Hybrid Evolutionary Algorithms for Sensor Placement on a 3D Terrain

Haluk Rahmi Topcuoglu
Computer Engineering Department,
Marmara University,
Kuyubasi, 34722, Istanbul, Turkey
Email: haluk@marmara.edu.tr

Murat Ermis
Industrial Engineering Department,
Turkish Air Force Academy,
Yesilyurt, 34149, Istanbul, Turkey
Email: m.ermis@hho.edu.tr

Mesut Sifyan
Computer Engineering Department,
Marmara University,
Kuyubasi, 34722, Istanbul, Turkey
Email: mesutsifyan@marun.edu.tr

***Abstract*—In this paper, we propose a framework for deploying and configuring a set of given sensors in a synthetically generated 3-D terrain with multiple objectives on conflicting attributes: maximizing the visibility of the given terrain, maximizing the stealth of the sensors and minimizing the cost of the sensors used. Because of their utility-independent nature, these complementary and conflicting objectives are represented by a multiplicative total utility function model, based on multi-attribute utility theory. In addition to theoretic foundations, this paper also present a hybrid evolutionary algorithm based technique to solve the sensor placement problem. It includes specialized operators for hybridization, which are problem-specific heuristics for initial population generation, intelligent variation operators which comprise problem specific knowledge, and a local search phase. The experimental study validates finding the optimal balance among the visibility, the stealth and the cost related objectives.**

***Keywords*-Sensor planning; multi-attribute utility theory; hybrid genetic algorithms.**

I. Introduction

In the military operations, reconnaissance, surveillance and target acquisition can include a plurality of sensor platforms that are used to collect information about an area under surveillance and play a vital role [1]. In order to detect position of foes, some sensors should be placed to cover a certain terrain to provide maximum visibility while maintaining sensors' stealth.

In this paper, we develop a framework and a novel solution approach for determining the optimal number of sensors, locating and setting their orientational sensor-specific parameters in a synthetically generated 3-D terrain with multiple objectives. Our solution approach relies on rational trade-off between three conflicting objectives which are maximizing the coverage area while maintaining the maximum stealth, and minimizing the total acquisition cost of deploying the sensors.

Motivated by our constructed framework, this paper explores the employing of a hybrid evolutionary algorithm for sensor placement and orientation problem. Simple evolutionary algorithms are generally poor for solving the complex combinatorial problems [2]. GAs are usually strengthen with the domain-specific characteristics [3], [4], [5], and they are combined with specialized heuristics to produce hybrid systems, which are called with different names including hybrid evolutionary algorithms and memetic algorithms [6].

In this paper, we propose two specialized crossover operators called the Contribution-Based Crossover (CBX) and the Proximity-Based Crossover (PBX) that comprise the domain specific information on sensor placement problem, and a local search technique for improving the quality of solution. Experiments on synthetic 3-D terrains with various characteristics are conducted in order to present the effectiveness of our GA-based framework. The results of the experimental study clearly show that our proposed approach is very successful in deploying and utilizing sensors by considering the multiple objectives.

The remainder of the paper is organized as follows: In Section 2, we first present an overview to Multiple Attribute Utility (MAU) theory, which is followed by our novel multi-attribute utility function model and its sub-objective formulations. Section 3 gives our hybrid GA-based formulation for solving the sensor optimization problem. Performance evaluation and experimental study is discussed in Section 4; and Section 5 concludes the paper.

II. Multi-Attribute Utility Function for Sensor Optimization Problem

Utility analysis is a widely preferred multi-objective optimization method, since it enables, especially in military applications, the testing of various scenarios (such as risk averse, risk prone, etc.) under uncertainty or limited data cases. If there is more than one objective, utility analysis gets very complicated and can only be applied if certain decomposition conditions (additivity, independency, etc.) are met [7], [8].

