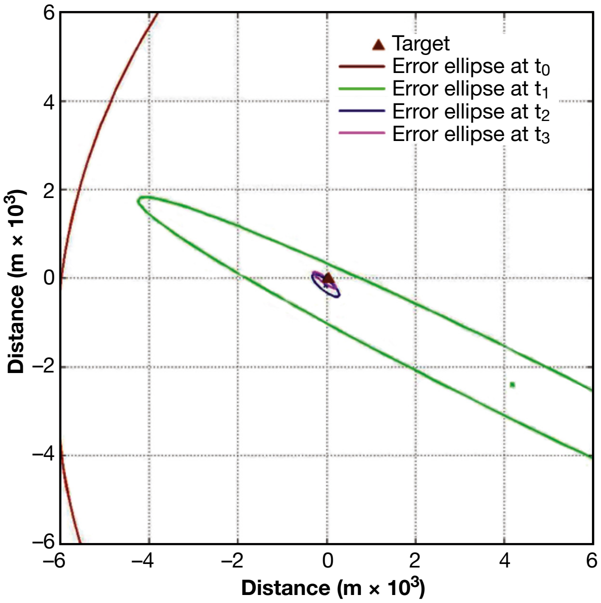

Figure 10. Convergence to a geolocation estimate, and collapse of the error ellipse, over time.

The plot in Fig. 10 also shows convergence to a geolocation solution over time. At the end of the 435-s data collection period, the error between estimated target location and true target location was 60 m.

Because of the calibration issue, the data in Fig. 10 and the 60-m error vector were derived from data fused from only a single aircraft loitering 550 m from the target. The flight path provided little angular diversity (a maximum of 29° with respect to the target), and data collection was over a relatively short time period (435 s). On the basis of simulation, when this data set is extended to three airplanes circling the target at the 550 m distance but 120° out of phase with respect to each other, the error collapses to 12 m and the error settling time illustrated in Fig. 9 is reduced to 30 s. Enhancements to the sensor payload and real-time calibration are expected to result in even smaller errors. Also, bringing the three airplanes closer to the target further reduces error. For instance, if the three airplanes circle the target at a distance of 100 m, the geolocation error is theoretically reduced to 2.2 m.

The second part of this experiment demonstrated the vehicle behaviors that were shown in simulation. A "virtual" RF emitter location was sent over the wireless network from a ground node to both UASs. That is, both UASs were provided the geolocation solution. This experiment, therefore, was intended to demonstrate the autonomous behaviors.

Just as in simulation, both UASs converged on the target and set up orbit around the target 90° out of phase from each other. These behaviors are shown in Figs. 11a and 11b, with the simulation trajectories shown on the left-hand sides next to the corresponding flight trajectories shown on the right. As can be seen, the actual flight paths are virtually identical to the simulation trajectories. This experiment presents strong evidence that the onboard autonomy and belief management system directed the UASs to behave as expected.

SUMMARY

These simulations and flight experiments demonstrate the feasibility of using RF sensors onboard multiple individual aircraft operating cooperatively as a quick, inexpensive, and reliable method of geolocating radio signal emitters. The technology behind the RF sensors was adapted from a technique known as pseudo-Doppler: the target signal was received on a constellation of four rapidly switched onboard antennas, and the phase difference was measured onboard to derive LOB. The LOB values from multiple airplanes were combined to estimate the geolocation solution.

