A *Q*-learning approach to automated unmanned air vehicle demining

Silvia Ferrari and Greyson Daugherty

The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology 2012 9: 83 originally published online 30 September 2011
DOI: 10.1177/1548512911414599

Original article

JDMS

Journal of Defense Modeling and Simulation: Applications, Methodology, Technology
9(1) 83–92

DOI: 10.1177/1548512911414599
dms.sagepub.com
SAGE

A Q-learning approach to automated unmanned air vehicle demining

Silvia Ferrari and Greyson Daugherty

Abstract

This paper develops a novel *Q*-learning approach to unmanned aerial vehicle (UAV) navigation, or path planning, for sensing applications in which an infrared (IR) sensor or camera is installed onboard the UAV for the purpose of detecting and classifying multiple, stationary ground targets. The main advantage of this approach over existing path planning techniques is that the optimal guidance policy is learned via the *Q*-function, without explicit knowledge of the system models and environmental conditions. As a result, the onboard guidance algorithm can adapt to different sensors, vehicle dynamics, and environmental conditions, without designer intervention, and without the need for accurate modeling of every system component. The approach is demonstrated through a demining application in which a UAV-based IR sensor is capable of determining the optimal altitude for properly detecting and classifying targets buried in a complex region of interest.

Keywords

aircraft, classification, demining, navigation, neural network, *Q*-learning, sensor, vehicle

1 Introduction

In this paper we develop a *Q*-learning sensor planning approach to unmanned aerial vehicle (UAV) navigation for sensing and surveillance applications. The problem can be considered as a geometric sensor path planning problem, because the geometry and position of the sensor's field of view (FOV) determine what targets can be detected and classified at any given time. The approach developed in this paper is applicable to airborne sensors, such as UAVs or helicopters with onboard sensors, that are deployed over a region of interest (ROI), for the purpose of detecting and classifying hidden (e.g. buried) targets, in variable and uncertain environments. The approach is demonstrated through an application involving an infrared (IR) sensor installed onboard an UAV, referred to as the UAV-IR sensor. The UAV-IR sensor flies over a two-dimensional ROI for the purpose of detecting and classifying buried targets, such as clutter, mines, or unexploded ordnance. The goal of the *Q*-learning algorithm is to determine the UAV guidance law or *policy* that maximizes the number of targets that are properly classified by the onboard IR sensor, without explicit knowledge of the UAV and sensor models, or of the environmental conditions. Environmental conditions, such as time of day, weather, and vegetation, have an influence on the IR sensor performance and, therefore, the optimal policy. However, they may be partially unknown or changing over time and, therefore, their influence on the sensor measurements is learned implicitly via the *Q*-function, based on the immediate sensing reward.

Sensor planning is concerned with determining a policy for gathering sensor measurements to support a sensing objective, such as target classification. When the sensors are installed on mobile platforms, an important part of the problem is determining the optimal sensor path or guidance policy.[1] Several approaches have been proposed for planning the path of mobile robots with onboard sensors to enable navigation and obstacle avoidance in unstructured dynamic environments.[2] However, these methods are not

Laboratory for Intelligent Systems and Control (LISC), Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC, USA

Corresponding author:
Silvia Ferrari, Laboratory for Intelligent Systems and Control (LISC), Department of Mechanical Engineering and Materials Science, Duke University, 176 Hudson Hall, Durham, NC 27708, USA.
Email: sferrari@duke.edu

directly applicable to robotic sensors whose primary goal is to support a sensing objective, rather than to navigate a dynamic environment.[3] The reason is that they focus on how the sensor measurements can best support the robot motion, rather than focusing on the robot motions that best support the sensing objective.[3] In this paper we address the problem of planning the path and measurements of a robotic sensor, in order to support the sensing objective of properly classifying multiple targets distributed in an obstacle-populated workspace. This problem can be viewed as a *geometric* sensor-path planning problem because the sensor's FOV determines what targets can be detected and classified at any given time.

Geometric sensing problems require a description of the geometry and position of the targets, and of the sensor's FOV.[4] Viewpoint planning has been shown by several authors to be an effective approach for optimally placing or moving vision sensors based on the target geometry and sensor's FOV, using weighted functions or tessellated space approaches.[5] In a geometric sensor path planning problem, the optimal sensor path and measurement sequence depend on the UAV dynamics, the IR sensor's characteristics and FOV, and on the characteristics of the ROI.[6] Probabilistic deployment is an effective approach for detecting targets in an ROI by computing a search path based on the probability of finding a target in every unit bin of a discretized obstacle-free ROI.[7] These existing approaches to geometric sensor path planning, however, require prior information, such as sensor and platform models, environmental conditions, and prior sensor measurements. The approach presented in this paper allows the sensor to learn an implicit model of its own dynamics and measurement process, and of the ROI, based on the immediate sensing reward calculated by flying over a training ROI in which only the actual classification of the targets is known *a priori*. Through Q-learning, the UAV-IR sensor learns an optimal guidance policy that can be applied to new ROIs, without the need for designer intervention. Since the size of the Q-learning training set grows exponentially with the dimensions of the state and action spaces, incremental learning and prior knowledge may be used to alleviate the computational burden and memory requirements.

The paper is organized as follows. The Q-learning approach is reviewed in Section 2. The mathematical models used to simulate the system and learn the optimal policy are described in Section 3. The Q-learning approach to UAV-IR sensor-path planning is presented in Section 4. Then, the methodology is demonstrated in Section 5 by means of a novel simulation comprised of an integrated demining system where the IR sensor is installed onboard a UAV that obeys a six-degree-of-freedom equation of motion derived from full-scale wind tunnel data and physical models reviewed by Stengel.[8]

