Evolutionary Control of an Autonomous Field

Mark W. Owen
SSC San Diego

Dale M. Klamer and Barbara Dean
Orincon Corporation

ABSTRACT

An autonomous field of sensor nodes must acquire and track targets of interest traversing the field. Small detection ranges limit the detectability of the field. As detections occur in the field, detections are transmitted acoustically to a master node. Both detection processing and acoustic communication drain a node's power source. To maximize field life, an approach must be developed to control processes carried out in the field. This paper presents an adaptive threshold control scheme that minimizes power consumption while still maintaining the field-level probability of detection. The power consumption of the field of sensor nodes is driven by the false alarm rate and target detection rate at the individual sensor nodes in this problem formulation. The control law to be developed is based on a stochastic optimization technique known as evolutionary programming. Results show that by dynamically adjusting sensor thresholds and routing structures, the controlled field will have twice the life of the fixed field.

INTRODUCTION

The Office of Naval Research (ONR) established the Deployable Autonomous Distributed System (DADS) program (Figure 1) to demonstrate the feasibility of increased performance for an advanced tactical/surveillance system that operates as a field of underwater distributed sensor nodes. The goal of DADS is to demonstrate the feasibility of a cooperative field-level detection and data fusion system that increases performance at a reduced cost. Given limited power, the objectives are to use distributed detection and data fusion to increase the lifetime of the field (reduced power consumption), decrease the false alarm rate of the field over that of the individual nodes, increase the field-level detection, increase the probability of correct classification, and increase the accuracy of target position estimates [1, 2, and 3].

A DADS field consists of individual sensor nodes operating autonomously. Each sensor node uses a set of acoustic and electromagnetic sensors to provide coverage of a small area of interest. Each DADS sensor node uses a matched-field tracking algorithm to provide target detections consisting of position, velocity, and classification information. Once a detection is constructed at a sensor node, the data are transferred to a DADS master node where field-level data fusion is performed.

Detection Theory

In the DADS program, a need exists to identify what constitutes target detections from the field of autonomous sensor nodes. The DADS program also requires an optimization algorithm to route communication messages efficiently, using as little power as possible. A field-level control/detection scheme is sought to detect targets of interest at a given field-level probability and to route messages optimally by using a minimal amount of power. Control of an autonomous set of sensor nodes is needed to meet a desired probability of detection for the field and to extend the life of the field.

To construct a field-level detection, we now define what is required to call out a field-level detection. Each sensor node contains an acoustic sensor suite and an electromagnetic sensor suite. To report a detection, both the acoustic and magnetic sensors must detect a target at a sensor node. Once one node has detected the target, a second node nearby is cued and another sensor node must detect the target. Once this second sensor node detects and reports the target, a field-level detection is called and reported

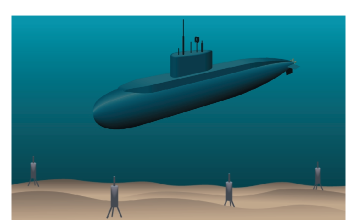

FIGURE 1. Field of DADS autonomous sensor nodes.

out by the master node for field-level fusion. Each sensor node has a threshold for the sensor suite given by an operating point on a receiver operating characteristic (ROC) curve as shown in Figure 2. The operating points on the figure are labeled R1 and R2 and represent different signal-to-noise ratio (SNR) levels for the sensor suite. Choosing different operating points on the ROC curve yields different probabilities of detection and probabilities of false alarm. A constant field-level probability of detection is desired for operation of the field of sensor nodes. By adjusting threshold levels at the sensor suite, that is, moving up and down operating points on the ROC curve at each sensor node, a constant field-level probability can be achieved.

Besides controlling the thresholds at the individual sensor suites at each node, another problem is to minimize the power consumption of the individual sensor nodes while meeting the field-level probability constraint. This issue addresses the routing of communication messages through the distributed field of sensor nodes. As messages are passed from sensor node to sensor node and finally arrive at the master node, the battery level is drained by the amount of communication power spent transmitting and relaying detections acoustically.

A field-level controller will adjust the detection threshold levels at each sensor node to meet the desired field-level probability of detection and to perform optimal routing of messages through the field. A typical example of a point on a ROC curve is shown in Figure 3.