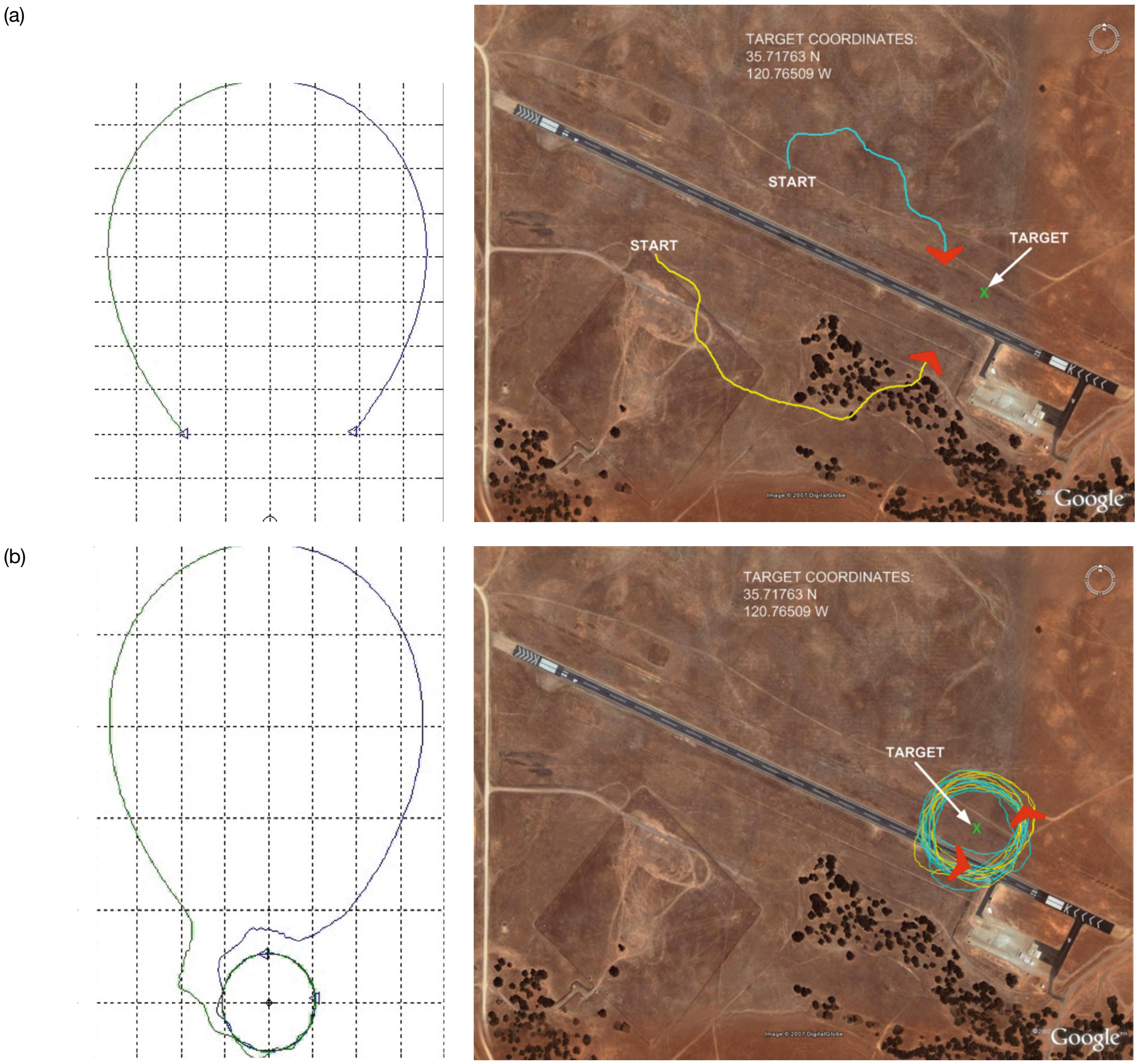

Figure 11. (a) Simulation on the left, and the corresponding flight on the right, show convergence to the target once a geolocation solution has been derived; the simulation and flight paths match closely. (b) Simulation on the left, and the corresponding flight on the right, show orbit of the target with the UASs 90° out of phase; again, the simulation and flight paths match closely.

The onboard autonomy algorithms guided the UASs and optimized trajectories for the best solution. All sensor operation, signal processing, geolocation algorithms, and autonomy were performed onboard the aircraft—that is, the airplanes were operated with no human in the loop except for preflight loading of mission objectives and ground observation of the real-time solution estimation. Although some development tasks remain, this effort provides a proof of concept for geolocation of RF emitters from small, autonomous aerial platforms.