2 Background on Q-learning

Approximate dynamic programming (ADP) methods, such as Q-Learning, are valuable tools for solving optimal control problems online, subject to partial or imperfect knowledge of the system state and models.[9] Optimal control problems involve a dynamic system (or process) that is either stochastic or deterministic. Although various notations are in use in the ADP literature,[10] in this paper we adopt the notation that is typically used in the optimal control and dynamic programming community (see Bertsekas[11] for a detailed description and introduction). Assuming time can be discretized and indexed by k, a deterministic dynamical system may be modeled by the difference equation,

$$x_{k+1} = f(x_k, u_k, k) \tag{1}$$

where the state x_k at time k is an element of the *state space* $\mathcal{X}$, and the control u_k at time k is an element of the space $\mathcal{A}$ of admissible actions or decisions. If the dynamical system is stochastic, then it may be modeled as a Markov decision process (MDP).[12] An MDP is a tuple $\mathcal{M} = \{\mathcal{X}, \mathcal{A}, \mathcal{T}, \mathcal{R}\}$ representing a random and sequential decision process. In this case, the state space is a finite set of possible state values, denoted by $\mathcal{X} = \{s_1, \ldots, s_n\}$, and the space $\mathcal{A} = \{a_1, \ldots, a_m\}$ is a finite set of admissible actions or decisions. Here T is the transition probability function, $T : \mathcal{X} \times \mathcal{A} \to P(\mathcal{X})$, which describes the MDP state transitions, such that whenever the state at time k has value $x_k = s_i$ and the decision is $u_k = a_j$, there is a probability $P(x_{k+1} = s_l \,|\, x_k = s_i, u_k = a_j)$ that the next state value is $x_{k+1} = s_l$. In many real-world applications of optimal control, however, the exact form of the difference equation (1) or the transition matrix T are unknown or approximate. ADP methods aim at learning the optimal policy over time using online state observations, and immediate state predictions and rewards, without the need for an explicit model of the system dynamics.

In optimal control problems, there exists a reward associated with the dynamic system that may be represented by a reward function $R : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ that specifies the value of the immediate reward $r_k = R(x_k, u_k)$ received after executing the action decision u_k in state x_k. A policy is a mapping of state values to actions, $\pi : \mathcal{X} \to \mathcal{A}$. Let the value function $V^\pi(x_k)$ denote the expected discounted return of a policy π, defined as

$$V^\pi(x_k) = E\left\{\sum_{i=0}^{\infty} \gamma^i r_{k+i} | \pi, x_k\right\} \tag{2}$$

where $E\{\cdot\}$ denotes the expectation, r_{k+i} is the reward received i steps into future, and the discount factor $0 \le \gamma < 1$ modulates the effect of future rewards on present decisions, with small values emphasizing near-term

gain and larger values emphasizing later rewards. Then, an *optimal policy* π^* is one that maximizes $V^{\pi}(x_k)$ for all possible states $x_k \in \mathcal{X}$. The Markov property guarantees that an optimal policy exists, although it may not be unique, and, thus, it is associated with an *optimal value function* $V^*(x_k) = \max_{\pi} V^{\pi}(x_k)$. The optimal policy of an MDP, $\mathcal{M}$, is a fixed point of the Bellman's equation, which can be determined iteratively using policy iteration or value iteration algorithms.[13]

In value iteration, the value of a state $V(x_k)$ is the total expected discounted reward accrued by a policy starting at $x_k \in \mathcal{X}$. The Q function of a state–action pair, $Q(x_k, u_k)$, is the total expected discounted reward accrued by a policy that produces $u_k = \pi(x_k)$.[13] The Bellman equation can be formulated in terms of the aforementioned functions, such that the state–action value function can be written as

$$Q(x_k, u_k) = E\{R(x_k, u_k) + \gamma V(x_{k+1})\}, \tag{3}$$

where

$$V(x_{k+1}) = \max_{u_{k+1} \in \mathcal{A}} Q(x_{k+1}, u_{k+1})$$

If two functions $Q(\cdot)$ *and* $V(\cdot)$ satisfy the above Bellman equation, then they specify an optimal *greedy* policy

$$\pi^*(x_k) = \arg \max_{u_k \in \mathcal{A}} Q(x_k, u_k) \tag{4}$$

Value-iteration algorithms use Equation (3) to iteratively determine $Q(\cdot)$ and $V(\cdot)$ and, subsequently, determine $\pi^*(\cdot)$.

Value iteration can be used to determine the optimal policy of an MDP, $\mathcal{M}$, provided that the transition probability function T is known. If T is unavailable, Q-learning can be utilized to learn an approximate state–action value function $Q(x_k, u_k)$ that is iteratively updated by the rule

$$Q(x_k, u_k) \leftarrow (1-\alpha)Q(x_k, u_k) + \alpha[r_k + \gamma \max_{u_k \in \mathcal{A}} Q(x_{k+1}, u_k)] \tag{5}$$

where α is the learning rate, and $0 < \alpha \leq 1$. In this paper, Q-learning is used to solve a new sensor planning problem described in the next section, which consists of obtaining the optimal guidance policy for an IR sensor deployed onboard an UAV for mine detection and classification.

3 Mathematical models and problem formulation

3.1 UAV-IR dynamic equation

The problem considered in this paper consists of learning an optimal guidance policy for an UAV with an onboard IR sensor that flies over a minefield $\mathcal{W} \subset \mathbb{R}^2$, or ROI, for the purpose of detecting and classifying buried landmines and unexploded ordnance (UXO). The UAV dynamics can be modeled by a six-degree-of-freedom equation of motion derived from Newton's second law using inertial- and body-reference frames.[8] The full aircraft state consists of the 12-dimensional vector $x_a = [u \quad v \quad w \quad x_r \quad y_r \quad z_r \quad p \quad q \quad r \quad \phi \quad \theta \quad \psi]^{\mathrm{T}}$, where u, v, w, and p, q, r are the UAV velocities and angular rates in body frame, respectively, and x_r, y_r, z_r, and ϕ, θ, ψ, are the UAV translational and angular positions in inertial frame, respectively. The body state accelerations, denoted by X_b, Y_b, Z_b, L_b, M_b, and N_b are a function of the available thrust, and of the aerodynamic force and moment coefficients produced by the controls for the present aircraft state and wind field. The model estimates low-angle-of-attack Mach effects, power effects, and moments and products of inertia by using available full-scale wind tunnel data and physical characteristics, according to the methods described by Stengel.[8] The moments of inertia I_{xx}, I_{yy}, I_{zz}, and product of inertia I_{xz} are estimated using simplified mass distributions, and are held fixed at all times. Then, using the classical aircraft angles definitions and coordinate transformations described by Stengel,[8] the following UAV equation of motion can be obtained