If there is more than one objective which are both conflicting and at the same time supporting each other and if all objectives are independent, multiplicative or multi-linear utility function may be used. Our basic assumption states that in a military operation, the perception behavior of the sensors should rely on rational trade-off between three conflicting criteria (priorities). These three criteria are: maximizing the information about the land, minimizing the data collected by the enemy and minimizing the total cost of the positioned sensors.

In our formulation, the total utility value is computed as a multiplicative function over the given three attributes. Formally, the total function to maximize, $U(A, S, P)$, of scanning an area A using a set S of sensors which are located on a set

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.127

P of polygons (i.e. the sensor s_i located on the polygon p_k), is represented with the following formulation:

$$\begin{aligned} U(A,S,P) = w_{vis}\,U_{vis}(A,S,P) + w_{st}\,U_{st}(A,S,P) + \\ w_{cost}\,U_{cost}(S,P) + w_{vis}\,w_{st}\,U_{vis}(A,S,P)\,U_{st}(A,S,P)\ + \\ w_{vis}\,w_{cost}\,U_{vis}(A,S,P)\,U_{cost}(S,P)\ + \\ w_{st}\,w_{cost}\,U_{st}(A,S,P)\,U_{cost}(S,P)\ + \\ w_{vis}\,w_{st}\,w_{cost}\,U_{vis}(A,S,P)\,U_{st}(A,S,P)\,U_{cost}(S,P) \end{aligned} \tag{1}$$

where $U_{vis}(A,S,P)$ is the utility of visibility of area A by the set of sensors S located on the set of polygons P; $U_{st}(A,S,P)$ is the utility of stealth of the set of sensors S located on set of polygons P and $U_{cost}(S,P)$ is the utility of the cost of the sensors S located on set of polygons P. In this equation, $w_{vis},\ w_{st},\ w_{cost}$ are the weights (coefficients) of visibility, stealth and cost utility functions, respectively, where $0 \le w_{vis}, w_{st}, w_{cost} \le 1$ and $w_{vis} + w_{st} + w_{cost} = 1$. These weights are set based on experimentation on a given terrain by considering various military scouting missions.

Additionally, we also consider the total utility of each sensor s_j located on polygon p_k in our computations, which is represented by $U^j(A,s_j,p_k)$. By using the Equation 1, this term requires $U^j_{vis}(A,s_j,p_k)$, $U^j_{st}(A,s_j,p_k)$ and $U^j_{cost}(A,s_j,p_k)$ terms, which are the sensors-specific utility of visibility, stealth and cost, respectively.

A. Computing the Utility of Visibility

The value of utility of visibility is derived by using the amount of visibility of the terrain, which is computed by adding the visibility of all polygons on it. Formally, the utility of visibility of area A by the set of sensors S (located at set of polygons P), $U_{vis}(A,S,P)$ is computed using the Equation 2,

$$U_{vis}(A,S,P) = \frac{\sum_{p_i \in A} V(S,P,p_i) \times W_{p_i}}{\sum_{p_i \in A} W_{p_i}} \tag{2}$$

where, W_{p_i} is the weight of the polygon p_i, which indicates the importance of the polygon, and $V(S,P,p_i)$ is the visibility value of polygon p_i by using the set of sensors S located on the set of polygons P, which is computed by the average of visibility of the points on polygon p_i. Since any point can be recognized by multiple sensors (with different visibility values), maximum visibility of the point is considered in computing the visibility value of polygon p_i. In our study, four points, i.e., the three corner points and the center of mass, are considered as the selected polygon points. The term $V_{s_j}(s_j,p_k,b)$ is the visibility of the destination point b (which can be one of the four points of polygons) from the sensor s_j located at source point a (which is the center of mass of polygon p_k). This term formally defined by Equation 3,

$$\begin{aligned} V_{s_j}(s_j,p_k,b) = (1 - \eta_{s_j} \times \frac{D(a,b)}{\Delta_{s_j}}) \times (1 - \max_{p_c \in a \to b} \psi^W_c) \\ \times (1 - \max_{p_c \in a \to b} \psi^O_c) \end{aligned} \tag{3}$$

where $D(a,b)$ is the distance between point a and point b; Δ_{s_j} is the depth of view and and η_{s_j} is the range effect coefficient of sensor s_j. The range values varies with respect to different types of the problem addressed; i.e., there will be three different range values (for detection, recognition, identification) of each sensor.