A brief overview of detection theory is provided below [4]. In Figure 3, two possible hypotheses, labeled H0 and H1, are shown. H0 is the false alarm hypothesis and H1 is the detection hypothesis. The threshold T is used to determine whether or not the SNR is high enough to call out a detection. The SNR in the figure is labeled γ. Under the two Gaussian curves, a probability of detection and a probability of false alarm can be determined. Integrating the H0 probability density function (pdf) from T to ∞, the false alarm probability is calculated. Integrating the H1 pdf from T to ∞, the probability of detection is calculated. Figure 4 shows several SNRs from a chosen ROC curve operating point. The objective of the field-level controller is to adapt the sensor node thresholds to acquire a target of interest and detect it successfully through the field. In the figure, the graph labeled nominal is shown to demonstrate a chosen operating point for the sensor node. The next two graphs show a decrease in SNR and an increase in SNR, respectively. As SNR levels vary, a target may become easier or more difficult to detect although the probability of false alarm remains constant across all three graphs. Only the probability of detection decreases or increases due to the SNR of the target. Our task is to adjust thresholds dynamically to make sure the target is acquired and tracked as it passes through the field. To do this, we will lower thresholds for subsequent cued detections to increase the detection range at a sensor node, but at the same time we increase the number of false alarms from a sensor node. When adjusting these thresholds at each sensor node, we must maintain a constant field-level probability of detection. A simple example of this threshold adjustment is to use a bathtub analogy. If one side of the bathtub water is pushed down, water on the other side of the tub will rise. This example shows what we will do when adapting thresholds: we will lower a certain set of sensor node thresholds while raising another set.

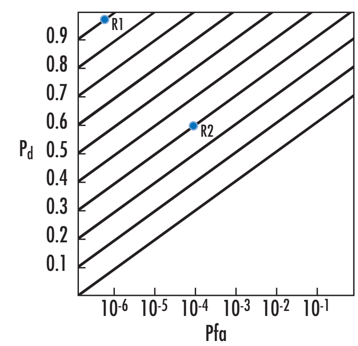

FIGURE 2. Typical ROC curve.

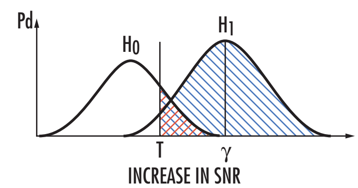

FIGURE 3. A single point from a ROC curve.

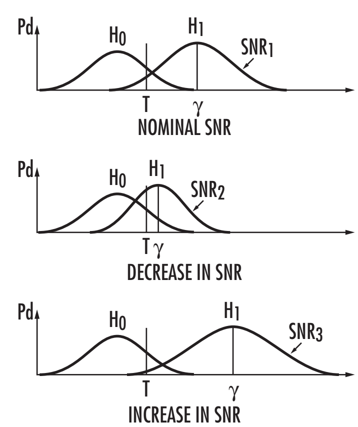

FIGURE 4. Possible detection curves.

Threshold Adaptation

Figure 5 shows a cookie cutter example of a field of sensor nodes. Each sensor node has a defined detection range given in red (small circles) for a high threshold (low false alarm rate, high SNR) and another detection range shown in blue (large circles) for a low threshold (high false alarm rate, low SNR). This figure demonstrates the adaptive process that must occur for the DADS field of sensor nodes to detect and continue to detect a target as it passes through the field.

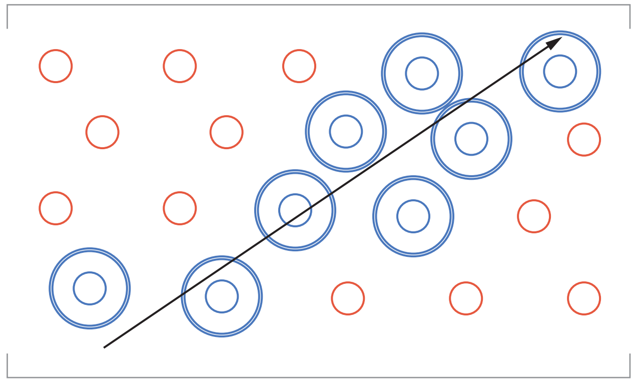

FIGURE 5. Sensor node threshold adjustments via field-level control.