$$\begin{aligned}
\dot{u} &= X_b + g_x + rv - qw \\
\dot{v} &= Y_b + g_{b_y} + pw - ru \\
\dot{w} &= Z_b + g_{b_z} + qu - pv \\
\dot{x}_r &= u\cos\theta\cos\psi + v(\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi) \\
&\quad + w(\cos\phi\sin\theta\cos\psi - \sin\phi\sin\psi) \\
\dot{y}_r &= u\cos\theta\sin\psi + v(\sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi) \\
&\quad + w(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) \\
\dot{z}_r &= -u\sin\theta + v\sin\phi\cos\theta + w\cos\phi\cos\theta \\
\dot{p} &= \frac{q}{(I_{xx}I_{zz} - I_{xz}^2)}\{I_{zz}L_b + I_{xz}N_b - p[I_{xz}(I_{yy} - I_{xx} \\
&\quad - I_{zz})] + r[I_{xz}^2 + I_{zz}(I_{zz} - I_{yy})]\} \\
\dot{q} &= \frac{(M_b - pr(I_{xx} - I_{zz}) - I_{xz}(p^2 - r^2))}{I_{yy}} \\
\dot{r} &= \frac{q}{I_{xx}I_{zz} - I_{xz}^2}\{I_{xz}L_b + I_{zz}N_b + r[I_{xz}(I_{yy} - I_{xx} \\
&\quad - I_{zz})] + p[I_{xz}^2 + I_{xx}(I_{xx} - I_{yy})]\} \\
\dot{\phi} &= p + (q\sin\phi + r\cos\phi)\tan\theta \\
\dot{\theta} &= q\cos\phi - r\sin\phi \\
\dot{\psi} &= \frac{q\sin\phi + r\cos\phi}{\cos\theta}.
\end{aligned} \tag{6}$$

The aircraft control inputs consist of the throttle δT, the elevator δE, the aileron δA, and rudder δR, i.e. $u_a = [\delta T \quad \delta E \quad \delta A \quad \delta R]^{\mathrm{T}}$. As shown by Ferrari and Stengel,[14] the UAV can be fully controlled by means of a reduced state vector $\mathbf{x}_{\mathrm{UAV}} = [V \quad \gamma \quad q \quad \theta \quad r \quad \beta \quad p \quad \mu]^{\mathrm{T}}$, which

is formulated in terms of the aircraft speed V, sideslip angle β, and path angle γ, where

$$V = \sqrt{u^2 + v^2 + w^2} \tag{7}$$

$$\beta = \sin^{-1}(v/V) \tag{8}$$

$$\gamma = \sin^{-1}(-w/V) \tag{9}$$

and in terms of the bank angle μ, defined by Stengel.[8]

3.2 Model of IR sensor measurements

The FOV of the onboard IR sensor is assumed to be a closed and bounded subset of an Euclidian space, $\mathcal{S} = [0, L_{IR}]^2 \subset \mathbb{R}^2$, with the square geometry illustrated by the gray area in Figure 1. It can be easily shown using planar geometry that the size of the FOV is a function of the aircraft altitude $H = -z_r$,

$$L_{IR} = H \sin \theta_{IR} \tag{10}$$

where z_r is defined positive downward by convention,[8] and θ_{IR} is the sensor's aperture angle. In this paper, it is assumed that θ_{IR} is held constant, and that the orientation of the IR sensor is fixed with respect to the UAV body frame. It follows that the position and size of the FOV are a function of time, $\mathcal{S} = \mathcal{S}(t)$, and change based on the aircraft trajectory or path. For simplicity, it is also assumed that the centroid of $\mathcal{S}(t)$ coincides with the UAV coordinates in inertial frame, $x_r(t)$ and $y_r(t)$, at any time t.

As illustrated in Figure 1, the position and geometry of the FOV determine which regions of the minefield can be measured by the airborne IR sensor. The IR sensor measurements are influenced by its height above the ground (H), and by the environmental conditions in the minefield. A two-dimensional grid is superimposed on the minefield dividing it into unit-square cells. Soil characteristics, vegetation, and time-varying meteorological conditions, modeled according to MacDonald[15] and Van Dam et al.,[16] are assigned to each cell, either at random or at user-specified positions. Buried targets are modeled as anti-tank mines (ATMs), anti-personnel mines (APMs), UXO, and clutter objects (CLUTs) that are sampled and reproduced using the Ordata Database,[17] which contains over 5,000 explosive items and 3,000 metallic and plastic objects that resemble anti-personnel mines. Each target i occupies one or more cells in the minefield depending on its size z_i, and is characterized by a depth d_i, and shape s_i (Table 1). The IR sensor mode, v_{IR}, is given by the UAV altitude (H) in kilometers, which is discretized in a set of m possible values $\{a_1, \ldots, a_m\}$. At any given time, the space of admissible values of v_{IR} depends on the UAV speed, and is known from the aircraft flight envelope. The aircraft flight envelope, denoted by $\mathcal{E}$, is the set of altitudes and velocities for which the aircraft can be trimmed (an example is shown in Figure 3). The envelope's boundary is designed by considering the stall speed, the thrust/power required and available, compressibility effects, and the maximum allowable dynamic pressure to prevent structural damage.[8]

IR sensors detect anomalies in IR radiation and, based on their height above the ground, build an image of a horizontal area, obtaining cursory measurements of shape and size for shallow-buried objects. Because they rely on temperature variations, their performance is highly influenced by illumination, weather, vegetation, and soil properties. As shown by Ferrari and Vaghi,[18] an IR sensor can be modeled by the Bayesian network (BN) in Figure 2, based on data and on the IR working principles and detailed studies of Agema Thermovision 900 sensors.[19] All BN nodes represent variables that have an influence on the IR measurement process, and are defined as shown in Table 1.