Weather density (ψ^W_i) and object density (ψ^O_i) of polygon p_i, where their values are in the range $0 \le \psi^W_i, \psi^O_i \le 1$, are used to compute the permeability value of a ray that traverses from an origin to a predefined destination through the given polygon p_i by considering object and weather conditions over the polygon.

The second term is the *weather permeability value*, which is derived by the density values of weather conditions. The term ψ^W_c is the weather density over a polygon p_c where p_c is a polygon that is in between point a and point b. The density values of weather conditions over all polygons in between points a and b are considered as part of LOS algorithms ([9], [10]) and the maximum value is returned if no intermediate point (between a and b) is obstructed by terrain.

The last term in Equation 3 is the *object permeability value*, which is set by using the density value of objects. There can be two types of objects located on synthetically generated terrain, which are natural objects such as trees and artificial objects such as buildings. It should be noted that the values of weather density (ψ^W_c) and object density (ψ^O_c) of polygon p_c are in the range $0 \le \psi^W_c, \psi^O_c \le 1$.

B. Determining the Utility of Stealth

The utility of stealth value for a set of sensors that are already located on the terrain is derived by subtracting the cost of the total visibility of the located sensors (by using enemy or opponent objects) from one. For this purpose, a predefined number of opponent objects of different types are scattered across the terrain randomly (by utilizing angle and distance constraints), as part of a given scenario *m*. These objects are the vehicles carrying opponent sensors. In our experimental study, the angle-based locational attributes (such as viewing angle, depth of view etc.) of opponent sensors can be set with those values of either best or worst sensor in our system.

The utility of stealth of a set of S sensors that are positioned on a set of polygons P by considering r different scenarios (for setting the opponent objects) is computed with the following equation

$$U_{st}(A,S,P) = \frac{\sum_{x=1}^{r}(1 - \sum_{s_i \in S} V_E(E,P,p_{s_i}) \times R_U(A,s_i,p_{s_i}))}{r} \tag{4}$$

where $V_E(E,P,p_{s_i})$ is the maximum visibility of the sensor s_i (located on polygon p_{s_i}) from the set of opponent vehicles E which are located on set of positions P. As in the sensor visibility, it is computed by,

$$V_E(E,P,p_{s_i}) = \max_{e_j \in E} \{V_E(e_j,p_k,p_{s_i})\} \tag{5}$$

The $V_E(e_j,p_k,p_{s_i})$ term in this equation is the visibility of a single point on the terrain (where the sensor s_i is located) from an opponent sensor located on polygon p_k. Here, this term is the dual of the term $V^P_S(s_j,p_k,b)$; therefore it is also computed with the Equation 3.

The $R_U(A, s_i, p_{s_i})$ term is the ratio of the utility of visibility of the given sensor s_i to the cumulative utility of visibility of all sensors, as shown in the following equation.

$$R_U(A, s_i, p_{s_i}) = \frac{U_{vis}(A, s_i, p_{s_i})}{\sum_{s_j \in S} U_{vis}(A, s_j, p_{s_j})} \tag{6}$$

It should be noted that the denominator in the previous equation is not the overall utility of visibility, but it is the cumulative utility of visibility. It is due to the fact that overall utility of visibility may be even equal to utility of visibility of a single sensor, which may generate a negative value in Equation 4. If a sensor with a high utility of visibility value (and therefore a high utility ratio) is completely seen by an opponent object, this will significantly decrease the utility of stealth.

The locational attributes (polygon number, heading and tilt angles, sensor types) of $r * |E|$ opponent sensors are set at the beginning of the program execution, where $|E|$ is the set of opponents considered at each scenario.