If the field were static, the small red circles would dictate the area of coverage in which the field could pick up detectable targets. In the figure, a hypothetical target has been drawn by a black line with an arrow at the tip. If the threshold were held at this higher level, only one possible detection might occur as this target traversed the field of sensor nodes. By lowering the thresholds (larger blue circles), which is done by cueing the field, a broader coverage of the field is achieved. The figure shows that up to four possible detections on a target of interest can occur by lowering the sensor node thresholds. This improved detectability concept will improve the overall field-level data fusion by providing more contact information than previously capable with a static set of sensor node thresholds. By lowering the threshold though, a larger number of false alarms can occur and cause power to be drained from the sensor nodes. False alarms also make the data fusion problem at the master node more susceptible to miscorrelation. Therefore, dropping all of the sensor node thresholds is not acceptable because it will limit the system operation. As explained previously, we will lower thresholds and raise thresholds at individual sensor nodes to maintain the desired field-level probability of detection while maximizing the life of the field.

2-of-2 Field Detector

To adjust thresholds, we propose to use a baseline model of a 2-of-2 detector. The detector will use communication costs, probabilities of detection and false alarm, node spacing of the field, and signal processing parameters used at the sensor node sensor suite. This formulation shows that false alarms as well as target detections drain the power at each sensor node. We will now present our baseline model equation for field-level control as derived in [5]. This formulation will allow the complete field to be controlled by the master node in the DADS system. The baseline model equation is as follows. The estimated power $\hat{P}^{(n)}$ consumed over a period of time T at each node n, $n = 1,\ldots, N$, is given by

$$\begin{aligned}\hat{P}^{(n)}(T) = &\sum_{k=1}^{\rho_s T} C_{on} + \left[1-\left(1-F_1^{(n)}F_2^{(n)}\right)^{N_p}\right]C_k^{(n)} \\ &+ \sum_{n' \in R_k^{(n)}} \left[1-\left(1-F_1^{(n')}F_2^{(n')}\right)^{N_p}\right]C_k^{(n)} \\ &+ \sum_{n' \in B_k^{(n)}} \left[1-\left(1-F_1^{(n')}F_2^{(n')}\right)^{N_p}\right]\left[1-\left(1-F_1^{(n)}F_2^{(n)}\right)^{\rho_s \delta N_p P[1+sD^2]/(\pi(r_d^{(2)})^2)}\right]C_k^{(n)} \\ &+ \sum_{n' \in R_k^{(n)}} \sum_{n'' \in B_k^{(n)}} \left[1-\left(1-F_1^{(n'')}F_2^{(n'')}\right)^{N_p}\right]\left[1-\left(1-F_1^{(n')}F_2^{(n')}\right)^{\rho_s \delta N_p P[1+sD^2]/(\pi(r_d^{(2)})^2)}\right]C_k^{(n)} \qquad (1)\end{aligned}$$

$$
\begin{aligned}
&+ \rho_T \mathrm{r_d}^{(n)}\left[1-(1-P_1^{(n)}P_2^{(n)})(1-F_1^{(n)}F_2^{(n)})^{N_p-1}\right]C_k^{(n)} / D \qquad \text{(1 contd)}\\
&+ \rho_T \sum_{n' \in R_k^{(n)}} \mathrm{r_d}^{(n)}\left[1-(1-P_1^{(n')}P_2^{(n')})(1-F_1^{(n')}F_2^{(n')})^{N_p-1}\right]C_k^{(n)} / D\\
&+ \rho_T \sum_{n' \in B_k^{(n)}} \mathrm{r_d}^{(n)}\left[1-(1-P_1^{(n')}P_2^{(n')})(1-F_1^{(n')}F_2^{(n')})^{N_p-1}\right]\\
&\left[1-(1-P_1^{(n)}P_2^{(n)})(1-F_1^{(n)}F_2^{(n)})^{\left[\rho_s \delta N_p P[1+sD^2]/(\pi(r_d^{(2)})^2)\right]-1}\right]C_k^{(n)} / D\\
&+ \rho_T \sum_{n' \in R_k^{(n)}} \sum_{n'' \in B_k^{(n')}} \left[1-(1-P_1^{(n'')}P_2^{(n'')})(1-F_1^{(n'')}F_2^{(n'')})^{N_p-1}\right]\\
&\left[1-(1-P_1^{(n')}P_2^{(n')})(1-F_1^{(n')}F_2^{(n')})^{\left[\rho_s \delta N_p P[1+sD^2]/(\pi(r_d^{(2)})^2)\right]-1}\right]C_k^{(n)} / D
\end{aligned}
$$

