Optim Lett (2013) 7:157–172
DOI 10.1007/s11590-011-0405-0

ORIGINAL PAPER

α-Coverage to extend network lifetime on wireless sensor networks

Monica Gentili · Andrea Raiconi

Received: 11 February 2010 / Accepted: 24 September 2011 / Published online: 26 October 2011

Abstract An important problem in the context of wireless sensor networks is the Maximum Network Lifetime Problem (MLP): find a collection of subset of sensors (cover) each covering the whole set of targets and assign them an activation time so that network lifetime is maximized. In this paper we consider a variant of MLP, where we allow each cover to neglect a certain fraction $(1-\alpha)$ of the targets. We analyze the problem and show that the total network lifetime can be hugely improved by neglecting a very small portion of the targets. An exact approach, based on a Column Generation scheme, is presented and a heuristic solution algorithm is also provided to initialize the approach. The proposed approaches are tested on a wide set of instances. The experimentation shows the effectiveness of both the proposed problems and solution algorithms in extending network lifetime and improving target coverage time when some regularity conditions are taken into account.

Keywords Sensor networks · Network lifetime · α-Coverage · Delayed column generation

1 Introduction

A wireless sensor network generally consists of a large number of sensors which perform together a complex sensing task that can be spread on a wide geographical region.

M. Gentili
Department of Computer Science, University of Salerno, Via Ponte Don Melillo, 84084 Fisciano, SA, Italy
e-mail: mgentili@unisa.it

A. Raiconi (✉)
Department of Mathematics, University of Salerno, Via Ponte Don Melillo, 84084 Fisciano, SA, Italy
e-mail: araiconi@unisa.it

Each device has a maximum sensing range and can collect information on a given sub-region (*sensing area*). Depending on the application, covering a specific region might imply covering its whole area (*area coverage problems*) or specific target points inside it (*target coverage problems*). However, as shown in [7], an area coverage problem can always be transformed into an equivalent target coverage problem in polynomial time. Therefore, in the following, we will always consider target coverage problems.

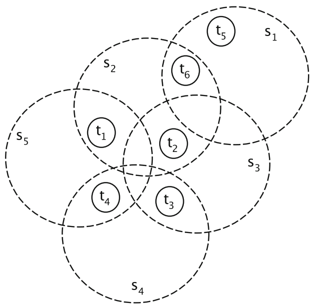
The design and management of a wireless sensor network presents several critical issues that can be effectively approached using optimization; for some recent developments see [2] and [9]. One of these fundamental aspects is the lifetime of the network, that is, the amount of time in which this monitoring activity can be performed. An obvious constraint on this factor is the limited power of the battery contained in the sensors, both for cost and size reasons. The lifetime of the network can be maximized by individuating *covers* (subsets of sensors that can cover the whole set of targets) and activating them subsequently for proper amounts of time. This problem is known as the *Maximum Network Lifetime Problem (MLP)*. MLP was proved to be NP-complete by reduction from the 3-SAT problem in [4]. Many different solution approaches have been proposed in the literature to solve it either exactly (see [6]) or approximately [1,3,4,7].

In this paper, we address a variant of MLP where we allow each cover to neglect a certain fraction $(1-\alpha)$ of the targets. We refer to these covers as α-covers and to the resulting problem as the *Maximum Network α-Lifetime Problem* $(\alpha\text{-}MLP)$.

The assumption to neglect the coverage of a portion of the area is not a limiting one. First of all in some cases it could lead to better solutions both in terms of network lifetime and targets coverage time. Consider the example in Fig. 1, with six targets and five sensors, where each sensor is represented by its sensing area. Assume that the battery life of each sensor is normalized to 1 unit of time. The maximum network lifetime is equal to 1, which is also the coverage time of each target. For example, an optimal solution consists in activating for one unit of time the cover $\{s_1, s_2, s_4\}$. Assume now, to allow each cover to neglect at most one of the six targets. We could choose two α-covers, namely $\{s_2, s_4\}$ and $\{s_1, s_3, s_5\}$, and activate each of them for one unit of time. With this solution, network lifetime is equal to 2 and coverage time of the targets has doubled as well for all the targets but for target t_5, for which it is not changed and is equal to 1.