C. Computing the Utility of Cost

In our study, the term *cost* of a given sensor s_i includes two separate meanings a) the (normalized) financial cost of sensor, $NCost_F(s_i)$, and b) the (normalized) placement cost of sensor to its current location (i,e., polygon k), which is represented by $NCost_L(s_i, p_k)$. Based on these terms, the utility of cost for a set of sensors S placed on a set of locations P is formally defined by

$$U_{cost}(S, P) = 1 - \{\frac{\sum_{S_i \in S}(\omega_F \times NCost_F(s_i))}{|S|} + \frac{\sum_{S_i \in S}(\omega_L \times NCost_L(s_i, p_k))}{|S|}\} \times N|S|. \tag{7}$$

It should be noted that financial and placement (locational) costs are independent, and ω_F and ω_L are the weights (from the range $[0..1]$) of the two cost terms, respectively. In our experiments, $\omega_F = 0.7$ and $\omega_L = 0.3$, unless otherwise specified. The placement cost term in the right side of this equation is computed by $NCost_L(s_i, p_k) = P_L(s_i, p_k) \times \varphi_{s_i}$ where $P_L(s_i, p_k)$ is the locating probability of sensor s_i on polygon p_k; and φ_{s_i} is a sensor specific constant that is set to 1, unless otherwise specified. The locating probability values of all terrain polygons are set by considering the heights and slopes of polygons and characteristics of the sensors (i.e., sensor carriers). In order to simplify the model, this term can be set with the weights of polygons based on the second method explained in Section II-A. In Equation 7, $N|S|$ is the normalized value of sensors usage, which is computed by

$$N|S| = \frac{|S|}{E(S)}, \tag{8}$$

where $|S|$ is the number of sensors used in the solution, and $E(S)$ is the expected value of the number of sensors for the given terrain, which is equal to the mean value of upper and lower limits of sensor usage.

III. HEA-Based Sensor Placement

In order to build hybrid evolutionary algorithms (HEA), there are various methods to incorporate specialized operators and domain specific knowledge with evolutionary algorithms. We consider problem-specific heuristics for the initial population generation. There are novel intelligent variation operators presented in our study, such as Contribution-Based Crossover and Proximity-Based Crossover operators that incorporate problem specific knowledge. Additionally, a local search phase is applied on the output of the variation operators.

In our GA-based approach, each solution contains the type and the locational attributes (position, heading angle, tilt angle) of sensors. There is no restriction on sensor quantity; therefore string representation supports variable-length chromosomes. Additionally, this work has no restriction on the order of sensors in a given solution. A steady-state Genetic Algorithm is applied which generates one individual at each iteration. We consider tournament selection mechanism and the tournament size is varied in the experiments. The value of the fitness function is set with the total utility value.

A. Initial Population Generation

The first phase to generate a solution for initial population is to determine the number of sensors considered in the solution, which is set randomly between the upper and the lower limit of the sensor quantity. The sensor mode (detection, recognition, identification) and the horizontal depth of view (Δ) on the given sensor mode are considered in order to calculate the limits. After the sensor quantity of a solution is determined, the type of each sensor is set randomly by preserving the inverse proportionality to sensor capacities (i.e., the view ranges with respect to given sensor mode). Then, locations of sensors in each solution are determined with a heuristic which aims to distribute the selected sensors evenly on the given terrain.

B. Crossover Operation

Our GA-based approach includes a set of variation operators (crossover and mutation operators) and a local search phase that is applied in between crossover and mutation operators. In our study we consider three different crossover operators, which are Contribution-Based Crossover (CBX), Proximity-Based Crossover (PBX) and Cut and Splice Crossover (CSX). It should be note that CBX and PBX are the ones that consider problem-specific knowledge. All of the three crossover operators generate a single offspring as the output.