where ρ_s is the basic sample rate and T is the time period of the estimated life of the node. The first term represents the power consumed C_{on} from the processor in the node. If the sensor node is on, a certain amount of processing power is drained from the battery. The second term represents the case that an initial false alarm is generated at node n, where $F_1^{(n)}$, $F_2^{(n)}$ are the probabilities of false alarm that are controlled by thresholds $T_1^{(n)}$ and $T_2^{(n)}$, and $C_k^{(n)}$ is the communication power used to transmit from node n to the next upstream node specified by the current communication route $R_k^{(n)}$ at time k. N_p is the size of the parameter space over which the detectors must test, e.g., if the detector must look over a discrete set of speed (say N_s) and closest point of approach (CPA), say N_{CPA} , thus giving $N_p = N_s N_{CPA}$. This is the second detection required for declaring a field-level detection from the field. The third term represents the case of a "downstream" node n' that generates a false alarm and node n is simply a passthrough; the communication route for node n at time k is specified by $R_k^{(n)}$. The fourth term represents the case that a false alarm is generated at node n as the result of being cued by another node n' in a set of neighboring nodes $B_k^{(n)}$. Specifically, P is the covariance of the track estimate at the time of the detection at the first node; $[1+sD^2]$ is the expansion factor for the track covariance until the second detection at the next node detection; $\pi(\mathrm{r_d}^{(2)})^2$ is the area of the detection space for the second sensor node; and D is the length of the sensor field. The fifth term represents the case of a downstream node n' that generates a false alarm as a result of being cued, and node n is simply a passthrough. The last four terms deal with the cases of a target present; ρ_T is the target rate. The sixth term represents a target detection at node n, where $P_1^{(n)}$, $P_2^{(n)}$ are the probabilities of detection, again controlled by the thresholds $T_1^{(n)}$ and $T_2^{(n)}$. This is a true target detection and not a false alarm. The seventh term represents the case of a downstream node n' detection where node n is simply a passthrough for the initial condition. The eighth term represents the case that a target detection is generated at node n as the result of being cued by another node n'. The final term represents a downstream node n' that generates a target detection as the result of being cued, and node n is simply a passthrough.

Given the current power $P^{(n)}$ available at each node, the estimated remaining power is

$$\varepsilon^{(n)}(T) = P^{(n)} - \hat{P}^{(n)}(T) .$$

The objective function for maximizing the life of the field is

$$\text{maximize } T,$$

subject to the constraints that each of the estimates of the remaining power is positive

$$\varepsilon^{(n)}(T) \geq 0 \,,\; n = 1,\ldots, N$$

and the field-level probability of detection is specified by

$$\begin{aligned} \mathrm{PD} = {} & N(\varepsilon_1, \varepsilon_2,\ldots, \varepsilon_N)\pi(r_d^2)\left[1 - (1 - P_1^{(1)}P_2^{(1)})(1 - F_1^{(1)}F_2^{(1)})^{N_p - 1}\right] \\ & \times\left[1 - (1 - P_1^{(2)}P_2^{(2)})(1 - F_1^{(2)}F_2^{(2)})^{[\rho\delta N_p P(1+sD^2)/(\pi r_d^{\,2})]-1}\right]/A(D) \end{aligned}$$

where $N(\varepsilon_1, \varepsilon_2,\ldots, \varepsilon_N)$ is the number of nodes with nonzero power remaining and $\pi(r_d^2)/A(D)$ is the area covered by an individual node. The objective is to maximize field life T subject to meeting the field-level constraint by adjusting probability of detection / probability of false alarm threshold levels and varying communication routes (through $R_k^{(n)}$). By choosing appropriate thresholds at each sensor suite, the field-level probability of detection constraint can be met and the field life extended. An algorithm that will choose thresholds to meet the probability of detection constraint and extend the field life is discussed in the next section.

Evolutionary Programming

Evolutionary programming (EP) is a stochastic optimization technique applied in this paper to optimize routing of the sensor node message traffic at minimal power cost and to meet a field-level probability constraint. EP falls under the domain of Evolutionary Computation that contains other algorithmic techniques such as genetic algorithms (GAs), genetic programming, as well as others [6]. One of the main differences between EP and GAs is that EP performs a mutation operation while GAs perform a mutation operation and a crossover operation. Genetic algorithms also operate from the bottom up when finding a solution. EP is a top down approach to finding optimal solutions. An evolutionary algorithm is shown in Figure 6. In simple terms, an evolutionary algorithm starts out with a population of possible solutions to a problem. A population consists of parent solutions and their corresponding offspring solutions. This stochastic optimization technique allows the whole parameter space to be searched and evaluated for a best-fitting solution. In the figure, the initial solutions are called parents. Each parent solution can be a good first guess at the correct answer or a randomly chosen solution that may be very poor. Each parent has the ability to create a set of offspring solutions by mutation or by crossover if a genetic approach was used. Each parent solution is mutated by changing its state to form an offspring solution. This mutation can be Gaussian or some other linear or nonlinear deviation. Once the population of parents has been mutated and the offspring solutions are created, the population consisting of parents and offspring solutions is then scored, as shown in the figure. Scoring or evaluation of the population for our purpose is done to make sure the sensor nodes meet a defined field-level probability constraint with their defined threshold settings. A selection process is then performed whereby the next generation of parents are selected to evolve better and better solutions. This selection process chooses the solutions that passed the constraint in the scoring process by selecting the solutions that yield the largest amount of field life.

