A Column Generation based Heuristic for Maximum Lifetime Coverage in Wireless Sensor Networks

Karine Deschinkel
LIFC Laboratory, Université de Franche-Comté
Belfort, France
Email: karine.deschinkel@univ-fcomte.fr

***Abstract*—Several studies in recent years have considered many strategies for increasing sensor network lifetime. We focus on a centralised management scheme where a large number of sensors are randomly deployed in a region of interest to monitor a set of targets and we propose an adaptive scheduling by dividing sensors into non-disjoint cover sets, each cover set being active in different period of time. In this paper, we design a column generation (CG) method based heuristic for efficiently solving the maximum lifetime coverage problem. We first model the problem with a linear programming (LP) formulation for non-disjoint cover sets where the objective is to maximise the sum of activation times of cover sets, with respect the sensor's battery lifetime. As the number of cover sets may be exponential to the number of sensors and targets, an initial set of cover sets is constructed and other cover sets are generated through the resolution of an auxiliary problem formulated as a integer programming (IP) problem. This problem is either solved at optimality by standard branch-and-bound or solved sub-optimally by a heuristic. Simulation results show the efficiency of the proposed heuristic which provides near optimal solutions while saving computational time.**

***Keywords*-target coverage; wireless sensor networks; centralised method; column generation;**

I. Introduction

Recent years have witnessed significant advances in wireless sensor networks which emerge as one of the most promising technologies for the 21st century [1]. In fact, they present huge potential in several domains ranging from health care applications to military applications. A sensor network is composed of a large number of tiny sensing devices deployed in a region of interest. Each device has processing and wireless communication capabilities, which enable to sense its environment, to compute, to store information and to deliver report messages to a base station. These sensor nodes run on batteries with limited capacities. To achieve a long life of the network, it is important to conserve battery power. Therefore, lifetime optimisation is one of the most critical issues in wireless sensor networks. In this paper we concentrate on the target coverage problem, with the objective of maximizing the network lifetime by using an adaptative scheduling. We assume that sensors are randomly sprayed for monitoring a set of targets with known locations and we also assume the sensors have location determination capabilities. We propose energy-aware centralized method by organizing the nodes in non-disjoint cover sets where each cover set is capable of monitoring all the monitored targets and by activate these cover sets successively. Scheduling and grouping sensors into non-disjoint sets is done by the base station, which informs every sensor of the time intervals to be activated. In this paper, the scheduling problem is formulated as a linear program where the variables are the active times of the different cover sets. The objective is to maximize the sum of their active times which corresponds to the network lifetime such that for any sensor its total active time does not exceed its initial lifetime. Unfortunately the number of cover sets may be huge (exponential in number of sensors and targets). Therefore we develop a resolution method based on a column generation (CG) process which is a well-known and widely practiced technique for solving linear programs with too many variables to include in the initial formulation explicitly. Our main contribution is to design a column generation method based heuristic for efficiently solving the maximum lifetime coverage problem.

The remainder of the paper is organized as follows. Section II reviews the related work in the field. Section III is devoted to the description of the target coverage problem and to its formulation by a linear program and explains the basics of the column generation technique. Next, in Section IV, we present the different algorithms of the proposed scheme. Section V discusses implementation details of our algorithms and shows the simulation results. Section VI concludes the paper.

II. Related work

Many works, including centralised, distributed and localized algorithms, have been proposed to extend the network lifetime. In this paper we focus on centralised algorithms because distributed algorithms are outside the scope of our work. Note that centralised coverage algorithms have the advantage of requiring very low processing power from the sensor nodes which have usually limited processing capabilities. Moreover, a recent study conducted in [2] concludes that there is a threshold in terms of network size to switch from a localized to a centralized algorithm. Indeed

 ISBN: 978-1-61208-144-1

the exchange of messages in large networks may consume a considerable amount of energy in a localized approach compared to a centralised one.

The authors in [3] and [4] independently highlight that the use of non-disjoint cover sets may increase the network lifetime by giving appropriate algorithms. For instance, Cardei et al. [3] formulate the maximum set covers problem as a mixed integer programming (MIP) and then apply a relaxation technique to design a LP-based heuristic of time complexity $O(n^3m^3)$ (where m is the number of targets). They also propose a greedy heuristic with a lower time complexity $O(dm^2n)$ (where d is the number of sensors that covers the most sparsely covered target). This heuristic forms individually set covers by covering first the most critical targets as in [5].