1) Contribution Based Crossover (CBX) Operator: The main idea behind this operator is to carry a sensor to the offspring from one of the parents based on its contribution, which is expressed in terms of utility of the sensor. The sensors in both parents are examined in this operator. One sensor is selected randomly from each parent and the better one, which has higher total utility value than the other sensor, is moved into the offspring as the first sensor of the offspring. Then, at each step, one of the remaining sensors from each parent is selected and the best one of selected sensors is moved to the offspring if the sensor keeps acceptable proximity distance with all sensors that are already in the offspring. The calculation steps of acceptable proximity distance is explained in the following part.

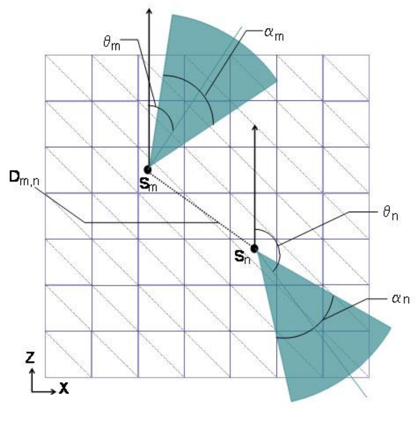

Fig. 1. Contribution Based Crossover.

Assume that the sensor s_m is the sensor which has the highest total utility value (highest contribution value) among the two sensors that are selected from the remaining sensors of each parent solution; and assume that s_m is already located on polygon m with the heading angle θ_m as given in Figure 1. Assume that s_n is one of the sensors which is already copied to offspring; and s_n is located on polygon n with the heading angle θ_n. Assume that (horizontal) viewing angles of s_m and s_n are α^m and α^n, respectively. In order to decide whether sensor s_m located on polygon m will be accepted for the offspring, the following two inequalities are considered.

$$D(n,m) \quad > \quad (\Delta_n \, + \, \Delta_m) \times \mu \tag{9}$$

$$|\theta^m - \theta^n| \quad > \quad \frac{\alpha^m \, + \, \alpha^n}{2} \times \kappa \tag{10}$$

The right side of the first inequality is the acceptable proximity distance. The distance between sensors should be equal or greater than the acceptable proximity distance, which is computed by multiplying the summation of depth of views with a problem specific constant, μ. The depth of view value of each sensor is based on the sensor mode considered in the experiments.

If s_m validates the first inequality, it is selected for the offspring; then, the second inequality is not considered any more. Similarly, if it fails in the first inequality but it validates the second one, the sensor is still moved to the offspring. If the angular distance between heading angles of sensors (the left side in the second inequality) is greater than the summation of half of horizontal viewing angles of two sensors, the candidate sensor is copied since they are looking at different directions even if the distance between them is less than the acceptable proximity distance. The term κ, an angular constant related with overlapping ratio of horizontal viewing angles of sensors, is set to 0.8 unless otherwise specified.

If it does not validate both of the inequalities, the candidate sensor is dropped and sensor with the next highest total utility value is considered. This process is repeated as long as the number of sensors ($|S|$) moved to the offspring is less than the upper limit, which is computed by using the following inequality.

$$|S| \; \leq \; (|S_1^P| \, + \, |S_2^P|) \, * \, \epsilon \tag{11}$$

In this inequality, $|S_1^P|$ and $|S_2^P|$ are the number of sensors allocated in the first and the second parents of the offspring, respectively; and ϵ is a constant which is less than 1.

2) Proximity Based Crossover (PBX) Operator: In this operator, firstly, a crossover point k is selected randomly by considering $1 \leq k \leq m-1$ where m is the minimum of the sensor quantities of the two parent solutions. Starting from the first sensor, k sensors of the first parent are copied to the offspring. The next phase is to copy sensors from the second parent. Starting from the first element of the second parent, sensors are copied to the offspring in the order by considering the inequalities given in Equation 10. The first equation is for testing the proximity distance; and if the current sensor of the second parent passes this test for all sensors already in the offspring, it is copied to the offspring. Otherwise the second test related with viewing angle is applied.