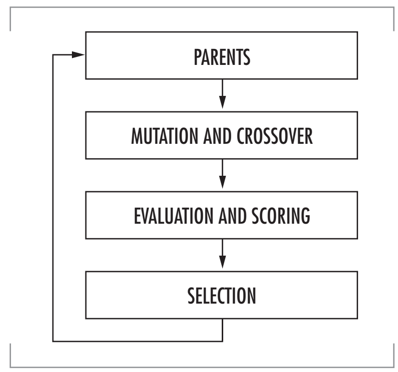

FIGURE 6. Evolutionary algorithm.

The standard EP approach consists of several steps (initialization, mutation, scoring, and selection) [6]. Initialization is performed by assigning thresholds to each sensor in the sensor suite (magnetic, acoustic) and using these thresholds, the sonar equation, and an error function to evaluate the probability of detection and probability of false alarm of the sensor node. This is done for each sensor node in the field given by

$$Pd(n) = 1/2^{*}(1.0 - erf(T(n) - SL(n) + NL(n))) \quad (2)$$

and

$$Pd(n) = 1/2^{*}(1.0 - erf(T(n) + NL(n))) \quad (3)$$

where Eq. (2) initializes the probability of detection Pd for sensor node n given its threshold T, the target source level SL, and the noise level at the sensor NL. Eq. (3) initializes the probability of false alarm Pfa for sensor node n given its threshold T, and the noise level at the sensor NL. This is performed for each sensor node until all thresholds and probabilities of detection and false alarm have been initialized. This fully initialized field of sensor nodes is deemed as a parent solution in the EP language and is a possible solution for the field-life problem. Possible solutions are defined as parents and are given as

$$P(k) = S(Pd(n), Pfa(n), T(n), R(n)) \quad (4)$$

where $P(k)$ are the k number of parents in the population solutions. Each solution S is made up of a field of sensor nodes with independent thresholds T, which dictate a Pd and Pfa for the sensor node, and a routing table R for communication with other nodes in the field. Once the population of parent solutions has been initialized, the EP algorithm is able to perform the next three steps (mutation, scoring, and selection) iteratively to converge to the best possible solution given time constraints and memory requirements of the system. The first step is the mutation process whereby parent solutions generate offspring solutions. Offspring solutions have the possibility of generating a better solution than their parents. This is the evolutionary step in the EP process. One of the mutation steps is to change the threshold at each sensor at a sensor node to yield a better solution. This is defined by

$$O[T(m,n)] = P[T(k,n)] + N(0,1) \quad (5)$$

where $O[T(m,n)]$ is the mutated threshold at offspring m for sensor node n, $P[T(k,n)]$ is the threshold at parent k for sensor node n, and $N(0,1)$ is a Gaussian random variable with zero mean and unit variance. Eq. (5) changes each parent's threshold to generate an offspring's threshold. Another mutation step is to change the routing table for communications at each node. This is defined by

$$O[R(m,n)] = P[R(k,n)] \pm Urv^{*}c \quad (6)$$

where $O[R(m,n)]$ is the mutated communication routes at offspring m for sensor node n, $P[R(k,n)]$ is the communication routes at parent k for sensor node n, Urv is a Uniform random variable, and c is the number of possible nodes for sensor node n to communicate with. The number of communication routes can increase or decrease according to Eq. (6). Eq. (6) changes each parent's communication route to generate an offspring's communication route. Each parent can perform these mutation steps and generate as many offspring as desired. Once this is done, the new population of parents and offspring are scored and evaluated against the system constraints. For example, if the desired field-level probability of detection is 0.8, each solution is evaluated using