Even when situations as the one above described do not occur, the α-coverage might be acceptable for various applications. For example, if we are monitoring the

Fig. 1 A sensor network with six targets, $T = \{t_1, \ldots, t_6\}$ and five sensors, $S = \{s_1, \ldots, s_5\}$



average pollution level of a certain region, excluding a small percentage of measurements each time the average is computed, could be practically irrelevant. In addition, by appropriately switching among the α-covers, the uncoverage of the targets can be virtually invisible to the underlying application. However, in some situations targets with unfortunate coverage levels could be excluded by most or even each α-cover. The worst case scenario is when the neglected set of targets never changes. For this reason, we investigated some regularity conditions to guarantee a minimum coverage level to the entire set of targets. The resulting problem is referred to as the *Regular Maximum Network α-Lifetime Problem* (α-$RMLP$). We will show that, when α-covers are considered together with some regularity conditions, not only the network lifetime could be increased, but also the total coverage time of each target could be effectively improved as well.

There are very few contributions in the literature addressing α-coverage problems. In particular, the authors in [10], defined the problem in terms of area coverage, and presented two approaches to determine an upper bound to the optimum network lifetime as well as a greedy heuristic to find a feasible solution. In [5] and [8], the interest is focused in minimizing the number of uncovered targets (*breach*).

The sequel of the paper is organized as follows. In Sect. 2 we give the needed notations. Section 3 formally defines the two problems studied in this paper. Section 4 provides the mathematical formulations and some insights on the computational complexity. Section 5 presents a Delayed Column Generation exact approach for α-MLP and α-RMLP. Section 6 describes a heuristic algorithm we developed both to obtain fast feasible solutions and to initialize our exact method. Experimental results are presented in Sect. 7. Finally Sect. 8 summarizes our work and presents future research guidelines.

2 Notation

Let $T = \{t_1, \ldots, t_n\}$ be the set of n target points and $S = \{s_1, \ldots, s_m\}$ the set of m sensors that constitute a wireless sensor network. For each sensor $s_i \in S$, let $T_{s_i} \subseteq T$ be the subset of targets covered by s_i. Since the positions of all the targets and sensors are fixed, we can assume each T_{s_i}, $\forall s_i \in S$, is known in advance. Given a subset $C \subseteq S$ of sensors, we define the set of targets covered by C as $T_C = \bigcup_{s_i \in C} T_{s_i}$. By extension, each target in T_C is said to be covered by C. A *cover* $C \subseteq S$ is a subset of sensors such that $T_C \equiv T$. A cover C is *minimal* if there does not exist $C' \subset C$ such that $T_{C'} \equiv T$, that is C does not contain a proper subset that is a cover as well. Given, a value $\alpha \in [0, 1]$, an $\alpha - cover$ C^α is a subset of sensors that covers at least T_α targets, that is $|T_{C^\alpha}| \geq T_\alpha$, where $T_\alpha = n \times \alpha$. For example if we fix a value of α such that $T_\alpha = 5$, in the network of Fig. 1, then possible α-covers are: $C_1^\alpha = \{s_1, s_2, s_3\}$ or also $C_2^\alpha = \{s_1, s_2, s_4\}$. Obviously, every cover is an α-cover, and, when $\alpha = 1$ an α-cover is a cover. We define an $\alpha - cover$ C^α to be minimal, if there does not exist $C' \subset C^\alpha$ such that $|T_{C'}| \geq T_\alpha$. We define an $\alpha - cover$ C^α to be *target-minimal* if it does not contain an α-cover covering the same set of targets, that is, there does not exist $C' \subset C^\alpha$ such that $T_{C'} \equiv T_{C^\alpha}$. We obviously have that a minimal α-cover is also target-minimal, but the contrary is not always true. Consider again the network of Fig. 1 and fix $T_\alpha = 4$. The $\alpha - cover$ $C_A^\alpha = \{s_2, s_3, s_5\}$ is not minimal, it contains, indeed, the $\alpha - covers$: $C^\alpha_B = \{s_2, s_3\}$, $C^\alpha_C = \{s_2, s_5\}$ and $C^\alpha_D = \{s_3, s_5\}$; however, it is target-minimal, indeed $T_{C^\alpha_A} \neq T_{C^\alpha_B}$, $T_{C^\alpha_A} \neq T_{C^\alpha_C}$ and $T_{C^\alpha_A} \neq T_{C^\alpha_D}$.