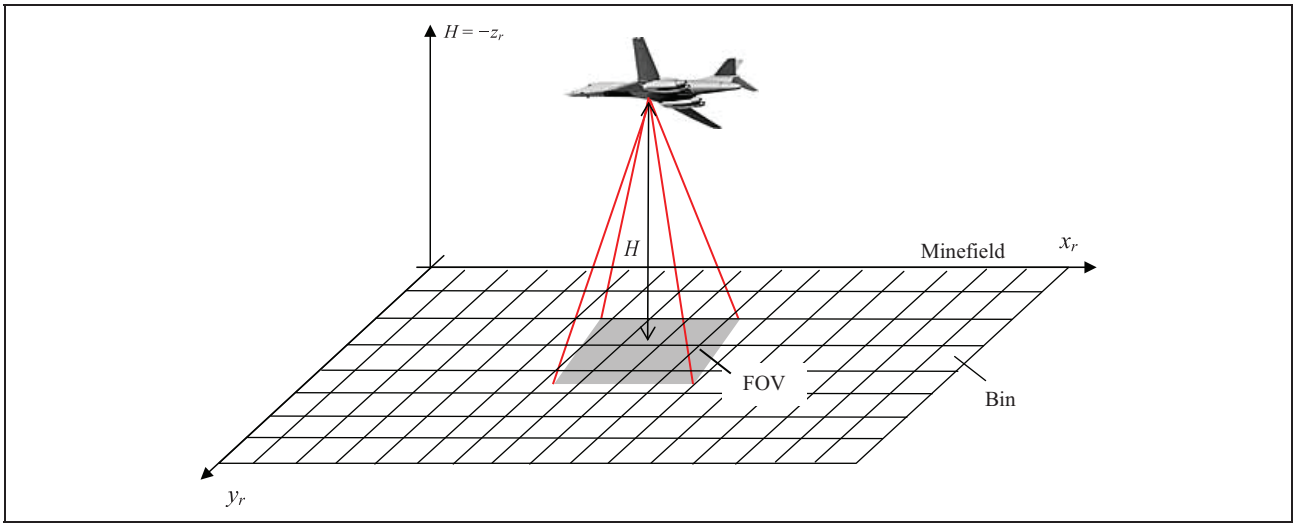

Figure 1. Problem description.

Table 1. Infrared sensor variables and environmental conditions.

Symbol	Nodes	Range
y_i	Target classification	{mine(1), not mine(0)}
v_{IR}	IR mode (km)	$\{a_1, \ldots, a_m \mid a_h = h \cdot 0.2 \text{ km}\}$
E_i	Soil moisture (%): s_r	{dry [0, 10], wet (10, 40], saturated (>40)}
	Soil composition: s_c	{very-sandy, sandy, high-clay, clay, silt}
	Soil uniformity: s_u	{yes, no}
	Vegetation: g	{no-vegetation, sparse, dense}
	Weather: w	{clear, overcast, raining}
	Illumination: i	{low (07:00 − 10:00 and 18:00 − 21:00), medium (10:00 − 13:00), high (13:00 − 18:00)}
F_i	Depth (cm): d_i	{surface [0], shallow-buried (0, 12], buried (12, 60], deep-buried (>60)}
	Size (cm): z_i	{small (2, 13], medium (13, 24], large (24, 40], extra-large (>40)}
	Shape: s_i	{cylinder, box, sphere, long-slender, irregular}

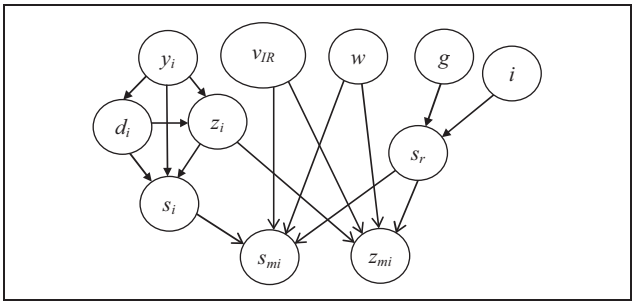

Figure 2. Bayesian network model of the infrared sensor (adapted from Cai and Ferrari[6]).

The IR BN model approximates the joint probability mass function (PMF) underlying the IR sensor measurements in terms of the recursive factorization

$$P(v_{\text{IR}}, E_i, M_i, F_i, y_i) = P(M_i \mid v_{\text{IR}}, E_i, F_i)P(F_i \mid y_i)P(y_i)P(v_{\text{IR}})P(E_i), \quad \forall i \tag{11}$$

where $F_i = \{d_i, z_i, s_i\}$ is the set the features of the ith target, $M_i = \{d_{m_i}, z_{m_i}, s_{m_i}\}$ are the *measured* target features extracted from sensor images, and y_i denotes the ith target classification with the range $\mathcal{Y} = \{\text{mine}, \text{ not mine}\}$. In this paper it is assumed that the environmental conditions E_i are constant and uniform everywhere in $\mathcal{W}$, but are possibly unknown. The approach can also be extended to the case of heterogeneous and time-varying environmental conditions by including E_i in the definition of the state, as will be shown in a separate paper. The factors in (11) are conditional PMFs given by the BN conditional probability tables (CPTs); see Jensen[20] for a comprehensive review of BNs. By this approach, non-Gaussian sensor models can be obtained and used for sensor planning, as shown in Section 4.

As shown in previous research,[6,18] when a sensor is installed on a mobile platform, the measurement gathering process can be modeled as a Markov decision process, under proper assumptions. Although the MDP transition probability matrix of the UAV-IR system could potentially be obtained from the nonlinear dynamic equation (6), the BN sensor model, environmental maps, and weather forecasts, it would be computationally prohibitive to determine it from all possible minefield, weather, UAV, and IR sensor characteristics. Therefore, the goal of this paper is to develop a Q-learning technique that can learn the UAV-IR guidance policy in real time from the sensing reward, without explicit knowledge of the transition probability matrix. By this approach, the same guidance algorithm can be applied to different airborne sensors and minefields, without redesigning the algorithm or modeling every system component.

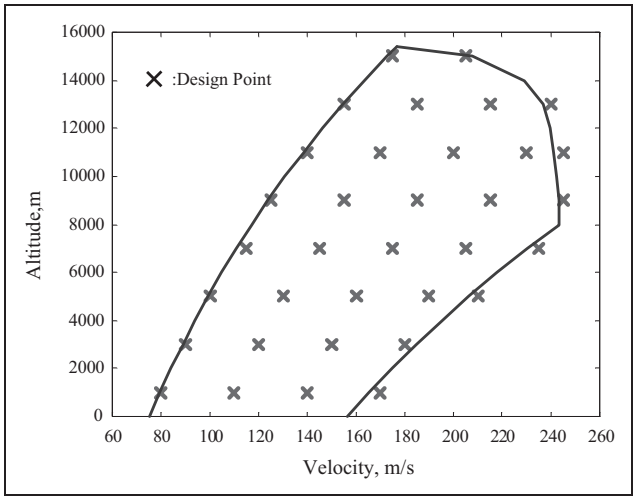

Figure 3. Aircraft flight envelope (adapted from Ferrari and Stengel[14]).

