Exact and Heuristic Approaches for Role Assignment Problem in Wireless Sensor Networks

Fernanda Sumika Hojo de Souza, *fersouza@dcc.ufmg.br*
Geraldo Robson Mateus, *mateus@dcc.ufmg.br*

Keywords: Role Assignment, WSN, Optimization Model, Heuristic.

1. Introduction

The wireless sensor networks (WSN) can be considered as a special type of mobile *ad-hoc* network able to monitor the physical environment through small sensors. Such networks became viable due to the convergence of micro-electro-mechanical systems technology, wireless communication and digital electronics [1]. These small devices present memory, energy and transmission constraints; so, they must be used in a large number and in an integrated form. In this way, they show satisfactory results in terms of quality, robustness, beyond their low cost and capability to operate in an independent form [2]. One of the main constraints in WSN is the severe limited energy due to its very compact form. Therefore, this characteristic is object of study of many works, as presented by Youssef-Massaad [12], Rajeswaran [10], Hu [8] and Zhao [13].

The WSN have many applications and some popular areas are military, home, healthy, environment, among others [1]. The expectation is that, soon, the WSN have a large presence, playing a wide variety of tasks. In this way, the study of WSN as well as algorithms that try to improve their working conditions has great importance and is a big challenge.

In this paper we propose solutions for the role assignment problem in WSN. In this problem, the objective is to define specific roles for each sensor node [6], such that the system lifetime is maximized, attending the monitoring requirements. Thus, a sensor node can play different sensing functions, such as temperature, humidity, pressure, microphone, camera, etc. This work presents an optimization model and a genetic algorithm for the role assignment problem in WSN. Both enclose the coverage problem, the role assignment and the routing problem. The optimization model is presented as a mixed-integer linear programming problem, that can be solved through optimization software available in the market. The genetic algorithm is a heuristic method, being another option for resolution.

The rest of the paper is organized as follows. Section 2 presents some related work. Section 3 describes the treated problem. In section 4 the optimization model for the problem is introduced while section 5 shows the genetic algorithm proposed. Computational results are presented and discussed in section 6, while the last section concludes the work.

2. Related work

Algorithms for generic role assignment are proposed in [6]. This paper shows that this assignment is efficient, robust and practically viable in WSN context. The research presented in [2] demonstrates that the role assignment technique is conceptually simple and extremely powerful since it allows some complex ways of data collection beyond limits through linear programs. In [4], an algorithm was developed to solve the problem of placing nodes in the monitoring area and role assignment such that the network lifetime is maximized, while guaranteeing that each point/region of interest is covered by at least a sensor node. The results demonstrated that the algorithm can offer significant improvements in the system lifetime, when compared to random placement and role assignment of the nodes. Considering that the role assignment problem is very generic, it has many variations. In this work, we treat a particular case of it, where the roles correspond to sensing functions. Besides, we solve the coverage and routing problem. Thus, this approach differs from the literature because it is a combination of three problems.

The role assignment can be static, dynamic or on-line. The static assignment consists of defining particular roles for the nodes when the system starts the monitoring process. In dynamic assignment, the role assigned to each node can be modified during the network lifetime depending on the events. Each new assignment can be defined in periods of time (pro-active). On-line assignment is similar to dynamics, the main difference is that new assignments occur anytime, stimulated by new events (reactive).

According to Frank [6], some classic problems have embeded instances of the role assignment problem: coverage, clustering and in-network data aggregation. Considering that the coverage problem (a particular case) is a NP-hard problem [11], the role assignment problem belongs to NP-hard class too. The Art Gallery problem [9], that can be stated as determining the minimum number of guards required to cover the interior of an art gallery, has a similarity with the coverage problem in WSN. Another well known problem related to it is the Facility Location problem. Here, a set of facilities should attend a set of demand points (which corresponds to cover an area in WSN with a set of sensor nodes). The aim is to locate the facilities in order to optimize the transportation and location costs [5].

3. Role Assignment Problem in WSN

The role assignment problem in WSN considered in this paper consists in determining specific roles for each sensor node, such that they assume different sensing functions, ensuring the coverage of the demand points and the information routing. Moreover, a static approach is considered, where the returned solution supply the problem requirements. A formal definition of the problem can be described as: Given a set of sensor nodes S, a set of sink nodes M, a sensing area A and a set of demand points D, with a set of roles P, necessary to cover A; the solution of the problem consists of determining the role of each sensor node guaranteeing the coverage of the demand points, as well as the existence of a route from each active sensor to one of the sink nodes, aiming to minimize the energy consumption.