In [4], Berman et al. clearly provide a LP formulation for non-disjoint cover sets. We reuse this formulation in our work but instead of developping an approximation algorithm as in [4], we exploit the CG technique to deal with the huge number of variables in the LP formulation. In [4], the authors propose an algorithm with an approximation factor of $(1+\epsilon)(1+2\log n)$ for any $\epsilon > 0$ based on the $(1+\epsilon)$-approximation of the Garg and Könemann algorithm.

More recently Zorbas et al. [6] present a novel algorithm that can produce both disjoint cover sets as well as non-disjoint cover sets by using a cost function taking into account various parameters as the monitoring capabilities of a sensor, its association with poorly monitored targets and also its remaining battery life. Through simulations, they compare their proposed algorithm with other approaches found in [3] and [5] and show that it exhibits comparable results in terms of generated cover sets but in faster execution times.

The closest work to ours are [7] and [8]. In [7] and [8], the authors adress the problem of maximizing lifetime in wireless network subject to QoS, energy or coverage requirements. They propose a CG approach to decompose the original formulation into a master problem and an auxiliary problem. The auxiliary (AUX) problem is an IP problem in [8] or a MIP problem in [7] which is solved at optimality by a branch-and-bound algorithm. In both papers analyses show that the resolution at optimality of the AUX-problem is time-consuming. Based on this observation, we propose in our work a heuristic for adressing the AUX-problem which produces good solutions in lower computational times.

III. Problem description

We try to produce an adaptive scheduling which allows sensors to operate alternatively so as to prolong the network lifetime. For convenience, the notations and assumptions are described first. Then the lifetime problem of sensor network covering a set of targets is formulated.

A. Notations and assumptions

- m : the number of targets
- n : the number of sensors
- I : the set of targets
- K : the set of sensors
- S_i : set of sensors which cover the target i
- Z_k : set of targets covered by the sensor k
- T_k : the lifetime of a sensor k, which is time duration when the sensor k is in the active state all the time

B. Formulation

The problem of monitoring all targets by activate non-disjoint cover sets successively in order to extend the network lifetime can be formulated as a LP. The variables in the LP are as follows. t_u is the lifetime of the cover set u, that means that all sensors in the cover set u are active during the time period t_u. We denote by U the set of all elementary cover sets. The problem is as follows :

$$\begin{cases} \max \sum_{u \in U} t_u \\ \text{subject to :} \\ \sum_{u \in U} a_{ku} t_u \leq T_k, \quad \forall k \in K \\ t_u \in R^+ \end{cases} \tag{1}$$

The objective function maximizes the total work time of all the cover sets. The constraint shows the lifetime constraint for each sensor k. a_{ku} is a binary index which is set to 1 if sensor k is active in the cover set u and 0 otherwise. An elementary cover set corresponds to a configuration where all targets are covered as well as no superfluous sensor is activated. However the number of elementary cover sets is very high.

C. Example

To illustrate our problem we provide a simple example with only 10 sensors and 4 targets. Table I presents the sensors which are able to cover each target. We consider only two cover sets. Each cover set is given as a tuple and covers all targets.

- Cover set 0 = (0,3,9).
- Cover set 1 = (0,4,8).

Target	Sensors
0	3,4,9
1	4,6,9
2	0,2
3	1,3,5,8

Table I
Collection of sensors to monitor a target

 ISBN: 978-1-61208-144-1

The linear programming corresponding to this simple example is the following :

$$
\left\{
\begin{array}{lllllll}
 & \max & t_0 & + & t_1 & & \\
\text{subject to :} & & & & & & \\
 & & t_0 & + & t_1 & \leq 1.00 & \text{(sensor 0)} \\
 & & t_0 & & & \leq 1.00 & \text{(sensor 3)} \\
 & & & & t_1 & \leq 1.00 & \text{(sensor 4)} \\
 & & & & t_1 & \leq 1.00 & \text{(sensor 8)} \\
 & & t_0 & & & \leq 1.00 & \text{(sensor 9)} \\
t_0, t_1 \in (R)^+ & & & & & &
\end{array}
\right. \quad (2)
$$

t_0 and t_1 are respectively the lifetimes of the cover sets 0 and 1. The right part of each inequality corresponds to the maximal lifetime of each sensor. Here sensor lifetime is set to 1. In this example we do not enumerate all elementary cover sets, we solve the LP with only two cover sets and the network lifetime obtained is equal to 1 where $t_0^* = 0.5$ and $t_1^* = 0.5$. That means that sensors of cover set 0 are active during 0.5 time unit and sensors of the cover set 1 are active during 0.5 time unit. Note that some sensors (1,2,5,6,7) do not appear in the LP because it is not part of any generated cover sets. Only sensor 0 has consumed its entire energy. If we had generated more cover sets, we would have reached the maximal lifetime of the network which is equal to 2. For instance, if we add cover set 2 = (2,3,4) and cover set 3 =(1,2,9), the optimal scheduling is obtained with $t_0^* = t_1^* = t_2^* = t_3^* = 0.5$.