In the next section we formally define our two variants of the MLP that use the concept of α-cover.

3 Problems definition

We generally assume that all sensors are based on the same hardware, and therefore they all have the same battery life that we assume to be normalized and be equal to 1. The Maximum Network α-Lifetime Problem is defined as follows:

Maximum Network α-Lifetime Problem (α-MLP)

Given a value $\alpha \in [0, 1]$, find a collection of pairs (C^α_j, w_j), $j = 1, 2, \ldots, p$, where C^α_j is an α-cover and w_j is its corresponding activation time, such that the sum of the activation times $\sum_{j=1}^p w_j$ is maximized, and, each sensor is used for a total time that does not exceed its battery: $\sum_{j\in\{1,\ldots p\}|s_i\in C^\alpha_j} w_j \leq 1$ for each $s_i \in S$.

When $\alpha = 1$ the α-MLP is exactly the MLP defined in [4].

As explained in the introduction, a solution to α-MLP could sometimes provide irregular coverage of the targets. Therefore, we introduce the concept of regularity of a collection of α-covers. In particular, given a collection of $\alpha - covers$ C^α_j, $j = 1, 2, \ldots p$, we say it is *regular* if each target is covered for a total time that is not less than a predefined threshold w_{min}, i.e.,

$$\sum_{j|t_k\in T_{C^\alpha_j}} w_j \geq w_{min} \quad \forall t_k \in T \tag{1}$$

Other regularity metrics could be considered, such as, for example, requiring each target to be covered by a number of covers greater than or equal to a predefined threshold, or also, maximizing the minimum coverage time among all the targets. We focus in this paper on the above definition of regularity, the analysis of other metrics is object of further research as outlined in the concluding section. Based on the above regularity definition, we can formally state now, the Regular Maximum α-Lifetime Problem as follows:

Regular Maximum Network α-Lifetime Problem (α-RMLP)

Given a value $\alpha \in [0, 1]$, find a collection of pairs (C^α_j, w_j), $j = 1, 2, \ldots, p$, where C^α_j is an α-cover and w_j is its corresponding activation time, such that the sum of the activation times $\sum_{j=1}^p w_j$ is maximized, each sensor is used for a total time that does not exceed its battery, i.e., $\sum_{j\in\{1,\ldots p\}|s_i\in C^\alpha_j} w_j \leq 1$ for each $s_i \in S$, and the set of α-covers is regular.

4 Mathematical formulation and complexity

The mathematical formulation given in this section for α-MLP is the same as the one proposed in the literature, for example in [6] for the MLP. This formulation assigns a variable to each feasible cover (α-cover) representing its activation time in the final solution, such that the total activation time of each sensor is not greater than its battery life. When applied to a particular wireless sensor network, the difference between the MLP and α-MLP formulation is only in the total number of variables; in particular it will be the same if $\alpha = 1$. On the other hand, some modifications to the formulation are required in order to model α-RMLP, as we will show in this section.