4. Optimization Model

An optimization model for the role assignment problem is presented as follows. The model encloses coverage, role assignment and routing problems.

Given a directed graph $G = (N, A)$, where:
N: set of nodes
A: set of arcs between nodes that belongs to the communication range of the sensor nodes.

We have the following subsets of nodes and arcs:
N^s: sensor nodes.
N^m: sink nodes.
N^d: demand points.
A^c: arcs connecting sensor nodes to demand points.
A^s: arcs connecting sensor nodes to other sensor nodes.
A^m: arcs connecting sensor node to sink nodes.
$E_j(A^s)$: arcs $(i, j) \in A^s$ entering a sensor node $j \in N^s$.
$S_j(A^s)$: arcs $(j, k) \in A^s$ leaving a sensor node $j \in N^s$.

Given the matrix:
D: energy matrix where each d^p_{ij} represents the communication energy of a role p, between the sensor node i and the sensor or sink node j. This energy is calculated based on the distance between the nodes and the role type.

Let the parameters be:
c^p_l: energy to active and keep the sensor node l with role p.
M: penalty applied when a demand point is not covered.
T^1: number of demand points.
T^2: number of sensor nodes.
b_l: capacity of a sensor node l.

Let the set of roles be:
P: set of roles that a node can play.

Let the variables of the model be:
x^p_{lj}: variable with value 1 if the sensor node l covers the demand point j playing role p.
z^{lp}_{ij}: variable with value 1 if the arc $(ij) \in (A^s \cup A^m)$ with role p is in the path from node l to a sink node and 0 otherwise.
t^p_l: variable equals 1 if the sensor node l is activated playing role p and 0 otherwise.
h_j: variable indicating that a demand point j is not covered.

The proposed optimization model can be used to solve the role assignment problem in different kinds of networks, such as, homogeneous or heterogeneous, random or specific placement of nodes, etc. Thus, the model is adaptable to many applications. Solving the model, the optimal solution specifies the set of sensor nodes that should be activated and which is its role (variables t^p_l), assuring the area coverage (variables x^p_{lj})

and the routing path (variables z_{ij}^{lp}). Next, the mixed-integer linear programming model representing the role assignment problem, coverage and routing in WSN is presented.

$$Min \sum_{l \in N^s} \sum_{(i,j) \in A^s \cup A^m} \sum_{p \in P} d_{ij}^p \, z_{ij}^{lp} \; + \; \sum_{l \in N^s} \sum_{p \in P} c_l^p \, t_l^p \; + \; M \sum_{j \in N^d} h_j \tag{1}$$

$$s.t.$$

$$\sum_{(l,j) \in A^c} x_{lj}^p \; + \; h_j \; \geq \; 1 \qquad \forall \, j \in N^d, \; \forall \, p \in P \tag{2}$$

$$\sum_{j \in N^d} x_{lj}^p \; \leq \; T^1 t_l^p \qquad \forall \, l \in N^s, \; \forall \, p \in P \tag{3}$$

$$\sum_{j \in N^s \cup N^m} z_{ij}^{lp} \; \leq \; T^2 t_i^p \qquad \forall \, l \in N^s, \; \forall \, i \in N^s, \; \forall \, p \in P \tag{4}$$

$$\sum_{i \in N^s} z_{ij}^{lp} \; \leq \; T^2 t_j^p \qquad \forall \, l \in N^s, \; \forall \, j \in N^s, \; \forall \, p \in P \tag{5}$$

$$\sum_{(ij) \in E_j^s(A^s)} z_{ij}^{lp} \; - \sum_{(ik) \in S_j^s(A^s \cup A^m)} z_{jk}^{lp} \; = \; 0 \qquad \forall \, j \in (N^s - l), \; \forall \, l \in N^s, \; \forall \, p \in P \tag{6}$$

$$\sum_{(ij) \in E_j^s(A^s)} z_{ij}^{lp} \; - \sum_{(ik) \in S_j^s(A^s \cup A^m)} z_{jk}^{lp} \; = \; -t_l^p \qquad j \; = \; l, \; \forall \, l \in N^s, \; \forall \, p \in P \tag{7}$$

$$\sum_{k \in N^s} d_{lk}^p \, z_{lk}^{lp} \; + \sum_{j \in N^s} \sum_{i \in N^s} d_{il}^p \, z_{il}^{jp} \; + \sum_{l \in N^s} c_l^p \, t_l^p \; \leq \; b_l \qquad \forall \, l \in N^s, \; \forall \, p \in P \tag{8}$$