D. Column generation method

As the set U of elementary cover sets may be huge we use a CG technique [9] to solve (1). That means that we solve a Restricted Master Problem (RMP) with only a subset $U' \subseteq U$ of elementary cover sets and we introduce an attractive cover set if necessary. Given a subset $U' \subseteq U$ and the dual multipliers $\pi_k \equiv \pi_k(U')$ for sensors k, the AUX-problem consists in finding the most attractive cover set $u \in U \setminus U'$, that means the cover set u with the maximal reduced cost $r_u = (1 - \sum_{k \in K} a_{ku}\pi_k)$. If $r_u > 0$ then the cover set u is said to be attractive and it is added in the formulation of the RMP, otherwise the problem (1) is optimal.

The AUX-problem is to find a new feasible cover set u which maximizes $r = (1 - \sum_{k \in A} \pi_k)$, or which minimizes $\sum_{k \in A} \pi_k$ where A denotes the set of active sensors in the cover set u. If this sum is less than 1, then the new valid cover set u is added in the Restricted Master Problem. We formulate the AUX-problem as an IP problem with binary variables y_k for each sensor k which is set to 1 if sensor k is active in the cover set u, and 0 otherwise. The following constraints represent the coverage guarantee for each target i ($1 \leq i \leq m$).

$$\sum_{k \in S_i} y_k \geq 1 \quad \forall i \in I$$

The AUX-problem is formulated as follows :

$$
\left\{
\begin{array}{ll}
\min \sum_{k \in K} \pi_k y_k & \\
\text{subject to :} & \\
\sum_{k \in S_i} y_k \geq 1 & \forall i \in I \\
y_k \in \{0, 1\} & \forall k \in K
\end{array}
\right. \quad (3)
$$

Note that the formulation of the AUX-problem corresponds to the model of the classical set covering problem [10]. This complete CG approach seems to be efficient. The RMP is formulated as a (LP) (1) where the entire set U of elementary cover sets must be replaced by a subset U' which contains initially a small number of elementary cover sets. Then the RMP (1) and the AUX-problem (3) are solved sequentially and the set U' grows up until no attractive cover set is generated. The optimal solution, that means the adaptive scheduling of cover sets which maximizes the network lifetime, is always found. The RMP is a classical LP problem, thus can be solved in polynomial time $O(n^3m^3)$ with the algorithm proposed by Ye [11]. However the AUX-problem, which is a IP problem, may require a large unacceptable running time (This problem is also classified as NP-hard [12]). The intuitive idea is to speedup the generation of attractive cover sets by the use of a heuristic. To measure the efficiency of our approach, we design three methods, called respectively the Exact Method, the Heuristic Method and the Mixed Method. The Exact Method consists of solving to optimality the AUX-problem at each step of the column generation process with an IP solver. For the Heuristic Method, we propose to generate an attractive cover set taking into account the dual multipliers of sensors k and without resolving the auxiliary problem at optimality. In the Mixed Method, in case of impossibility for the heuristic to generate an attractive cover set, we solve the (AUX)-problem at optimality. Note that the Exact and the Mixed methods lead to an optimal scheduling compared to the Heuristic Method which provides near optimal solution. The resolution method based on CG technique and its three versions are explained in more details in the following part.

IV. Resolution method

The resolution method requires to generate some elementary cover sets to form the set U' at the beginning. Note that the initial number of cover sets will not affect the final optimal output. The generation of elementary cover sets involves two steps. First, a cover set is generated and is then analyzed to determine if some sensors are not superfluous.

A. Production of cover set

The algorithm 1 ensures the production of a cover set where all targets are covered. This algorithm does not produce an elementary cover set because some active sensors could be superfluous. That is why we have to check if it is possible to desactivate some sensors through algorithm 2. This algorithm is applied for each generated cover set whatever the generation process.