$$\mathrm{PD} = N\varepsilon_1, \varepsilon_2,\ldots, \varepsilon_N)\pi(\mathrm{r_d}^2[1-(1-P_1^{(1)}P_2^{(1)})(1-F_1^{(1)}F_2^{(1)})^{N_p-1}]$$
$$\times[1-(1-P_1^{(2)}P_2^{(2)})(1-F_1^{(2)}F_2^{(2)})^{[\rho\delta N_p P(1+sD^2)/(\pi r_d^{\ 2})]-1}]/A(D) \quad (7)$$

which is the probability of detection for a field of sensor nodes defined above. (See 2-of-2 Field Detector.) We will use a simulated annealing approach to meet this constraint. For example, if 0.8 is desired, we may allow solutions to lie between (0.7, 0.9) in the beginning and slowly converge toward 0.8 while we iterate. All solutions that pass this field-level probability constraint are then passed to the selection process. Selection is done by picking the best k solutions that meet the constraint and minimize the power consumption defined from the baseline model from Eq. (1). These best k solutions then become the parents for the next iteration. The process continues until the best solution is found. This evolutionary process extends the field life by optimizing the thresholds of the field and planning the optimal routes for message passing.

RESULTS

Now we present some results of our EP solution to the adaptive threshold control problem. These results are for a complete field of sensor nodes. Each node has a set of thresholds solved for by the EP algorithm as well as the optimal routes for communication to extend field life.

Simulation Overview

As stated previously, the claim of this paper is that it can be shown that field life can be doubled by using a field-level controller to dynamically adjust thresholds and routing structures, as compared to a fixed field that uses static thresholds and routing structures.

The EP software written for this paper generates solutions that are representative of a field under the control of a field-level controller. To make the comparison to a fixed field, a fixed-field implementation had to be generated.

The Fixed Field

The fixed field required a nominal routing structure and a set of sensor thresholds, which would meet the field-level probability of detection. To generate the nominal routing structures, a field initialization scheme was emulated. The emulation of this field initialization scheme consists of the following steps:

1. The Master Node broadcasts a Wakeup Message.
2. Any node that can hear responds with a Wakeup Response message. In this case, any node within the cookie cutter range can hear.
3. Nodes that responded to the Master Node will be direct communication routes. This means that these nodes will relay their packets directly to the master node.
4. Nodes that heard the Master Node will broadcast to their neighbors.
5. Any node that can hear within the cookie cutter range will respond.
6. If the node that responds does not have a destination node yet, the node that broadcast will become the destination node.
7. This sequence is repeated until every node in the field has been assigned exactly one destination node.

The above sequence generated a nominal routing structure for a fixed field as shown in Figure 7. In conjunction with the routing structures, sensor thresholds that met the field-level probability of detection were required. To obtain these thresholds, the EP model was run, and the thresholds from the optimal solution were used.

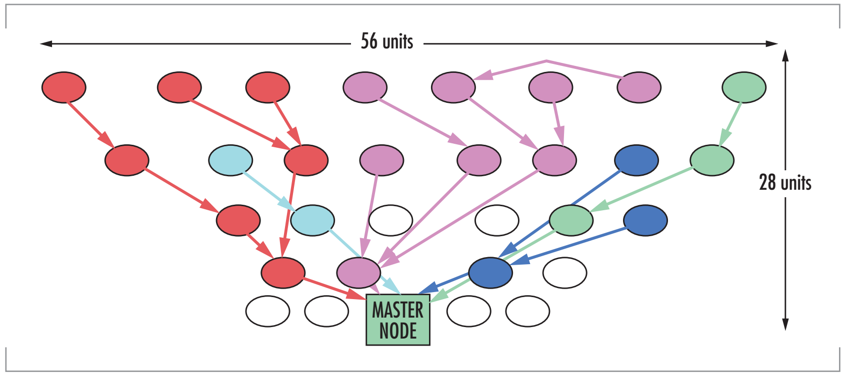

FIGURE 7. Fixed-field routes.

The Controlled Field

In the simulations, two types of results are generated for the controlled field. The first type is referred to as a "single optimized" solution. This solution is generated using the EP software. Once the EP algorithm finds an optimal combination of thresholds and routing structures, it uses that solution for the life of the field. Figure 8 shows the optimal routes found for the single optimized solution.