Let $C_1^\alpha, \ldots, C_\ell^\alpha$ be the collection of all the possible feasible α-covers of a given wireless sensor network. We define variables $w_1, \ldots, w_\ell$ to denote the activation time of each α-cover. Moreover, we consider, for each sensor s_i, $i = 1, 2, \ldots, m$, and each $\alpha - cover\ C_j^\alpha$, $j = 1, 2, \ldots, \ell$, the binary parameter a_{ij} to be equal to 1 if s_i belongs to C_j^α, and 0 otherwise. The mathematical formulation of α-MLP is the following:

$$[\boldsymbol{\alpha} - \mathbf{MLP}] \qquad \max \sum_{j=1}^{\ell} w_j \tag{2}$$

$$\text{s.}t. \sum_{j=1}^{\ell} a_{ij} w_j \leq 1 \ \forall i = 1, \ldots, m \tag{3}$$

$$w_j \geq 0 \ \forall j = 1, \ldots, \ell \tag{4}$$

The sum of the activation times of all the α-covers determines the network lifetime that is maximized by the objective function (2). Constraints (3) ensure the battery lifetime of each sensor is not exceeded. Since the number of variables of the model is huge, to exactly solve it, we embedded the model into a Delayed Column Generation approach that is described in Sect. 5.

Let us consider regularity condition (1) to model α-RMLP. We need to consider the additional binary parameter b_{kj}, for each $\alpha - cover\ C_j^\alpha$ and each target t_k, that is equal to 1 if target t_k is covered by C_j^α, and 0 otherwise. Moreover, let w_{min} be the minimum coverage threshold value. The mathematical formulation **[α-RMLP]** is obtained from **[α-MLP]** by adding the following set of constraints that ensure the resulting collection of α-covers satisfies the chosen regularity condition:

$$\sum_{j=1}^{\ell} b_{kj} w_j \geq w_{min} \quad \forall k = 1, \ldots, n \tag{5}$$

Both the problems are NP-hard. Indeed, MLP is a special case of α-MLP when $\alpha = 1$, and α-MLP is a special case of α-RMLP when $w_{min} = 0$.

5 Column generation approach

In [6] the authors present a Delayed Column Generation (CG) approach for MLP, here we present this method adapted to solve α-MLP and α-RMLP.

Consider the mathematical formulation **[α-MLP]**, restricted only to a subset of feasible α-covers. Let π_i, $i = 1, 2, \ldots, m$, be the set of dual optimal multipliers associated with each primal constraint (that is, with each sensor). To check for optimality the following separation problem can be solved:

$$\textbf{[S1]} \qquad \min \sum_{i=1}^{m} \pi_i x_i \tag{6}$$

$$\text{s.}t. \sum_{i=1}^{m} \delta_{ki} x_i \geq y_k \quad \forall k = 1, \ldots, n \tag{7}$$

$$\sum_{k=1}^{n} y_k \geq T_\alpha \tag{8}$$

$$x_i \in \{0, 1\} \quad \forall i = 1, \ldots, m \tag{9}$$

$$y_k \in \{0, 1\} \quad \forall k = 1, \ldots, n \tag{10}$$

where x_i is a binary variable that is equal to 1 if sensor s_i is selected to be in the α-cover, and is equal to 0 otherwise; y_k is a binary variable that is equal to 1 if target t_k is covered by the α-cover, and is equal to zero otherwise; δ_{ki} is a binary parameter that is equal to one if target t_k is covered by sensors s_i and 0 otherwise. Objective function (6) ensures the returned α-cover has the minimum reduced cost. Constraints (7) impose y_k can be set to one only if at least one of the sensors that covers target t_k has been selected. Constraint (8) ensures the total number of covered targets is at least T_α. This separation problem differs from the one used in [6] for MLP since it uses additional decision variables $y_k, \forall k \in 1, \ldots, n$, to identify covered targets, and adds constraint (8) to check whether they are enough to obtain an α-cover. If the optimal objective function value of **[S1]** is not less than 1, then the solution of the restricted problem is optimal for the entire problem; otherwise, the returned new column (defined by the optimal solution value of variables x_i) is introduced into the master problem and the process is iterated.