3) Cut and Splice (CSX) Operator: This operator is similar to the original single-point crossover operator proposed in the literature. As in the previous operator, a crossover point is selected randomly from the range $[1..m-1]$ where m is the minimum of number of sensors exist in two parents. Then, both parents are separated at the given crossover point, and two children are created by exchanging the tails; and the better one is selected as offspring. The tests of proximity distance and viewing angle are not applied in this operator, and there is no upper bound for number of sensors considered.

C. Local Search Phase

After a solution is generated using the crossover operator, the local search phase targets on improving the quality of the solution by modifying angular attributes of sensors. At each iteration of the local search the sensor with the minimum utility of visibility value is selected.

The heading angle is updated by adding a predefined increment amount (δ_H) repeatedly until the first improvement on utility of visibility for the given sensor. If there are large number of polygons on the terrain, δ_H can be low; otherwise a higher value can be set. The next phase is to modify the tilt angle of the sensor in order to improve sensor's visibility. An increment amount (δ_T) is added to the tilt angle of the given sensor repeatedly. This process is repeated until all possible alternatives covered.

This process is repeated until one of the following two conditions occurs: a) the cumulative improvement ratio on the utility of visibility of the modified sensors is greater than or equal to 20%, b) at a least 10% of sensors in the original solution is considered as part of the local search process. Only local improvements of sensors are considered and number of sensors to be examined is limited in order to bound the running time of this phase.

D. Mutation Operators

Mutation operators in our study can be considered in two categories, which are the locational mutation operators, and the angular mutation operators.

1) Locational Mutation Operators: There are three types of mutation operators in this category which deal with the locations of sensors: the *update*, the *delete* and the *insert* operators. The other parameters of our experimentation is set with the default values.
The first operator updates the location of a sensor in two steps: i) selecting the sensor, and ii) selecting the new location. The sensor for the update operator is selected either randomly or it is the one which has the minimum utility of visibility value. On the other hand, the new location is selected randomly either from the whole terrain, or from the same region where the original sensor is located on. The second operator deletes one of the sensors, which can be either a randomly selected sensor or the one which has the minimum utility of visibility value. The last operator in this category adds a new sensor to the solution on a polygon that is selected randomly either from the whole terrain or from the region which already includes the sensor with the minimum utility of visibility value. The sensor type and the heading and the tilt angles of the new sensor is set randomly by considering the feasibility ranges of angles.

2) Angular Mutation Operator: This category is to update the angular properties of a sensor, which is either selected randomly or the sensor that minimizes the utility of visibility value. There are two operators in this category, which are *mutation on the heading angle* and *mutation on the tilt angle*. The former one updates the heading angle of the sensor randomly, and the latter one updates the tilt angle randomly from a predefined feasible range. One extension is to select one of the two mutation operators from this class non-uniformly. Specifically, mutation on the heading angle can be selected with a higher probability than mutation on the tilt angle.

In addition to these two classes of mutations, updating the sensor type is also considered as a mutation operator. In our experimental study, type of a randomly selected sensor is updated randomly at every 100 generations; if the update causes an improvement, then it is accepted; otherwise it is rejected. The update on the sensors type is applied for both of the mutation classes given above.

IV. Experimental Study

In this section, we present the results of experiments that evaluate the effectiveness of our algorithm. The experiments in this study were performed on a cluster of PCs, each of which has Intel Xeon 2.33GHz processor running Linux operating system. The default values of general parameters in our hybrid evolutionary algorithm listed in Table I are considered.

