SAS 2008 – IEEE Sensors Applications Symposium
Atlanta, GA, February 12-14, 2008

# An Extensible Model for the Deployment of Non-Isotropic Sensors

Brian Carter and Rammohan Ragade
Computer Engineering and Computer Science
University of Louisville
Louisville, KY

***Abstract*—Existing approaches for determining the optimal deployment positions of sensors suffer from a number of critical drawbacks. First, homogeneous deployment models have been commonly assumed, but in practice deployments of heterogeneous sensors are typical. Second, existing approaches assume isotropic sensing ranges but it has been found that hardware and environmental conditions cause imperfections in sensing. Third, existing models are very application-dependent. We propose an extensible modeling framework for the problem of determining optimal deployment positions for a set of heterogeneous, non-isotropic sensors to cover a set of points in an area. The problem is formulated using a genetic algorithm where the objective is to minimize the cost to cover all points. Our technique is to decouple the coverage determination method from the sensor deployment model. This allows the sensor deployment model to remain consistent and address the critical drawbacks of previous models. A homeland security application is presented to illustrate the capabilities of our approach.**

## I. Introduction

Recent advances in sensor technologies have expanded their application in many practical applications for civil and military purposes. Commercially available, off-the-shelf sensors range in complexity from simple acoustic sensors to sophisticated imaging equipment [1]. Many applications require a collection of heterogeneous sensors with varying sensing abilities. The type of sensors deployed and their deployment positions will affect the cost and effectiveness of the sensor network. This complexity is compounded with the inclusion of non-isotropic sensors, sensors which have coverage properties that are dependent on their deployment direction.

Given an area to monitor, determining the set of sensors to deploy is a difficult problem. This problem is often termed as the sensor deployment problem (SDP) and has been shown to be NP-complete [2]. In previous research, the SDP has been studied based on simplistic models which determine the minimal number of isotropic sensors to cover a set of points [3], [4]. Research has proposed advanced techniques for non-isotropic sensor deployment and field-of-view (FOV) problems [5], [6]. The studies present different coverage determination methods and models for determining the optimal placement of sensors.

In this paper, we study the SDP using non-isotropic sensors where the goal is to minimize the cost to cover all target points. A genetic algorithm (GA) is applied to optimize the problem. Our contribution is the decoupling of the coverage determination method from the deployment model. This allows the model to remain consistent when applied to SDPs with varying complexity.

In the homeland security domain, sensor deployment problems may include open area deployments, border monitoring, or complex city surveillance. Each scenario has a set of problem specific parameters: location of points to be covered, location of points where sensors may be deployed, sensing abilities of the available sensors, and in which direction or orientation a given type of sensor may be deployed. As shown in the following sections, our proposed model is defined to handle very diverse deployment problems without developing a problem specific model for each situation. This is accomplished by defining a coverage matrix, which is problem specific, as an input into the model.

The rest of the paper is organized as follows: In section 2, the optimization problem is formulated. In section 3, a model is defined and a genetic algorithm approach is given in section 4. In section 5, the case study results are documented. Finally in section 6, some conclusions are drawn.

## II. Problem Formulation

The SDP is the problem of covering the rows of a $m$-row, $n$-column, zero-one coverage matrix $(a_{ij})$ by a subset of the columns at minimal cost. Defining $x_j = 1$ if column $j$ is used in any rows of the solution and $x_j = 0$ otherwise. The cost of deploying column $j$ is defined as $c_j$. The SDP can be formulated as an optimization problem:

$$Minimise \qquad \sum_{j=1}^{n} c_j x_j \tag{1}$$

$$\text{Subject to} \qquad \sum_{j=1}^{n} a_{ij} x_j \geq 1, \quad \text{i=1,...,m} \tag{2}$$

$$x_j \in \{0,1\}, \quad \text{j=1,...,n} \tag{3}$$

Equation (1) defines the optimization problem where the objective is to minimize cost. The solution is constrained in equation (2), which requires that each row is covered by at least one column. Equation (3) is the integrality constraint.

To solve the optimization problem, a number of algorithms have been presented in literature [7], [8], [9]. It has been shown that optimal algorithms are not applicable due to the runtime required to find a solution [8]. Greedy heuristics have been proposed to lower the runtime for homogenous sensor deployments [4], [10]. It has been shown that evolutionary algorithms provide near-optimal results for heterogenous SDPs using GA and simulated annealing algorithms [2], [7], [8], [9].