$$\sum_{p \in P} t_l^p \; \leq \; 1 \qquad \forall \, l \in N^s \tag{9}$$

$$0 \leq x \leq 1; \;\; z, \; t \; \in \{0, 1\}; \;\; h \geq 0 \tag{10}$$

The objective function (1) aims to minimize the sum of energy costs for data transmission between the nodes, energy costs for activate and keep a node sensing and the costs for not attending some demand point (penalty). The set of constraints (2) ensures that each demand point with role p is being covered by at least one sensor node playing role p, otherwise, it is not covered. The constraints (3) guarantee that if a sensor node is inactive, it should not be attending a demand point. The constraints (4) and (5) define that the information flow is possible only between active nodes. The constraints set (6) ensures the flow conservation between each active node and a sink node. In (7), the constraints indicate the routing between an origin node l and a sink node. The constraints (8) assure that the sensor's capacity is not violated, i.e., the node has enough energy to be activated. In (9), it is guaranteed that each sensor node plays only a role at each time. Finally, the constraints (10) define the coverage variables as continuous, between 0 and 1, the connectivity and activation variables as binary and the non coverage variables as non negative.

5. Genetic Algorithm

The Genetic Algorithms (GA) became popular known by the mid-1960's, when John Holland [7] proposed a programming technique based on evolution and natural selection idea to solve problems. The preference for using GA in this work due to two factors: it is possible to escape from local minimums in complex systems and they produce a great amount of solutions simultaneously. The GA have many components that characterize them as an analogy to the natural evolution process and hereditary succession. Thus, being applied to the role assignment problem, some adaptations were made, to adjust each of these components to the problem particularizations.

1. **Solution Representation:** The solution representation used for the problem consists of an integer array of length n (number of sensor nodes). Each position c_i, $i = 0...n$, admits integer values, indicating if a sensor node is active or not and, in case it

is active, which sensor role (temperature, pressure, video) is assigned to it. Additionally, a binary array of length m (number of demand points) is used to indicate if the demand points are covered or not and the routing paths are created by a heuristic procedure to find the shortest path for each active node.

2. **Generation of the initial population:** The initial population for the problem was constructed creating a solution set (arrays C_n). The solutions were generated by a greedy constructive method, where at each iteration a sensor node is selected to be activated and thus successively until the demand points have been covered or there is no node that does not violate the connectivity to be activated. The role to be assigned to the node is always that one having the greater number of discovered demand points. Thus, for such role, one of the nodes with the best cost/benefit relation is selected, i.e., nodes that consume low energy and cover much demand points. Moreover, it is necessary the existence of the node's connectivity.
3. **Fitness function:** The adopted fitness function was the proper objective function of the problem. Thus, an individual (solution) with lower value for the fitness function (energy consumption) has a greater aptitude level than a greater function one.
4. **Genetic operators:** In this work, two kinds of genetic operators were adopted: mutation and crossover.
   Mutation: consists of the random selection of two sensor nodes and assign them new roles. The new assignment is random, such that it can be chosen anyone of the roles and even though to inactivate the sensor. Such assignment does not guarantee the new solution's feasibility, i.e., demand points can be not covered and the connectivity for all sensor nodes cannot exist.
   Crossover: this operator constructs two offspring chromosomes combining an assignment subsequence from a parent p_i and an assignment subsequence from another parent p_j. The cut point of the parents's chromosomes is randomly chosen. The offsprings f_1 and f_2 inherit the subsequences before the cut point of p_i and p_j respectively and the subsequences after the cut point are inherit from p_1 for f_2 and from p_2 for f_1. As well as in the mutation process, impracticable solutions can be generated.
5. **Selection process:** At each iteration, half of the population is selected to survive in the next generation. The selection is carried through a fitness-based process where fitter solutions are more likely to be selected.
6. **Parameters definition:** Normally, it is adopted the empirical parameter tuning, using common-sense and the problem previous knowledge. Thus, we used the following values: 50 individuals (population's size), 5% (mutation rate), 90% (crossover rate) and maximum number of generations without improvement varying between 20 and 50, depending on the problem size.
7. **GA's pseudo-code:** Algorithm 1 presents the GA's pseudo-code applied to the Role Assignment problem in WSN. The algorithm receives as input data the network whose problem must be solved, the size of the initial population, the mutation probability, the crossover probability and the maximum number of generations without improvement.