TABLE I
Default Settings of Selected Parameters of Experimentation

Parameter	Value
-Sensor Allocation Probability in a Terrain Region (center, random)	(20%, 80%)
-Coefficient for Depth of View in Crossover Operators	$\mu = 0.4$
-Coefficient for Sensor Quantity Inherited from Parents to Offspring	$\epsilon = 0.5$
-Improvement Ratio in Local Search Phase	$I_R = 20\%$
-Sensor Ratio in Local Search Phase	$SR = 10\%$
-Number of Enemies (detection, recognition and identification modes)	(10, 40, 100)
-Tournament Size	5
-Number of Generations for Termination	1000

The first set of experiments is for identifying the values of GA-parameters, which are the population size, the type of the crossover operator, the type of the mutation operator and the sensor selection criteria in mutation. The population size is assigned from the set $\{30, 50, 100\}$. Three crossover operators and two categories of mutation operators given in Section III are considered in the experiments. Additionally, the source sensors for mutation is selected either randomly or the one which has minimum visibility. The combination of those parameters constructs 54 different cases, each of which is run with 30 replications. Therefore, a total of 1620 tests are conducted in the first set of experiments. On the other hand, the other parameters of our experimentation is set with the default values. Specifically, initial population of our algorithm is set randomly without any heuristic. The increment amount of heading angle (δ_H) is set to 25^o, which is one-fourth of the minimum horizontal viewing angle of sensors considered. The increment amount of the tilt angle is set to $\delta_T = 3^o$. A visibility-dominant mission by selecting recognition mode of sensors with a rough terrain is considered.

The optimization results are obtained by setting the goals for the total utility value and generating the optimal conditions. When the results of experiments are examined, the population size is set to 50, the contribution-based crossover is selected, and the sensors with the minimum visibility value is considered for the mutation operator. However, the mutation operator does not have a significant effect on total utility, when a visibility-dominant mission is performed.

The second set of experiments evaluates the effects of the following parameters on the performance, which are the strategy for selecting the initial population, increment values for the heading and the tilt angles, the mission-specific utility weights and the terrain type. Initial population is set either randomly or based on the heuristic presented in Section III. The increment value in heading and tilt angles are assigned from the sets $\{25, 50, 75\}$ and $\{3, 5, 7, 10\}$, respectively. Three different missions (visibility-dominant, stealth-dominant and cost-dominant cases) and two terrain types (smooth and rough terrains) are taken into account in the experiments. The coefficients $(w_{vis}, w_{st}, w_{cost})$ given in Equation 1 are set with $\{0.6, 0.3, 0.1\}$, $\{0.3, 0.6, 0.1\}$, and $\{0.25, 0.25, 0.5\}$ for visibility-dominant, stealth-dominant and cost-dominant missions, respectively. Based on the analysis of variance, it is found that the first three parameters do not significantly affect the total utility values. Both the terrain type and the mission type affect the quality of solutions as expected.

A. Performance Evaluation and Discussion

In this part, we present the effectiveness of our algorithm for detection, recognition and identification modes of sensors on both smooth and rough terrains. Since sensors have limited range for all modes, a steep slope on the view cone of a sensor decreases the visibility range of the sensor. Therefore, the total utility value observed on a smooth terrain is higher than the total utility value on a rough terrain, when comparable number of sensors are utilized on both terrains (see Tables II & III). Three different sensor modes on both the visibility-dominant (M1) and the stealth-dominant (M2) missions are considered in these tables.

TABLE II
PERFORMANCE RESULTS ON A SMOOTH TERRAIN WITH DIFFERENT SENSOR MODES

Sensor Mode	Mission Type	Algorithm	Number of Sensors	Total Utility
Detection	M1	RS	13.9	0.5267
		HEA	14.7	0.7419
	M2	RS	9.7	0.6894
		HEA	14.7	0.9216
Recognition	M1	RS	80.7	0.5524
		HEA	59.7	0.7061
	M2	RS	54.9	0.6465
		HEA	62.3	0.8731
Identification	M1	RS	227.7	0.5138
		HEA	178.2	0.6474
	M2	RS	162.7	0.6620
		HEA	170.3	0.8409

The performance comparison of the HEA with a random search method (RS) is presented for all sensor modes and mission types pairs in Tables II and III. The best individual in a randomly generated initial population of the HEA with 50 individuals is the output of the RS method for each test. The individual utility values and the total utility of the solutions generated by the HEA are significantly outperforms those of the RS method for both smooth and rough terrains. Additionally, the required number of sensors to cover a 3-D terrain varies according to the selected sensor mode. Since, the behavioral attributes of sensors including depth of view, horizontal and vertical angles for the detection mode are better than those values for the other modes, fewer sensors are required to cover in the detection mode for both smooth and rough terrains.