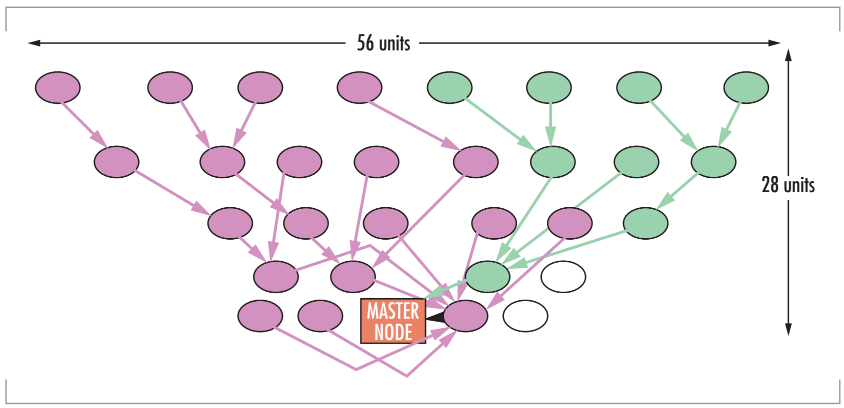

FIGURE 8. Single optimized field routes.

The second type of a controlled field solution is referred to as a "vector-optimized" solution. As with the single optimized solution, the EP algorithm finds a solution set, which maximizes field life. However, in this solution, the routes and thresholds can be adjusted every 24 hours, thus resulting in a vector of solutions. Because the control algorithm is run each day and the routes are potentially changed, it is not possible to show each daily graphical solution in this paper.

Field Laydown

Simulations were run for two field laydowns. In each laydown, the field consists of 30 sensor nodes and 1 master node arranged in a (56 by 28) unit grid. The difference between the two laydowns is the placement of the master node. In the first field laydown, the master node is a square box on the edge of the field as shown in Figures 7 and 8. In the second laydown, the master node is in the center of the field of sensor nodes.

Detector Types

The objective function defined previously (see 2-of-2 Field Detector) is for a 2-of-2 detector. This paper also defined an objective function for a 1-2 detector. The 1-2 detector requires an initial detection from the magnetic sensor on one node followed by a confirmed detection from the acoustic sensor on a second node. Results for both the 2-of-2 detector and the 1-2 detector are reported below.

Simulation Results

The results from the simulation are given in Table 1. The results are provided in units of days.

Figure 9 shows the results from running the fixed-field simulation. In the fixed field, the routing assignment was performed by using the minimum number of hops between the master node and each node in the field. This result is for the 2-of-2 detector processing for the second field laydown. It shows that running no optimization algorithm and just a greedy algorithm to assign a route for the field only yields a field life of 74 days. As shown in Figure 9, one single node begins to lose its power immediately. This node is the main communication node to the master node. Once one node in the field loses all of its power, the field is considered to be dead.

TABLE 1. Simulation results in days.

Field Laydown	Detector	Fixed Field	Single Optimized	Vector-Optimized
1	1-2	21	32	45
	2-of-2	40	70	118
2	1-2	26	45	55
	2-of-2	74	106	154

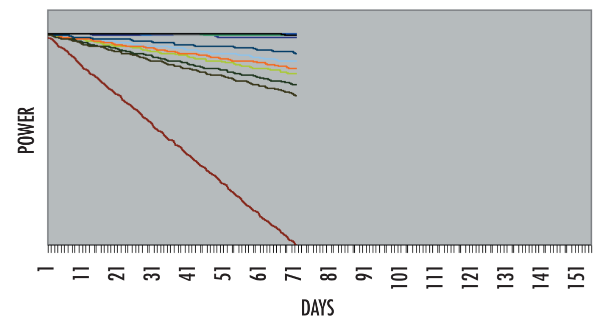

FIGURE 9. Fixed-field life.

Figure 10 shows the results from the single optimized field simulation. The routes for this result were calculated by running the EP algorithm once for the whole life of the field. This optimization result yielded a field life of 106 days for the 2-of-2 detector for the second field laydown. As shown in this figure, a single node still drives the field to death, but there are several other sensor nodes that are also losing power at a similar rate.

The field life was extended over the fixed-field implementation by using at least one planned optimal route for the whole simulation.

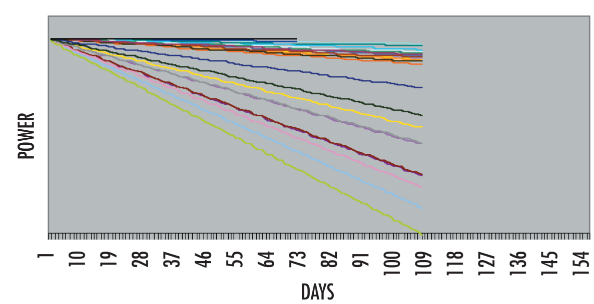

FIGURE 10 . Single optimized field life.