4 *Q*-learning approach to UAV-IR sensor path planning

The problem of determining the optimal sensor path for searching and classifying hidden targets, known as a

treasure hunt problem, was first formulated as an MDP by Ferrari and Cai.[21] An effective Q-learning technique for solving treasure hunt problems was presented by Cai and Ferrari,[22] and demonstrated through the benchmark problem of the game of CLUE®. In this paper, Q-learning is applied to the new UAV-IR demining problem described in Section 3, which can be viewed as a new application example of a treasure hunt. Since the sensor is installed onboard a UAV, the UAV path determines what cells can be intersected by the sensor's FOV, and measured by the IR sensor at any moment in time.

4.1 Definition of Q-learning state and control vectors

As a first step, the aircraft dynamics (6) are evaluated at N equally spaced discrete points in time, $t_k = t_0 + k\Delta t$, $k = 0, \ldots, (N-1)$, over the interval $[t_0, t_f]$, where $\Delta t = (t_f - t_0)/N$ is the discretization interval. Between any two points in time, the control is assumed to be piecewise-constant and the UAV dynamics (6) are integrated by a third-order Runge-Kutta integration routine.[23] In order to apply the Q-learning technique in Section 2, the MDP state x_k must be observable, and may be defined as a subset of the full system state. Thus, based on the problem formulation in Section 3, the state vector defined as $x_k = [x_r(t_k) \quad y_r(t_k) \quad z_r(t_k) \quad V(t_k)]^{\mathrm{T}}$ determines the IR sensor FOV's size and position at any time t_k. The FOV's size and position, in turn, determine the hidden target characteristics in cell i, denoted by the set $\zeta_i = \{d_i, z_i, s_i, y_i\}$, and the hidden environmental conditions E_i, through the subset of cells that are intersected by the FOV, with index set I_k. The environmental conditions may or may not be known depending on the scenario. The set ζ_i of hidden variables can be estimated only after the FOV has intersected cell i.

The objective of the optimal guidance policy, $u_k = \pi^*(x_k)$, is to compute the next UAV position, at t_{k+1}, such that the resulting UAV-IR sensing performance is maximized over time. Therefore, the control vector is defined as the next waypoint, i.e. $u_k = [x_r(t_{k+1}) \; y_r(t_{k+1}) \quad z_r(t_{k+1})]^{\mathrm{T}}$, where $z_r(k+1)$ determines the next IR sensor mode $v_{\mathrm{IR}}(t_{k+1})$. In this approach, the Q-learning technique provides an outer-loop algorithm that outputs a desired trajectory to be followed by means of the inner-loop proportional–integral (PI) controller described by Ferrari and Stengel.[14]

4.2 Definition of Q-learning reward

After the IR measurements are obtained from all of the cells inside the sensor's FOV, the IR BN model (11) is used to estimate (or infer) the target classification based on the *measured* target features (M_i) extracted from sensor images, the sensor mode v_{IR} and, possibly, known environmental conditions. In this paper, BN inference is performed by the junction-tree algorithm available through the Matlab® BN-Toolbox commands *jtree_inf_engine*, *enter_evidence*, and *marginal_nodes*.[24] The inference algorithm provides the posterior PMF $P(y_i, d_i, z_i, s_i | v_{\mathrm{IR}}, d_{i_m}, z_{i_m}, s_{i_m}, E_i)$, and the target classification is estimated by choosing the value of highest posterior probability, i.e.

$$\hat{y}_i = \arg \max_{y_i^k \in \mathcal{Y}_i} P(y_i | v_{\mathrm{IR}}, d_{i_m}, z_{i_m}, s_{i_m}, E_i) \tag{12}$$

The estimated target classification is then accompanied by a so-called confidence level (CL),

$$c_i = P(\hat{y}_i | v_{\mathrm{IR}}, d_{i_m}, z_{i_m}, s_{i_m}, E_i), \tag{13}$$

which represents the confidence in the estimated value $\hat{y}_i$, and for a binary variable obeys $0.5 \le c_i \le 1$.

Let y_i^* denote the actual classification of the target in cell i. Then, the classification error defined as

$$e_i = |\hat{y}_i - y_i^*| \tag{14}$$

is also binary, and takes a value of zero when the estimate is correct and a value of one when the estimate is incorrect. If the estimate is correct, a higher CL is desirable, but, if the estimate is incorrect, a lower CL is desirable because it indicates that the confidence in the estimate is low. Thus, the IR sensor performance is reflected in the classification error (14) and in the CL. In addition, the sensor performance depends on application-specific mission objectives for deploying the UAV-IR. For example, in some applications it may be of interest to minimize the number of false alarms, whereas in others it may be of interest to find cells without targets, in order to determine a safe path through $\mathcal{W}$. In this paper, the mission objectives are characterized by a discrete risk function defined as

$$\rho_i = \begin{cases} w_1 & \text{if } \hat{y}_i = 1, \; y_i^* = 0 \text{ (false alarm)} \\ w_2 & \text{if } \hat{y}_i = 0, \; y_i^* = 1 \text{ (misclassification)} \\ w_3 & \text{if } \hat{y}_i = 1, \; y_i^* = 1 \text{ (mine detection)} \\ w_4 & \text{if } \hat{y}_i = 0, \; y_i^* = 0 \text{ (void-cell detection)} \end{cases} \tag{15}$$

where $w_1, \ldots, w_4$ are user-defined positive constants that weigh the relative importance of the four cases listed in (15). If in a mission a false alarm poses a much greater risk than a misclassification, then $w_1 \gg w_2$. If, in addition, it is of secondary importance to correctly classify mines, then $w_1 \gg w_3 \gg w_2, w_1$, and so on. In this paper, $w_1, \ldots, w_4$ are $O(1)$, but in some applications it may be useful to define weights of higher orders of magnitude (e.g. $O(3)$ or $O(4)$) in order to greatly emphasize one mission objective over the others.

Then, the immediate reward from cell i can then be defined as a tradeoff between the measurement value and error

$$r_i = W_v[(1 - e_i)]\ c_i\rho_i - W_e\,(e_i c_i \rho_i) \tag{16}$$

where W_v and W_e are user-defined positive constants that represent the desired tradeoff between the measurement value of obtaining correct classifications of mines or void cells, and the measurement error of incorrectly classifying mines or false alarms. At every time t_k, the IR sensor obtains measurements from a set of cells in its FOV, $\mathcal{S}(t_k)$ and, thus, the total value of the immediate reward is

$$r_k = R(x_k, u_k) = \sum_{i \in \mathcal{S}(t_k)} W_v[(1 - e_i)]\ \ c_i\rho_i - W_e\,(e_i c_i \rho_i) \tag{17}$$

and can be computed from the IR sensor measurements, and the actual target classification y_i^*.