TABLE III
PERFORMANCE RESULTS ON A ROUGH TERRAIN WITH DIFFERENT SENSOR MODES

Sensor Mode	Mission Type	Algorithm	Number of Sensors	Total Utility
Detection	M1	RS	11.3	0.4645
		HEA	16.3	0.6545
	M2	RS	8.0	0.6516
		HEA	16.3	0.8561
Recognition	M1	RS	77.3	0.4975
		HEA	63.0	0.6638
	M2	RS	44.5	0.6321
		HEA	61.3	0.8385
Identification	M1	RS	214.3	0.4777
		HEA	180.6	0.6121
	M2	RS	141.9	0.6564
		HEA	166.5	0.8148

V. CONCLUSIONS

Positioning and utilizing multiple sensors for acquisition of a given environment is one of the fundamental research topics in various domains including military operations, computer vision and robotics. The contributions of this paper can grouped in two-folds. Firstly, we present a novel multi-attribute utility-based framework for deploying and configuring multiple sensors in a 3D terrain that combines three conflicting objectives into a unified total utility function, which are maximizing the coverage area while maintaining the maximum stealth and minimizing the total acquisition cost of deploying sensors. Secondly, this paper presents a new hybrid evolutionary algorithm (HEA) based solution for the constructed framework. The computational study clearly points out the effectiveness and robustness of our HEA-based solution under various values of several experimental parameters.

ACKNOWLEDGMENTS

This research was supported by The Scientific and Technological Research Council of Turkey (TUBITAK) with a research grant (no. 106E159). Additionally, parts of the computations have been carried out by using UYBHM at ITU through a grant (20432008).

REFERENCES

[1] E. Riseman, H. Schultz, W. Lim, B. Draper, and A. Hanson, "Deadalus battlefield visualization system," in *Proc. ARPA Image Understanding)*, 1996.
[2] Z. Michalewicz and D. B. Fogel, *How to Solve It: Modern Heuristics*. Springer-Verlag, 2000.
[3] B. R. Fox and M. B. McMahon, "Genetic operators for sequencing problems," in *Proceedings of the First Workshop on Foundations of Genetic Algorithms and Classifier Systems*, G. Rawlins, Ed. Morgan Kaufmann, 1990.
[4] J. J. Grefenstette, "Incorporating problem specific knowledge into genetic algorithm," in *Genetic Algorithms and Simulated Annealing*, L. Davis, Ed. Morgan Kaufmann Publishers, 1987.
[5] G. Raidl, "Hybrid evolutionary algorithms for combinatorial algorithms," Habilitation Thesis, Vienna University of Technology, Vienna, Austria, 2002.
[6] P. Moscato. Memetic algorithms' home page. [Online]. Available: www.ing.unlp.edu.ar/cetad/mos/memetic_home.html
[7] R. Keeney and H. Raiffa, *Decision with Multiple Objectives: Preferences and Value Tradeoffs*. New York: John Wiley and Sons, 1993.
[8] J. Dyer, "Remarks on the analytic hierarchy process," *Management Science*, vol. 36, no. 3, pp. 249–259, Mar. 1990.
[9] M. D. Proctor and W.J.Gerber, "Line-of-sight attributes for a generalized application program interface," *the Journal of Defense Modeling and Simulation*, vol. 1, no. 1, pp. 43–57, 2004.
[10] D. Handerson, "Modterrain: A proposed standart for terrain representation in entity level simulation," Master's Thesis, Naval Postgraduate School, California, 1999.