Figure 11 shows the results from the vector-optimized field simulation. This result has its routes recalculated each day by running the EP optimization algorithm. This optimization result yielded a field life of 154 days for the 2-of-2 detector for the second field laydown. As shown in this figure, a group of sensor nodes all lose power similarly at the same rate. Approximately one-third of the sensor nodes in the field died on day 154. This result more than doubled the life of the field over the fixed-field result of Figure 9. It also increased the life of the field from 106 days for the single optimized solution shown in Figure 10 to 154 days for the vector-optimized solution.

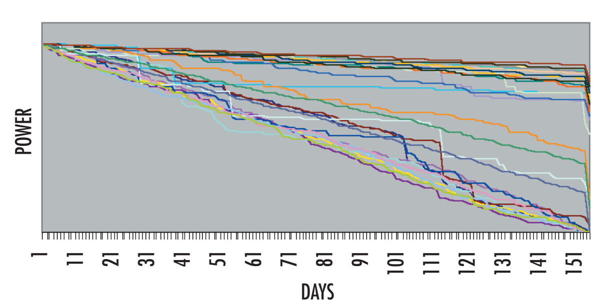

FIGURE 11. Vector-optimized field life.

Observations

The following observations are made regarding the simulation results:

1. The vector-optimized solution more than doubled field life as compared to the fixed-field solution.
2. The 2-of-2 detector has a longer life than the 1-2 detector. This is because the 2-of-2 detector has stringent initial detection rules, which translates to fewer reports and less communication as shown in Table 1.

3. Field life increased when the master node was moved from the edge of the field to the center of the field for the second field laydown. This is because when the master node is in the center of the field, there are more direct routes to the master node, which spreads out battery drain.
4. The vector-optimized solution has a longer field life than the single optimized solution. This is because changing the routes every 24 hours allows the battery drain to be spread more evenly across the field. With the vector-optimized solution, approximately one-third of the field will die on the same day.

CONCLUSIONS

In this paper, we have applied a stochastic optimization technique to adapt the thresholds of an autonomous sensor field and plan the communication routes. This stochastic optimization algorithm is known as evolutionary programming. The evolutionary program adapted the thresholds of a 2-of-2 detector for a set of sensors as well as a 1-2 detector. The algorithm is an evolutionary computation technique where an analytic solution is not attainable mathematically. Each sensor node in the 2-of-2 detector contained two thresholds to adapt, yielding four total thresholds to compute. The four thresholds are combined to meet a field-level probability of detection constraint and extend the life of a field of sensor nodes. Results show the benefits of adaptive threshold control in an autonomous sensor field by reducing communication costs and extending the life of the field by two.

AUTHORS

Dale M. Klamer

MS in Mathematics, San Diego State University, 1972.
Current Research: Adaptive data fusion; advanced tracking; information assurance.

Barbara Dean

BS in Electrical Engineering, Widener University, 1989
Current Research: Software engineering for Navy sonar systems.

Mark W. Owen

MS in Electrical Engineering, California State University Long Beach, 1997
Current Research: Data fusion; signal processing; autonomous control.

REFERENCES

1. Hatch, M., M. Owen, et al. 1998. "Data Fusion Methodologies in the Deployable Autonomous Distributed System (DADS) Project," *Proceedings of the International Conference on Multisource-Multisensor Information Fusion*, (July), Las Vegas, NV, pp. 470-477.
2. Jahn, E., J. Kaina, and M. Hatch. 1999. "Fusion of Multi-Sensor Information from an Autonomous Undersea Distributed Field of Sensors," *Proceedings of the Second International Conference on Multisource-Multisensor Information Fusion*, (July), Sunnyvale, CA, pp. 4-11.
3. Shea, P. and M. Owen. 1999. "Fuzzy Control in the Deployable Autonomous Distributed System," *Proceedings of SPIE: Signal Processing, Sensor Fusion, and Target Recognition VIII 1999*, (Ivan Kadar, ed.), vol. 3720, (April), Orlando, FL.

4. Helstrom, C. W. 1968. *Statistical Theory of Signal Detection*, Pergamon Press, New York.
5. Klamer, D. and M. Owen. 2000. "Adaptive Threshold Control in an Autonomous Sensor Field," *Proceedings of SPIE: Signal and Data Processing of Small Targets 2000*, (Oliver Drummond, ed.), vol. 4048, (April), Orlando, FL.
6. Fogel, D. B. 1995. *Evolutionary Computation*, IEEE Press, New York.

❖