 ISBN: 978-1-61208-144-1

Algorithm 1 Cover_Set_Generation(u)

Require: A set of targets I, a set of sensors K
Ensure: A random cover set u
  $V \leftarrow I$
  $u \leftarrow \emptyset$
  while V is not $\emptyset$ **do**
    Select a target $i \in V$ randomly
    $V \leftarrow V \setminus \{i\}$
    Select randomly a sensor $k \in S_i$ to cover the target i
    $u \leftarrow u \cup \{k\}$
    for all targets $h \in Z_k$ **do**
      $V \leftarrow V \setminus \{h\}$
    end for
  end while

Algorithm 2 Check_Elementary_Cover_Set(u)

Require: A cover set u
Ensure: An elementary cover set u
  $G \leftarrow u$
  while G is not $\emptyset$ **do**
    Select a sensor $k \in G$ randomly
    Check if it is possible to desactivate sensor k
    if yes **then**
      $u \leftarrow u \setminus \{k\}$
    end if
    $G \leftarrow G \setminus \{k\}$
  end while

B. Generation of an attractive cover set

Once some elementary cover sets are formed and compose the initial set of variables/columns, the CG process consists of introducing new attractive columns in the RMP. This task may be done by the exact resolution of the IP AUX-problem as written in the algorithm 3 or by using a heuristic as described in the algorithm 4.

Algorithm 3 Generation_Attractive_CoverSet_Exact(π,u,r)

Require: The dual multipliers π_k for each sensor $k \in K$
Ensure: The generated cover set u and the associated reduced cost r
  $u \leftarrow \emptyset$
  $r \leftarrow 0$
  Solve the IP (3) with dual multipliers π_k
  $(y_k^*) \quad \forall k \in K$ are the optimal values
  for all $k \in K$ **do**
    if $y_k^* = 1$ **then**
      {The sensor k is active}
      $u \leftarrow u \cup \{k\}$
    end if
  end for
  $r \leftarrow 1 - \sum_{k \in u} \pi_k$

Here a heuristic is proposed to provide a new cover set u such that all targets are covered. Considering the dual multiplier π_k for each sensor k as a cost, the objective is to activate less expensive sensors in the cover set such that the resulting reduced cost of this cover set is positive. We first select randomly a target, then we choose a sensor with minimal cost that covers this target. We repeat the process until all targets are covered. If there are multiple sensors of minimum costs, the choice of one of them is made randomly. The algorithm 4 of complexity $O(mn)$ presents the generation of an attractive cover set with the heuristic. As our heuristic integrates a random part, it may be applied several times (no more than Nb_Max_Ite iterations) until a cover set with positive reduced cost is found. Note that the two generation methods do not necessarily generate an elementary cover set. Each time an attractive cover set is generated, we call the algorithm 2 to eliminate superfluous sensors.

Algorithm 4 Generation_Attractive_CoverSet_Heuristic(π,u,r)

Require: The dual multipliers π_k for each sensor $k \in K$
Ensure: The generated cover set u and the associated reduced cost r
  $V \leftarrow I$
  $u \leftarrow \emptyset$
  $r \leftarrow 0$
  while V is not $\emptyset$ **do**
    Select a target $i \in V$ randomly
    Select a sensor $k \in S_i$ with minimal cost (π_k)
    $V \leftarrow V \setminus \{i\}$
    $u \leftarrow u \cup \{k\}$
    $r \leftarrow r + \pi_k$
    for all targets $h \in Z_k$ **do**
      $V \leftarrow V \setminus \{h\}$
    end for
  end while
  $r \leftarrow 1 - r$

C. Global process

The algorithm 5 presents our resolution method based on CG which provides a cover set's scheduling to prolong the network lifetime.

V. EXPERIMENTAL RESULTS

Section V is dedicated to experimental results.