In this paper, the Q function is approximated by a feedforward sigmoidal neural network (NN),

$$Q(x_k, u_k) = W_2\Phi(W_1[x_k^{\mathrm{T}}\ u_k^{\mathrm{T}}]^{\mathrm{T}} + b_1) + b_2 \tag{18}$$

by means of the update rule (5) as the UAV explores the state and control spaces, $\mathcal{X}$ and $\mathcal{A}$. The s-dimensional operator Φ represents one hidden layer of s sigmoidal functions of the form $\sigma(n) \equiv 1/(1 + e^{-n})$. The NN weights $W_1 \in \mathbb{R}^{s \times (n+m)}$, $W_2 \in \mathbb{R}^{1 \times s}$, $b_1 \in \mathbb{R}^s$, and $b_2 \in \mathbb{R}^2$, are determined by the resilient backpropagation algorithm ('trainrp').[25] A training set for (18) is formed according to the Q-learning approach. As a first step, the Cartesian product of the state and control spaces $\mathcal{X} \times \mathcal{A}$ is discretized. With the state and control definitions in Section 4.1, this is achieved by discretizing the flight envelope (e.g. see the crosses in Figure 3). As a second step, the rule in (5) is applied iteratively over the discrete time t_k, while exploring $\mathcal{W}$ (already discretized into cells) by flying the UAV at every feasible pair of altitudes and velocities.

After the reward (17) is evaluated for every pair of state and control values explored by the UAV-IR, the data can be used to learn the Q function using (18). Then, the optimal policy, $u_k = \pi^*(x_k)$, is determined by maximizing the learned Q function using the greedy rule in (4). The effectiveness of this approach is demonstrated in the next section, using the system models described in Section 3.

5 Numerical simulations and results

The UAV is simulated by integrating the ODE in (6), using the Matlab® program FLIGHT, developed by Stengel.[8] This program simulates the aircraft flight, and determines the trim conditions for any nominal altitude and velocity that lie inside the aircraft flight envelope $\mathcal{E}$, illustrated in Figure 3. In this paper, it is assumed that the UAV maintains steady-level longitudinal flight. The Q-guidance algorithm determines the optimal nominal altitude H^* based on the UAV-IR sensing reward obtained by flying over a minefield $\mathcal{W}$ with unknown geometric and environmental conditions, and known targets.

The UAV-IR sensor, and the minefield $\mathcal{W}$ are simulated numerically, using the mathematical models described in Section 3. In the simulation, as soon as the FOV of the IR sensor, $\mathcal{S}$, intersects a cell containing a target, measurements are reproduced and deteriorated based on the target features, the sensor's mode and working principles, and the environmental conditions in the cell.[18] As an example,

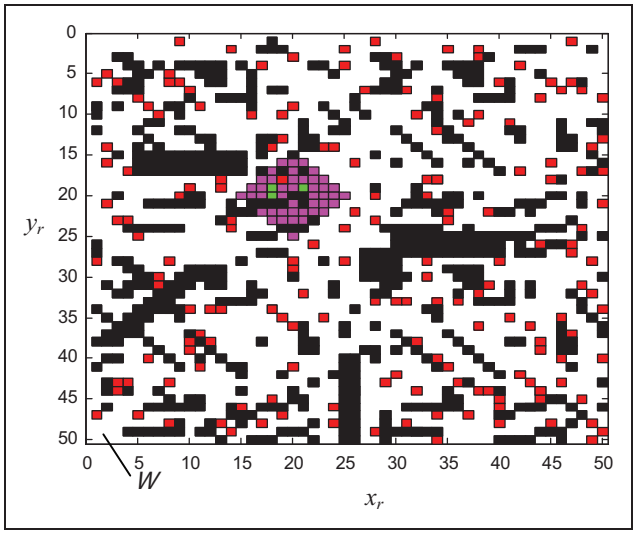

Figure 4. Instantaneous UAV-IR sensor's FOV at $a_8 = 1.6$ km.

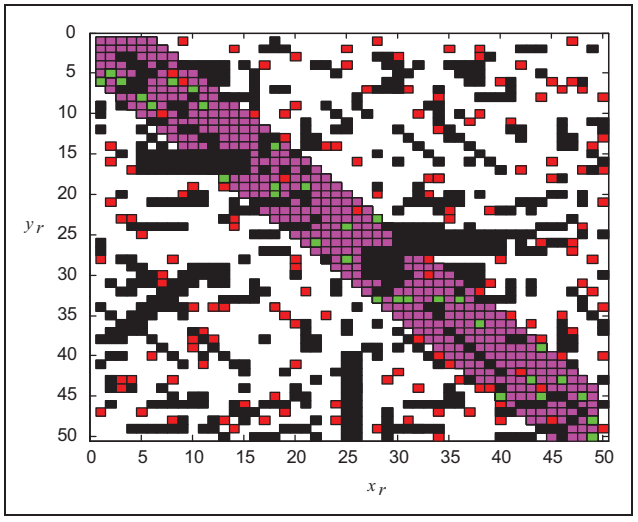

Figure 5. Cells measured by UAV-IR during $[t_0, t_f]$, at $a_8 = 1.6$ km.

the set of cells measured by the UAV-IR sensor at a sample moment in time and an altitude $a_8 = 1.6$ km is shown by purple bins in Figure 4. Red bins represent undetected targets (mines or clutter), green bins represent detected targets, and black bins represent obstacles.

In this paper, the environmental conditions in Table 1 are assumed uniform in $\mathcal{W}$ and, therefore, the UAV-IR can learn the Q-function by flying over $\mathcal{W}$ at various altitudes and velocities in $\mathcal{E}$. At higher altitudes, the IR measurements are typically less accurate, but more cells are measured because the FOV is larger. At lower altitudes, the FOV is smaller, translating to less cells being measured by the sensor, but the IR measurements are more accurate. As an example, the cells measured by the UAV-IR sensor along a path at $a_8 = 1.6$ km, and during the time interval $[t_0, t_f]$, are shown by the purple bins in Figure 5, using the same color legend used in Figure 4. For comparison, the cells measured along the same path at $a_{32} = 6.4$ km is shown in Figure 6.