Algorithm 1 GA's algorithm

```
1: procedure GA(network, popSize, crossProb, mutProb, maxNumberGen)
2:     for i ← 1 to popSize do                                   ▷ generation of the initial population
3:         P.add(createGreedySolution(.));
4:     end for
5:     bestSol ← evaluatePopulation(.);                          ▷ population's evaluation
6:     while t < maxNumberGen do
7:         t++;
8:         applyCrossover(crossProb);                            ▷ crossover process
9:         applyMutation(mutProb);                               ▷ mutation process
10:        bestSol ← evaluatePopulation(.);                      ▷ population's evaluation
11:        if changed(bestSol) then
12:            t ← 0;
13:        end if
14:        P ← selectPopulation(.);                              ▷ selection process
15:    end while
16:    return bestSol;
17: end procedure
```

The initial population is generated through a successive construction of greedy and random solutions for the problem until the population size established has been reached (lines 2 and 3). An evaluation is performed to find the best solution until this moment, such that it is kept in $bestSol$ variable (line 5). After that, the evolution process starts, respecting the stop criterion determined by the maximum number of generations without improvement (line 6). The variable t stores the current number of generations, being incremented at each iteration (line 7) and coming back to zero whenever an improvement is reached. In line 8, the new population $P(t)$ is created from $P(t$-1), containing the double size of the initial population, including the offsprings created through the crossover operation. The mutation is applied to some individuals, according to the established probability (line 9). The population's evaluation is performed again to choose the survival individuals and the best one. If the best solution is improved, variable t comes back to zero (line 12). Thus, the population is reduced to half, i.e., its initial size (line 16). In the end of the algorithm execution, the best solution is returned (line 16).

6. Computational Results

The computational tests were executed using an instance generator. The input parameters correspond to configuration characteristics as follow: number and coordinates of the sensor and sink nodes, number of demand points, communication and sensing radius, area dimension, number and role specification, sensor node's capacity, non coverage penalty for a demand point, activation energy cost and routing energy cost (both depending on the role). The generator's output consists of a file representing the model presented in section 4. The same input parameters were used to solve the problem via GA.

It was considered a random sensor placement in a square area, and the sink nodes were disposed in the four extremities of the area. The demand points were located uniformly in the whole area and a random role was assigned to each one. Two scenarios were considered in the tests as presented in Table 1. The first one keeps the sensors density, i.e., the area grows if the number of sensor nodes is increased. The second scenario keeps the same area size while increasing the number of nodes. Both scenarios consist of four sets with 33 instances each one. Different costs were defined to each role, since they offer distinct services. Assuming that a video service implies in a bigger energy than a simple temperature one, the cost is based on the package size. Moreover, depending on the distance between the nodes, the transmission power can vary.

	Scenario 1				Scenario 2			
# of sensor nodes	12	25	51	102	12	25	51	102
# of sink nodes	4				4			
# of demand points	100				100			
# of roles	3				3			
comm. radius	5				5			
sensing radius	5				5			
area	10x10	15x15	22x22	30x30	10x10			
penalty	10000				10000			

Table 1: *Scenarios*

The tests were run using the optimization package CPLEX 9.0 [3], to solve the instance sets above. The GA was also applied to solve the instances. In both cases, the tests were carried out on a Pentium 4 at 2.4GHz and 1GB RAM. For each set, containing 33 instances of the same size (12, 25, 51 and 102 nodes), the average value is presented. We can see that CPLEX gives the optimal value while GA (avg) corresponds to the average of 10 executions of the same instance and GA (bst) shows the best execution. Evaluating the energy consumption in Figures 1 and 2, we can say that smaller instances consume less energy, in scenario 1. Increasing the number of nodes, the gap between the heuristic and CPLEX also increases. Scenario 2 has a different behavior due to fixing the area size and increasing the number of nodes. Note that we have more possibilities to choose which nodes should be activated, so the role assignment can be better. This can be proven by the decreasing energy values for higher number of nodes. As we have seen in scenario 1, the gap amplifies with the number of nodes. The computational times depicted in Figures 1 and 2 show that