## III. Problem Modeling

We propose a sensor deployment model (SDP-Model) which addresses the drawbacks of previous models. Any sensor network that has a longevity of more than a single deployment will involve the use of different sensing equipment. We have experienced this in our lab, where certain intrusion detection sensors have been discontinued. Given this new set, the collection of sensors have varying abilities. The previously proposed homogeneous models are no longer applicable [11]. Second, the ability of a sensor to sense an intrusion at a given point is dependent on the hardware, the deployment position, and the deployment orientation of the sensor. Also, objects in the target area may block or limit the sensing ability. The previously perfect binary detection circle (isotropic) models are also not applicable [3], [4]. Third, previously proposed models are application-dependent to address a specific problem [5], [6]. The method used to determine if a point is covered is tightly coupled with the deployment model. With so many techniques proposed for determining coverage, a decoupled method is required.

These drawbacks are addressed in the SDP-Model. In our research, we found that the method used to determine coverage is application-dependent. Determining if a point is covered by a given sensor is dependent on the hardware, the deployment point, and the orientation. These elements are decoupled from the SDP-Model by defining a coverage matrix.

Using the coverage matrix $a_{ij}$, we define each row $i$ as a three-dimensional grid point $< x, y, z >$ which must be covered by at least one sensor. Each column $j$ is defined by a deployment-tuple $d_j$:

$$< sensor\ type,\ deployment\ point,\ orientation >$$

Given $a_{i,j} = 1$, indicates target point $t_i$ (row) is covered by deployment-tuple $d_j$. Where $d_j$ defines the inclusion of a specific sensor type which is deployed at a given point with a given orientation.

To establish the coverage matrix, the first step is to define the points to be covered. A row is added to the matrix corresponding to each target point. The next step is to define the deployment-tuples. Given the available combinations of $< sensor\ type,\ deployment\ point,\ orientation >$, the candidate deployment-tuples are defined. A column is added for each deployment-tuple. The final step is to determine the set of points covered by each deployment-tuple using a coverage determination method. Without loss of generality, we shall assume that the coverage determination method is defined using a pre-established approach [4], [5], [8], [6].

The coverage determination method used is specific for each problem. For example, a simplistic SDP may use the Euclidean distance for coverage determination where a complex SDP may use a FOV metric. Independent of the method used, the coverage matrix $a_{ij}$ is formulated as shown in Figure 1. The target points $t_i$ require coverage by selecting the deployment-tuples $d_j$. In the example, deployment-tuple $d_1$ covers points $\{t_1,\ t_2\}$.

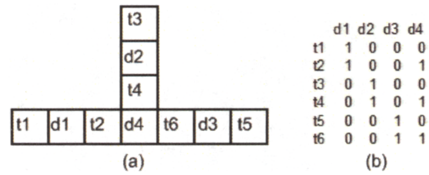


Fig. 1. (a) Visual illustration of a matrix using six points and four deployment-tuples; (b) corresponding matrix $a_{ij}$.

Our approach for defining the coverage matrix $a_{ij}$ has many benefits. The model remains consistent for simple and complex problems by decoupling the model from the coverage determination method. The only task required to optimize a SDP is to provide the coverage matrix and algorithm parameters to the model.

The model implements a genetic algorithm to optimize the problem as shown in figure 2. The output of the model is a subset of deployment-tuples which cover all target points at a minimal cost. As shown in the next section, we devise a suitable representation scheme of the coverage matrix $a_{ij}$ for the GA approach.

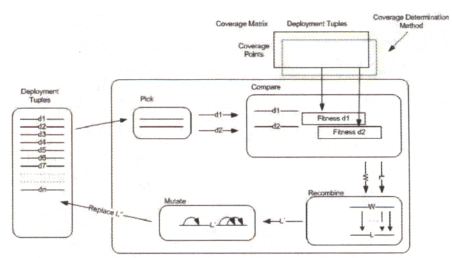


Fig. 2. Sensor Deployment Model (SDP-Model).

## IV. Genetic Algorithm Approach

A genetic algorithm is often defined as an "intelligent" probabilistic search algorithm [7]. Holland [12] originally established the theoretical foundations using an evolutionary process. The principles of natural selection and "survival of the fittest" are used to reproduce children which are more "fit" for their environment. It has been shown that genetic algorithms can be applied to a variety of combinatorial optimization problems [13].

With the success of GAs, many adaptations have been presented. For this paper, the Microbial Genetic Algorithm [14] is applied by defining the following five components [13]:

- a genetic representation for potential solutions to the problem,

- a way to create an initial population of potential solutions,
- an evaluation function that rates solutions in terms of their ”fitness”,
- genetic operators (crossover, mutation) that alter the composition of children,
- values for various parameters used in the algorithm.

First, the chromosome is defined. Reviewing Figure 1, the task is to select a subset of the deployment-tuples which cover all points subject to minimizing the cost. The chromosome can be represented as a binary vector $v_i$. The vector $v_1$ = (1110) would represent the use of deployment-tuples $\{d_1, d_2, d_3\}$. A ”0” in the vector indicates the exclusion of $d_4$. The number of points covered $coverage$ is 6. The length of the vector is always the number of deployment-tuples $n$.