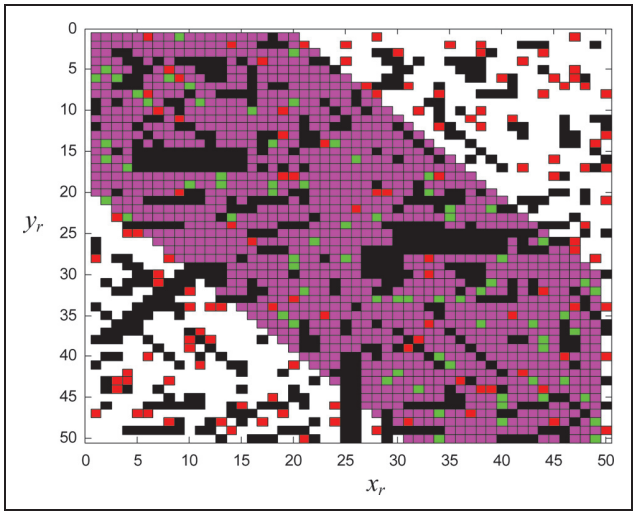

Figure 6. Cells measured by UAV-IR during $[t_0, t_f]$, at $a_{32} = 6.4$ km.

As shown in Figure 7, the percentages of mines (red bins) and clutter targets (yellow bins) that are correctly classified is higher at a_8, whereas the percentage of targets that were undetected (blue bins, labeled by 'U') is higher at a_{32}. The CL (%) of the classifications, plotted on each target detected in Figure 7, also tends to be higher at lower altitudes. While the UAV-IR explores the state space, its measurements and the actual classification of the targets are used to compute the reward in Equation (17), and to learn the Q-function from Equation (5). In this paper, the weights in the risk function (15) are chosen as $w_2 = w_3 = 1$ in order to represent a high risk for incorrectly classifying mines, and as $w_1 = w_4 = 0.1$ in order to represent a low risk for incorrectly classifying cells that contain clutter or are empty. As a result, the Q-function learned by UAV-IR, and approximated by the NN in (18), is the highly nonlinear function that is plotted in Figure 8 with respect to v_{IR}.

Once learning is completed, the greedy policy in (4) is computed by maximizing the learned Q-function using the Matlab® Optimization Toolbox.[26] This policy provides the optimal sensor's altitude at time t_{k+1} as a function of the present aircraft state (altitude and velocity) at t_k. In this study, the optimal altitude is found to be $H^* = a_{25} = 5.0$ km. Thus, as verified by the sensing performance comparison in Table 2, when the UAV-IR sensor flies at this

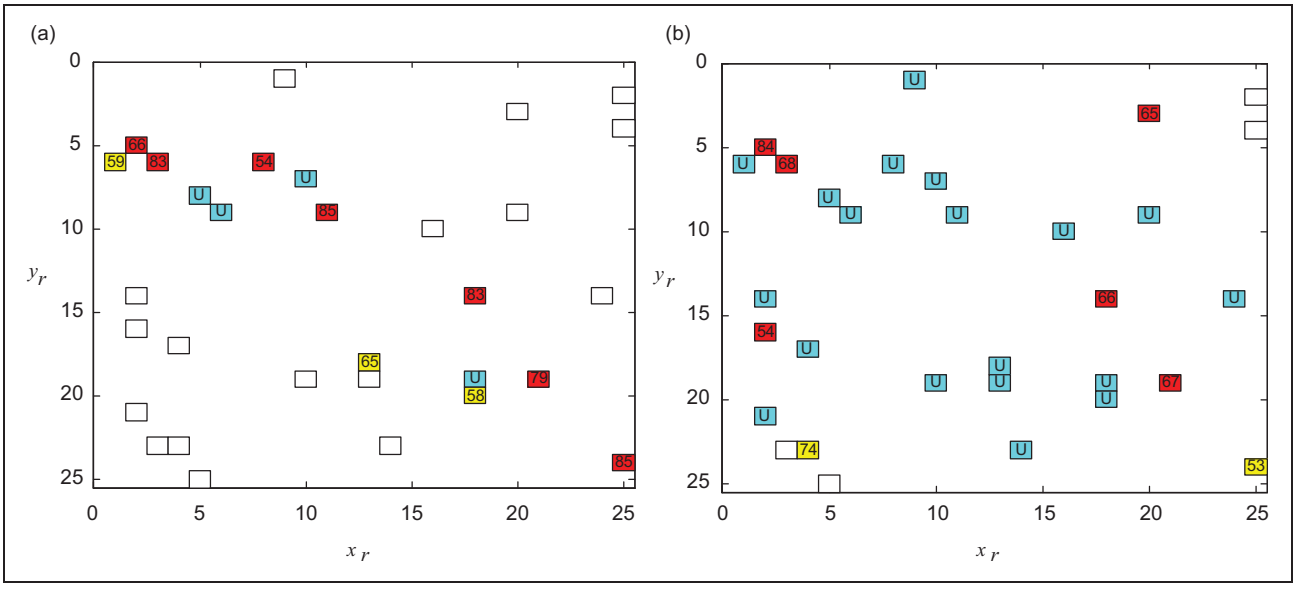

Figure 7. Classification results for the UAV-IR at (a) $a_8 = 1.6$ km and (b) $a_{32} = 6.4$ km, during a sample time interval.

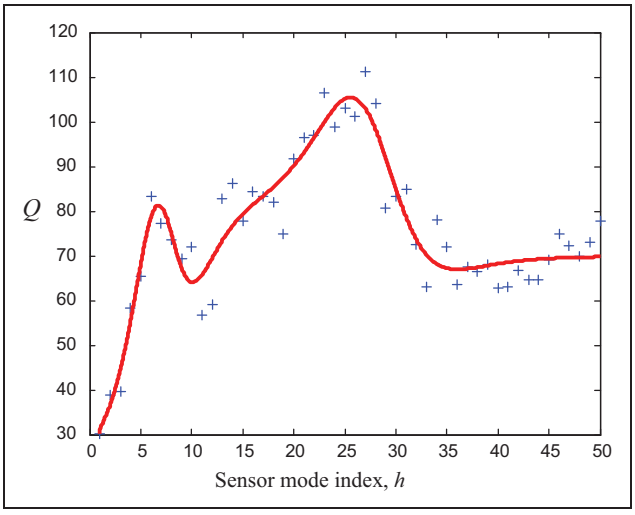

Figure 8. *Q*-function learned by UAV-IR as a function of the sensor mode.