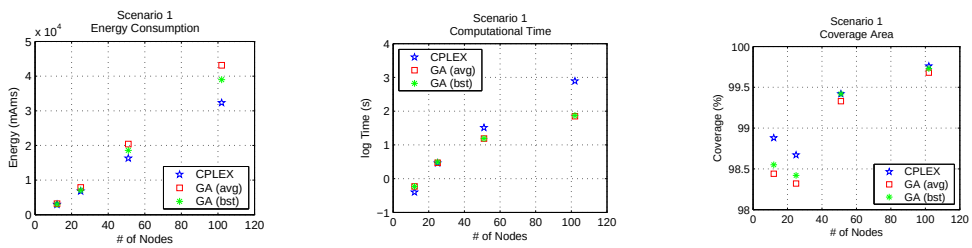

Figure 1: *Scenario 1*

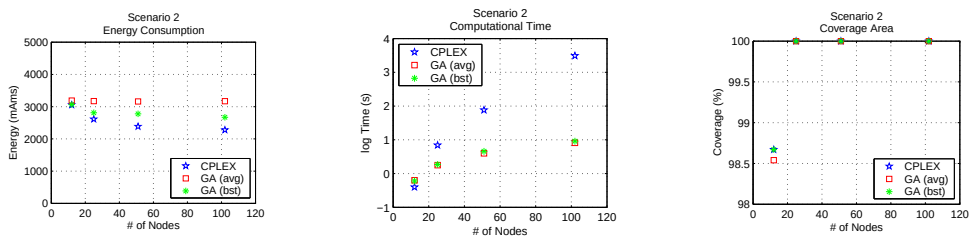

Figure 2: *Scenario 2*

time raises fast for the exact method, such that 102 nodes instances even spent an hour and half for some instances. On the other hand, note that the heuristic time is a little bit higher for 12 nodes instances, but for bigger instances the time keeps yet low. Next, we see the coverage area in Figures 1 and 2. In the first scenario, the coverage of the whole area was not possible. However, considering that we have a hundred

demand points, a coverage between 98% and 100% is acceptable. Scenario 2 shows a 100% coverage for 3 sets of instances. Only for 12 nodes 2% of points were not covered.

7. Conclusion

In the present work an optimization model for the role assignment problem in WSN was proposed as well as a genetic algorithm. Both methods were used to solve the problem and the results have shown that for small size instances the exact approach is able to find the optimal solution in a feasible time. However, increasing the number of sensor nodes, the time raises quickly, while the heuristic approach keeps a low time. Thus, although the heuristic does not guarantee the optimal solution, it can be seen through its behavior that good results were reached, taking into account the lower computational time. Must be observed that for WSN it is expected to work with thousand of nodes.

We can conclude that optimization techniques can be used to improve network design, as presented. The exact approach restricts the applicability only for small instances, but it reveals important in the comparison of different solution possibilities. As future works it is intended to develop the problem in a dynamic approach and provide a distributed algorithm to solve the problem. The metaheuristic GA allows the use of more complex functions later. Moreover, it can be possible to adjust it to the dynamic approach in an easy way.

References

[1] Akyildiz, I. F., W. Sankarasubramaniam, Y. Su, E. Cayirci, *Wireless Sensor Networks: A Survey*, Computer Networks, Vol. 38, pp. 393-422, 2002.

[2] Bhardwaj, M., A. P. Chandrakasan, *Bounding the Lifetime of Sensor Networks Via Optimal Role Assignments*, IEEE Infocom, New York, pp. 1587-1596, 2002.

[3] CPLEX, www.cplex.com.

[4] Dasgupta, K., M. Kukreja, K. Kalpakis, *Topology-aware placement and role assignment for energy-efficient information gathering in sensor networks*, IEEE Symposium on Computer and Communications (ISCC), pp. 341, 2003.

[5] Drezner, Z., H. W. Hamacher, *Facility location: applications and theory*, Springer-Verlag, Berlin, 2002.

[6] Frank, C., K. Rmer, *Algorithms for Generic Role Assignment in Wireless Sensor Networks*, ACM International Conference on Embedded Networked Sensor Systems (Sensys), pp. 230-242, 2005.

[7] Holland, J. H., *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, MA, 1992. First edition: University of Michigan Press, 1975.

[8] Hu, Z., B. Li, *On the Fundamental Capacity and Lifetime Limits of Energy-Constrained Wireless Sensor Networks*, IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2004), pp. 38-47, 2004.

[9] O'Rourke, J., *Art Gallery Theorems and Algorithms*, Oxford University Press, Oxford, 1987.

[10] Rajeswaran, A., R. Negi, *Capacity of Power Constrained Ad Hoc Networks*, Proc. IEEE Infocom, pp. 443-453, 2004.

[11] Slijepcevic, S., M. Potkonjak, *Power Efficient Organization of Wireless Sensor Networks*, IEEE International Conference on Communications, Vol. 2, pp. 472-476, 2001.

[12] Youssef-Massaad, P. G., M. Medard, L. Zheng, *Impact of Processing Energy on the Capacity of Wireless Channels*, International Symposium on Information Theory and its Applications (ISITA 2004), 2004.

[13] Zhao, W., M. Ammar, E. Zegura, *The energy-limited capacity of wireless networks*, IEEE Sensor and Ad Hoc Communications and Networks, pp. 279-288, 2004.