REFERENCES

[1] Scheidt, David, and Stipes, Jason, "Cooperating Unmanned Vehicles," in *Proc. 2005 IEEE International Conf. on Networking, Sensing, and Control*, Tucson, AZ, pp. 326–331 (2005).

[2] Moore, Kevin L., White, Michael J., Bamberger, Robert J., and Watson, David P., "Cooperative UAVs for Remote Data Collection and Relay," in *Proc. AUVSI Unmanned Systems North America 2005 Symp.*, Baltimore, MD (2005).

[3] Denisowski, Paul, "A Comparison of Radio Direction-Finding Technologies," Rhode & Schwarz application note.

[4] Urruela, A, and J. Riba, "A Novel Estimator and Performance Bound for Time Propagation and Doppler Based Radio-Location," in *Proc. 2003 IEEE International Conf. on Acoustics, Speech, and Signal Processing (ICASSP '03)*, pp. V-253–V-256 (2003).

[5] Witzgall, Hanna, "A Reliable Doppler-Based Solution for Single Sensor Geolocation," in *Proc. IEEE Aerospace Conf.*, Big Sky, MT, pp. 1–7 (2013).

[6] Aloi, Daniel N., and Sharawi, Mohammad S., "Modeling and Validation of a 915 MHz Single Channel Pseudo Doppler Direction Finding System for Vehicle Applications," in *Proc. 2009 IEEE 70th Vehicular Technology Conf. (VTC 2009 Fall)*, Anchorage, AL, pp. 1–5 (2009).

[7] Scheidt, David H., "Organic Persistent Intelligence, Surveillance, and Reconnaissance," *Johns Hopkins APL Tech. Dig.* **31**(2), 167–174 (2012).

[8] Mamei, M., Zambonelli, F., and Leonardi, L., "Co-Fields: A Unifying Approach to Swarm Intelligence," in *Proc. 3rd International Workshop on Engineering Societies in the Agents' World*, Madrid, Spain, pp. 68–81 (2003).

[9] Koren, Y., and Borenstein, J., "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation," in *Proc. IEEE Conf. on Robotics and Automation,* Sacramento, CA, pp. 1398–1404 (1991).

[10] Whitlock, Roderick, "High Gain Pseudo-Doppler Antenna," in *Proc. IEEE Loughborough Antennas and Propagation Conf. (LAPC),* Leicestershire, UK, pp. 169–172 (2010).

[11] Bamberger, R., Scheidt, D., Hawthorne, C., Farrag, O., and White, M., "Wireless Network Communications Architecture for Swarms of Small UAVs," in *Proc. AIAA Unmanned Unlimited Conf.*, Chicago, IL, paper AIAA-2004-6594 (2004).

The Authors

Robert J. Bamberger is a member of the APL Principal Professional Staff and Section Supervisor in the Robotics and Autonomous Systems Group of the Research and Exploratory Development Department (REDD). He acted as Principal Investigator, systems integrator, and test lead for this independent research and development (IR&D) effort. **Jay G. Moore** is an APL Senior Staff member and Assistant Section Supervisor in the Advanced RF Capabilities Group in the Asymmetric Operations Sector. Jay developed all of the geolocation algorithms and Kalman filters used for the geolocation solution. Jay also developed the simulation tools used for creation of the geolocation software. **Ravi P. Goonasekeram** is a member of the APL Principal Professional Staff in the RF & Optical Systems Section of REDD's Electrical Engineering Systems Analysis and Design Group. Ravi acted as RF subject matter expert (SME) and lead hardware engineer. He also designed and fabricated the miniature high-performance ground beacon, as well as the onboard circuitry that derived LOB on the basis of measured received signal phase shift. **David H. Scheidt** is a member of the APL Principal Professional Staff and is Program Manager in the Science Applications Group in the Space Sector. David first developed the autonomy concepts that were used to control the UAS path planning and behaviors. He acted as autonomy SME for this IR&D effort. For further information about the work reported here, contact Robert Bamberger. His e-mail address is robert.bamberger@jhuapl.edu.

The *Johns Hopkins APL Technical Digest* can be accessed electronically at **www.jhuapl.edu/techdigest**.