The next task is to determine the initial population $P$. A population is a set of potential solutions (or individuals, chromosomes, vectors). The number of vectors included in the population is determined by the population size parameter $POP$. All bits in the vectors are initialized randomly.

When applying any GA, the difficult task is to define the fitness function. The evaluation function $evaluate$ for binary vectors $v$ is equivalent to the fitness function $f$: $evaluate(v) = f(x)$. We defined $evaluate$ as the summation of cost for all sensors included in the solution plus a weighted penalty for not covering cells (Equation 4). The value $(m - coverage)$ represents the number of target points not covered. The variable $w$ represents the weight of the penalty. Using a weighted penalty allows us to relax Constraint (2) in our fitness function to allow better fitting vectors to evolve. In our fitness function, a lower fitness value indicates a better fit.

$$f(x) = \sum_{j=1}^{n} c_j x_j + w(m - coverage) \quad (4)$$

For the genetic operators (crossover, mutation), the methods defined by Harvey [14] are used. For the SDP, the following parameters are defined:

- coverage matrix $a_{ij}$,
- number of tournaments $TOURS$,
- population size $POP$,
- probability of crossover $p_c$,
- probability of mutation $p_m$.

The GA is adapted for the SDP as shown in Algorithm 1. The function $rnd$ returns a random number between 0.0 and 1.0 and $rnd(1..POP)$ returns an integer between 1 and $POP$. Lines 4-5 of the algorithm perform the selection of two random vectors to evaluate. Their evaluation is performed in Lines 6-10, using the fitness function defined in equation (4). The crossover is performed in lines 11-13. The mutation of the losing vector $v_l$ is performed in lines 14-16. The algorithm completes after a number of iterations ($TOURS$). The final output of the algorithm is the best solution vector $v_{max}$ which has the best fit and satisfies all the constraints.

**Algorithm 1** GA for Non-Isotropic SDPs

```
1:  input: a_ij
2:  initialize P = {v_i : i = 1..POP}
3:  for i = 1 to TOURS do
4:     a=rnd(1..POP)
5:     do b=rnd(1..POP) while (a==b)
6:     if evaluate(v_a) > evaluate(v_b) then
7:        w = a, l = b
8:     else
9:        w = b, l = a;
10:    end if
11:    for j = 1 to n do
12:       if(rnd < p_c) v_l[j] = v_w[j]
13:    end for
14:    for j = 1 to n do
15:       if(rnd < p_m) v_l[j] = 1 - v_l[j]
16:    end for
17: end for
18: output: v_max
```

## V. Case Study: Homeland Security Sensor Fence

In this section, a simplified SDP is presented to allow the reader to evaluate our approach. A two dimensional area with 6 points to be covered and 5 deployment points is defined as shown in Figure 3. The six triangles are defined as the points that require coverage. The 5 circles represent the points where sensors may be deployed. The figure may represent a river or road, where sensors can be deployed only on the outer edges of the target area. By defining the SDP-Model with coverage points and deployment points, such a realistic case study can be evaluated.

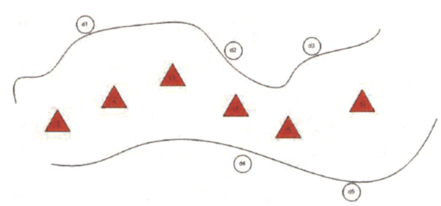

Fig. 3. Homeland Security Sensor Fence.

There are two types of sensor available:

- $s_1$: a short range image sensor which has a cost of \$100 with an non-isotropic sensing range of 30 meters and 45 degree field of view.
- $s_2$: a medium range acoustic sensor which has a cost of \$150 with an non-isotropic sensing range of 50 meters and 90 degree field of view.

The problem is to select a subset of the deployment-tuples which minimize cost subject to covering all points. Without loss of generality, we shall assume that the coverage determination method is defined using a pre-established field of view approach [5], [6]. To minimize the number of deployment-tuples to display, it is assumed that the sensors are fixed and have only one orientation. With 5 deployment locations and 2 sensors each with 1 deployment orientation, resulting in the following 10 deployment-tuples:

- $d1 =< s_1, d_1, S >$,
- $d2 =< s_2, d_1, S >$,
- $d3 =< s_1, d_2, S >$,
- $d4 =< s_2, d_2, S >$,
- $d5 =< s_1, d_3, S >$,
- $d6 =< s_2, d_3, S >$,
- $d7 =< s_1, d_4, N >$,
- $d8 =< s_2, d_4, N >$,
- $d9 =< s_1, d_5, N >$,
- $d10 =< s_2, d_5, N >$,

Where $s_1$ indicates the inclusion of sensor type 1 and the $S$ indicates a south facing orientation ($N$ for north). The deployment points are indicated by the $d_i$ deployment location. Using the field of view for the coverage determination method, the matrix $a_{ij}$ can be generated shown in Table I.