A. Experimental setup and environment

We have implemented the three methods presented in section IV. Our experiments have been conducted on a regular Linux workstation with a AMD Athlon(tm) 64 X2 Dual Core Processor 4000+ of 2,1 GHz. Resolution of the LP or IP problems are respectively carried out the simplex method and the branch-and-bound method implemented in

 ISBN: 978-1-61208-144-1

Algorithm 5 Resolution Method

$U \leftarrow \emptyset$
{Generation of E elementary cover sets}
for $e = 0$ to E **do**
  Cover_Set_Generation(u)
  Check_Elementary_Cover_Set(u)
  $U \leftarrow U \cup u$
end for
Restricted_Master_Problem_Resolution(U)
$Stop \leftarrow 0$
while ($Stop = 0$) **do**
  $r \leftarrow 0$
  {Search of an attractive cover set (3 versions)}

  Version 1 : Exact Method
  Generation_Attractive_CoverSet_Exact(π,u,r)

  Version 2 : Heuristic Method
  $Nb\_Ite \leftarrow 0$
  while (($r <= 0$) and ($Nb\_Ite \leq Nb\_Max\_Ite$)) **do**
    Generation_Attractive_CoverSet_Heuristic(π,u,r)
    $Nb\_Ite \leftarrow Nb\_Ite + 1$
  end while

  Version 3 : Mixed Method
  $Nb\_Ite \leftarrow 0$
  while (($r <= 0$) and ($Nb\_Ite \leq Nb\_Max\_Ite$)) **do**
    Generation_Attractive_CoverSet_Heuristic(π,u,r)
    $Nb\_Ite \leftarrow Nb\_Ite + 1$
  end while
  if ($r <= 0$) **then**
    Generation_Attractive_CoverSet_Exact(π,u,r)
  end if

  if ($r <= 0$) **then**
    {the method did not provide an attractive cover set}
    $Stop \leftarrow 1$
  else
    {An attractive cover set is added}
    $U \leftarrow U \cup \{u\}$
    Restricted_Master_Problem_Resolution(U)
  end if
end while

GLPK (GNU linear Programming Kit) [13] available in the public domain.

In this section we evaluate the performance of our algorithms by way of simulations. We simulate a network with sensor nodes and target points randomly located in a $500m \times 500m$ area. We assume the sensing range is equal for all the sensors in the network and is set to $150m$. In the different scenarios we vary the number of randomly deployed sensor nodes n between 50 and 200 with an increment of 50. The number m of targets to be covered varies between 30 and 120 with an increment of 30. Each sensor has a lifetime of 1. The following requirements are satisfied: each sensor covers at least one target and each target is covered by at least one deployed sensor, the connectivity of the network is ensured and all sensors are capable of communicating with the base station. We measure the network lifetime and the execution times. For each scenario, results are averages over 10 instances (we generate 10 random topologies per scenario). In the algorithms we set Nb_Max_Ite to 10. The set of elementary cover sets is initialized with ($E = 10$) configurations.

B. Comparison of the execution times

First we compare and comment the CPU execution times of the different resolution methods. Table II gives the optimal network lifetime and the distribution of the execution times (in seconds) for the three methods over the 16 scenarios. Results of table II are consistent with those obtained in the literature : network lifetime and execution times increase with sensor density, network lifetime decreases with targets number for a fixed number of sensors because they are more requested. From the above results, we see that the Mixed Method can be up to 6 times faster than the Exact Method which solves an integer programming at each iteration. And the computing times of the Heuristic Method is really lower than the others each time that the number of targets exceeds 60. We observe that the Mixed Method uses 1.83 times on average the algorithm 3 for the resolution of the AUX-problem at optimality, which is really little but enough to slow its execution time.

N	M	$Lifetime$	$Exact$	$Heuristic$	$Mixed$
50	30	3.8	0.25	0.30	0.12
	60	3.0	1.03	0.53	0.52
	90	2.8	2.95	0.82	1.55
	120	2.7	8.40	1.20	4.03
100	30	8.7	3.29	2.97	1.03
	60	7.2	26.53	4.25	8.41
	90	6.9	243.95	6.82	74.19
	120	6.7	749.46	9.70	220.64
150	30	14.7	17.17	14.51	4.94
	60	12.3	315.66	22.21	48.96
	90	11.8	2365.65	30.61	525.21
	120	11.3	9249.81	48.15	1987.04
200	30	19.6	38.80	34.85	9.50
	60	17.3	750.40	56.34	126.39
	90	16.6	8229.53	132.46	1297.82
	120	15.5	28942.49	105.87	4393.04

Table II
EXECUTION TIMES (IN SECONDS) BETWEEN THE 3 METHODS

C. Comparison of the objective value

We compare the optimal solution value obtained with the Exact Method with approximate solution values obtained with the Heuristic Method. And we conclude that the Heuristic Method is a very efficient method because this method is able of finding the same solution value as the value of the optimal solution in all simulated cases with computing times drastically lower. We have also tested an other heuristic which is not presented here due to space limitations and this second heuristic finds the same solution values as the Exact Method with two exceptions over the 160 tests and the difference is equal to 0.0885 in the first case and to 0.2482 in the second case. These results are very promising and should be confirmed on other problem instances with a larger number of sensors and targets.