Table 2. Performance comparison.

UAV-IR mode	Total bins	Correct mine detections (CL)	Correct clutter detections (CL)	Undetected targets
a_8	515	15 (74%)	5 (77%)	50
a_{25}	1,295	19 (69%)	8 (71%)	73
a_{32}	1,551	17 (67%)	9 (67%)	81

altitude, it obtains the highest number of correct mine detections, maximizing the mission objectives specified by the risk function (15). Future work will extend the proposed approach to heterogeneous and time-varying environmental conditions, by including E_i in the state vector x^k.

6 Conclusions and future work

In this paper we have presented a novel *Q*-learning approach to geometric sensor path planning that is applicable to UAV navigation for sensing and surveillance applications. The goal of the *Q*-learning algorithm is to determine the UAV guidance policy that maximizes the number of targets that are properly classified by the onboard IR sensor, without explicit knowledge of the UAV and sensor models, or of the environmental conditions. By this approach, the UAV-IR sensor learns the *Q*-function based solely on the actual classification of known targets buried in the ROI. Through the *Q*-function, the sensor learns about the system models implicitly, and obtains a greedy policy by maximizing the *Q*-function with respect to the control. The approach is demonstrated through an application involving an IR sensor installed onboard a UAV that flies over a two-dimensional ROI for the purpose of detecting and classifying buried targets, such as clutter, mines, or unexploded ordnance.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

1. Acar EU. Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods. *Int J Robotic Res* 2003; 22: 175–196.
2. Lazanas A and Latombe JC. Motion planning with uncertainty—a landmark approach. *Artif Intell* 1995; 76: 287–317.
3. Choset H. Coverage for robotics: A survey of recent results. *Ann Math Artif Intell* 2001; 31: 113–126.
4. Hager GD and Mintz M. Computational methods for task-directed sensor data fusion and sensor planning. *Int J Robotics Res* 1991; 10: 285–313.
5. Chen SY and Li YF. Vision sensor planning for 3cd model acquisition. *IEEE Trans Syst Man Cybernet B* 2005; 35: 894–904.
6. Cai C and Ferrari S. Information-driven sensor path planning by approximate cell decomposition. *IEEE Trans Syst Man Cybernet B* 2009; 39: 672–689.
7. Zhang G, Ferrari S and Qian M. An information roadmap method for robotic sensor path planning. *J Intell Robotic Syst* 2009; 56: 69–98.
8. Stengel RF. *Flight Dynamics*. Princeton, NJ: Princeton University Press, 2004.
9. Ferrari S and Stengel RF Model-based adaptive critic designs. In Si J, Barto A and Powell W (eds), *Learning and Approximate Dynamic Programming*. New York: John Wiley and Sons, 2004.
10. Si J, Barto A and Powell W. Learning and approximate dynamic programming. *Proc IEEE* 1992; 80: 1384–1399.
11. Bertsekas DP. *Dynamic Programming and Optimal Control, Vols I and II*. Belmont, MA: Athena Scientific, 1995.
12. Bertsekas DP and Tsitsiklis JN. *Introduction to Probability*. Belmont, MA: Athena Scientific, 2002.
13. Russell S and Norvig P. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice Hall, 2003.
14. Ferrari S and Stengel RF. Classical/neural synthesis of nonlinear control systems. *J Guidance Control Dynam* 2002; 25: 442–448.
15. MacDonald J. *Alternatives for Landmine Detection*. Rand Publications, 2003.
16. Van Dam R, Borchers B, Hendrickx J and Hong S. Soil effects on thermal signatures of buried nonmetallic landmines. In *Detection and Remediation Technologies for Mines and Minelike Targets VIII. Proc SPIE* 5089: 1210–1218.
17. Explosive, Ordnance, Disposal, (EOD), and Technicians. *ORDATA Online*, 2006. Avaliable at: http://maic.jmu.edu/ordata/mission.asp.
18. Ferrari S and Vaghi A. Demining sensor modeling and feature-level fusion by bayesian networks. *IEEE Sensors* 2006; 6: 471–483.
19. Van Dam RL. Soil effects on thermal signatures of buried nonmetallic landmines. In *Detection and Remediation*

Technologies for Mines and Minelike Targets VIII. Proc SPIE 2003; 5089: 1210–1218.
20. Jensen FV. *Bayesian Networks and Decision Graphs*. Berlin: Springer-Verlag, 2001.
21. Ferrari S and Cai C. Information-driven search strategies in the board game of CLUE® . *IEEE Trans Syst Man Cybernet B* 2009; 39: 607–625.
22. Cai C and Ferrari S. A Q-learning approach to developing an automated neural computer player for the board game of CLUE®. In *International Joint Conference on Neural Networks*, Hong Kong, 2008, pp. 2347–2353.
23. Stengel RF. *Optimal Control and Estimation*. New York: Dover Publications, Inc., 1986.
24. Murphy K. *How To Use Bayes Net Toolbox*, 2004. Avaliable at: http://www.ai.mit.edu/murphyk/Software/BNT/bnt.html.
25. Mathworks. function: trainbr. *MATLAB Neural Network Toolbox*, 2004. Avaliable at: http://www.mathworks.com.
26. Mathworks. function: fminmax. *Matlab Optimization Toolbox*, 2004. Avaliable at: http://www.mathworks.com.

Author Biographies

Silvia Ferrari is Paul Ruffin Scarborough Associate Professor of Engineering at Duke University, where she directs the Laboratory for Intelligent Systems and Controls (LISC). Her principal research interests include robust adaptive control of aircraft, learning and approximate dynamic programming, and optimal control of mobile sensor networks. She received the BS degree from Embry-Riddle Aeronautical University and the MA and PhD degrees from Princeton University. She is a senior member of the IEEE, and a member of ASME, SPIE, and AIAA. She is the recipient of the ONR Young Investigator Award (2004), the NSF Career Award (2005), and the Presidential Early Career Award for Scientists and Engineers (PECASE) (2006).

Greyson Daugherty Greyson Daugherty received his BA in physics, mathematics, and Russian language from Vanderbilt University in Nashville, TN in 2009. He then completed a Master of Engineering Management at Duke University in 2010, where he is now a PhD student in mechanical engineering. His research interests include stochastic decision theory, Bayesian and neural networks, and sensor performance optimization. He is a student member of SPIE.