TABLE I
CASE STUDY COVERAGE MATRIX

| CP | d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 | d9 | d10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 6 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |

For this problem, the optimal solution derived from the algorithm can be confirmed deterministically by generating all the possible permutations. We used the permutations to check the results from our simulation program which implements the SDP-Model. There are two chromosomes which satisfy the constraints: (0100000001) and (0100010000). Calculating their fitness (Function 4), both solutions cover all points at a cost of 300.

## VI. CONCLUSION

A modeling framework for the problem of deploying a set of non-isotropic sensors in an area is presented. In this model, the problem is formulated as an optimization problem with the objective to minimize cost while covering all points. A coverage matrix is defined to decouple the coverage determination method from the model. This allows the framework to be used for a wide range of SDPs. A GA approach is presented to solve the optimization problem. A homeland security sensor fence case study is given to illustrate the capabilities of the framework.

We defined the model in an attempt to address the diverse complexities found in the homeland security domain. In the case of an event, efficient and effective deployments of the available sensing and surveillance equipment must be determined. This evaluation may be recalculated at regular intervals based on failures or changes in the event.

Such NP-complete problems require a flexible deployment model which can be applied and adjusted based on parameters. Past problem specific models would require the development of new models, which may not be possible due to time constraints or availability of resources.

The model presented in this paper address the diverse complexities through the use of parameters. Most notable is the definition of a coverage matrix. The matrix provides the flexibility to include any combination of sensor types, deployment points, and orientations in the model. The model determines a sensor deployment where costs are minimized and all points are covered.

The case studies demonstrate very promising results. We will continue to improve the framework by considering additional ways to optimize SDPs through the use of coverage metrics. Effects of varying the mutation, crossover, and other parameters based on coverage will be addressed in a future paper.

## REFERENCES


[1] D. J. Russomanno, C. R. Kothari, O. A. Thomas, Building a sensor ontology: A practical approach leveraging ISO and OGC models, in: H. R. Arabnia, R. Joshua (Eds.), Proceedings of the 2005 International Conference on Artificial Intelligence, ICAI 2005, Las Vegas, Nevada, USA, June 27-30, 2005, Volume 2, CSREA Press, 2005, pp. 637–643.

[2] Q. Wu, N. Rao, X. Du, S. Iyengar, V. Vaishnavi, On efficient deployment of sensors on planar grid, Elsevier's Computer Communications (2007) accepted, to appear.

[3] S. Meguerdichian, F. Koushanfar, M. Potkonjak, M. B. Srivastava, Coverage problems in wireless ad-hoc sensor networks, in: IEEE Infocom 2001, 2001, pp. 1380–1387.

[4] P. C. F.Y.S. Lin, A near-optimal sensor placement algorithm to achieve complete coverage-discrimination in sensor networks, in: Communications Letters, IEEE, Vol. 9, IEEE, 2005, pp. 43–45.

[5] J. Adriaens, S. Megerian, M. Potkonjak, Optimal worst-case coverage of directional field-of-view sensor networks, in: Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on, IEEE, 2006, pp. 336–345.

[6] J. Ai, A. A. Abouzeid, Coverage by directional sensors in randomly deployed wireless sensor networks, J. Comb. Optim 11 (1) (2006) 21–41.

[7] P. C. J. Beasley, A genetic algorithm for the set covering problem, European Journal of Operational Research 94 (1996) 392–404.

[8] R. Ramadan, H. El-Rewini, K. Abdelghany, Optimal and approximate approaches for deployment of heterogeneous sensing devices, EURASIP Journal on Wireless Communications and Networking 2007 (2007) Article ID 54731, 14 pages.

[9] S. J. Habib, Modeling and simulating coverage in sensor networks, Comput. Commun. 30 (5) (2007) 1029–1035.

[10] A. Efrat, S. Har-Peled, J. Mitchell, Approximation algorithms for two optimal location problems in sensor networks, Broadband Networks, 2005 2nd International Conference on 1 (5) (2005) 714 – 723.

[11] X. Shen, J. Chen, Y. Sun, Grid scan: A simple and effective approach for coverage issue in wireless sensor networks, Communications 8 (3) (2006) 3480–3484.

[12] J. H. Holland, Adaption in Natural and Artificial Systems, University of Michigan Press, 1975.

[13] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, 3rd Edition, Springer-Verlag, 1996.

[14] I. Harvey, The microbial genetic algorithm (1996).