D. Comparison of the number of generated cover sets

Table III gives the distribution [1] of the number of attractive cover sets which have been generated to build an adaptive scheduling. We see that the scheduling resulting from the Heuristic Method contains a higher number of cover sets. We may have expected this result because the Exact Method always generates the most attractive cover set at each iteration so that the maximal lifetime of the network is reached with a minimal number of cover sets. The Heuristic Method could be improved to generate less cover sets but more attractive. Nevertheless this method is very efficient as it quickly produces a good solution by generating a slightly higher number of cover sets. And it may be interesting for wireless sensor networks to deal with a large number of cover sets such that sensor nodes frequently oscillate between an active and an inactive state as recommended in [5].

	$Exact$	$Heuristic$	$Mixed$
MIN	1.00	1	1.00
FST	10.00	15	16.00
MED	24.50	38	34.00
THD	58.25	74,5	69.50
MAX	130.00	131	124.00

Table III
NUMBER OF GENERATED ATTRACTIVE COVER SETS

VI. CONCLUSION AND FUTURE WORK

Energy-efficiency is crucial in power-limited wireless sensor network, since nodes have significant power constraints (battery life). In this paper we have investigated the problem of prolonging the network lifetime by organizing sensors into non-disjoint cover sets which operate sucessively in order to monitor all targets. We have formulated this problem as a linear programming where variables are the activation times of the cover sets and we have proposed a column generation approach to solve it. Instead of solving the auxiliary problem to optimality to generate an attractive cover set, we design a efficient heuristic. Simulation results show the performance of the heuristic which obtains very good solutions with very low time complexity. Although the method is a centralised one, it may be used to measure the quality of distributed solutions and it can be easily extended to deal with different QoS requirements.

[1] MIN stands for MINimum, FST for FirST quantile (25% of the population), MED for MEDian (50 % of the population), THD for THirD quantile (75% of the population) and MAX for MAXimum

REFERENCES

[1] I. Akyildiz, W. Su, Y.Sankarasubramniam, and E. Cayirci, "A survey on sensor networks," *IEEE Comm. Magazine*, pp. 102–114, 2002.

[2] T. Padmavathy and M. Chitra, "Extending the network lifetime of wireless sensor networks using residual energy extractionhybrid scheduling algorithm," *Int. J. of Communications, Network and System Sciences*, vol. 3, no. 1, pp. 98–106, 2010.

[3] M. Cardei, M. T. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *in IEEE Infocom*, 2005, pp. 1976–1984.

[4] P. Berman, G. Calinescu, C. Shah, and A. Zelikovsky, "Power efficient monitoring management in sensor networks," in *Wireless Communications and Networking Conference, WCNC. 2004*, 2004.

[5] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," in *IEEE International conference on Communications*, 2001, pp. 472–476.

[6] D. Zorbas, D. Glynos, P. Kotzanikolaou, and C. Douligeris, "Solving coverage problems in wireless sensor networks using cover sets," *Ad Hoc Netw.*, vol. 8, pp. 400–415, June 2010.

[7] A. Alfieri, A. Bianco, P. Brandimarte, and C.-F. Chiasserini, "Maximizing system lifetime in wireless sensor networks," *European Journal of Operational Research*, vol. 181, no. 1, pp. 390–402, 2007.

[8] Y. Gu, Y. Ji, J. Li, and B. Zhao, "Qos-aware target coverage in wireless sensor networks," *Wirel. Commun. Mob. Comput.*, vol. 9, pp. 1645–1659, December 2009.

[9] G. Dantzig and P. Wolfe, "Decomposition principle for linear programs," *Operations Research*, pp. 101–111, 1960.

[10] A. Caprara, M. Fischetti, and P. Toth, "Algorithms for the set covering problem," *Annals of Operations Research*, vol. 98, p. 2000, 1998.

[11] Y. Ye, "An o(n^3l)) potential reduction algorithm for linear programming," *Mathematical Programming*, vol. 50, pp. 239–258, 1991.

[12] M. Garey and D. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.

[13] A. Mahkorin, *GNU Linear Programming Kit, Reference Manual*, 2010.