The choice of the initial set of columns for the restricted master is important for the convergence of the algorithm. In Sect. 6 a heuristic procedure, we developed to initialize the procedure, is described. Moreover, to avoid the generation of already generated covers the following set of constraints can be added to **[S1]**. Let $C_1^\alpha, C_2^\alpha, \ldots, C_u^\alpha$ be the set of α-covers generated by the algorithm so far:

$$\sum_{i=1}^{m} a_{ij} x_i \leq \sum_{i=1}^{m} a_{ij} - 1 \quad \forall j = 1, \ldots, u \tag{11}$$

Constraints (11) are the same as the ones applied to MLP in [6]. These inequalities ensure that the new α-cover returned by the separation problem differs from the already generated α-covers in at least one sensor.

To apply this approach to solve α-RMLP with regularity condition (1), we need to take into account also the set of optimal dual prices associated with the additional constraints (5). Let us denote these multipliers $\gamma_k, k = 1, 2, \ldots, n$. Note that all the dual prices are positive once **[α-RMLP]** is considered in canonical form. The separation problem in this case is:

$$\textbf{[S2]} \qquad \min \sum_{i=1}^{m} \pi_i x_i - \sum_{k=1}^{n} \gamma_k y_k \tag{12}$$
$$\text{s.}t. \qquad (7)-(10)$$

This model differs from **[S1]** only in the objective function. Note that, because of the regularity requirement, there could exist optimal solutions that are not composed of minimal α-covers and therefore constraints (11) can not be added to **[S2]** since they might exclude meaningful α-covers. However, it is easy to show that, if the problem is feasible, there exists an optimal solution for α-RMLP composed of only target-minimal α-covers. Therefore, we could add a set of constraints that allows the generation of α-covers containing previous ones only if they cover a wider set of targets. In particular, let $C_1^\alpha, C_2^\alpha, \ldots, C_u^\alpha$ be the set of α-covers generated by the algorithm so far. We need to introduce an additional set of binary variables z_j such that $z_j = 0$ if the new α-cover C_{new}^α being built does not contain more targets than C_j^α. We add the following constraints to **[S2]**:

$$\sum_{i=1}^{m} a_{ij} x_i - z_j \le \sum_{i=1}^{m} a_{ij} - 1 \quad \forall j = 1, \ldots, u \tag{13}$$
$$(n+1)(1-z_j) - 1 \ge \sum_{k=1}^{n} b_{kj} - \sum_{k=1}^{n} y_k \quad \forall j = 1, \ldots, u \tag{14}$$
$$z_j \in \{0, 1\} \quad \forall j = 1, \ldots, u \tag{15}$$

Constraints (13)–(14) are used instead of (11). If C_{new}^α does not cover more targets than C_j^α, z_j must be equal to 0 to satisfy (14) and the related constraint in (13) is equivalent to the one in (11). Otherwise, z_j can be equal to 1 and C_{new}^α can be accepted even if it is not minimal.

If the optimal objective function value of **[S2]** is not less than 1 then the final solution has been found, otherwise the new column defined by the solution value of variables x_i and y_k is introduced into the master problem.

6 Heuristic approach

In this section we describe a greedy approach to find feasible solutions to α-MLP. We also used these solutions to initialize the CG procedure, while our method to initialize the exact approach for α-RMLP will be described in the next section. Our algorithm (α-Greedy) is based on the heuristic Greedy-MSC presented in [4] for MLP and introduces some greedy criteria aimed to increase the regularity of the heuristic solutions, which experimentally proved to improve the convergence of the exact procedure. After finding the set of α-covers, their activation times are computed by solving the mathematical formulation **[α-MLP]** restricted to this subset. An outline of the procedure is presented in Algorithm 1; in the following we analyze this outline, describe the greedy criteria and discuss the computational complexity of the procedure.

Line 1 contains the input parameters. Granularity factor $gf \in (0, 1]$ represents a fixed amount of activation time assigned to each generated α-cover during the algorithm execution. The *SENSORS* set initialized in line 2 contains the list of sensors with a residual energy greater than 0. Parameters R_{s_i} initialized in lines 3–5 represent the amount of residual energy for each sensor s_i. Line 6 checks whether the remaining sensors can be used to produce a new α-cover. The loop in lines 9–16 iteratively selects *critical* targets (targets that we want to be included in the current α-cover) and sensors with the *greatest contribution* covering them, until a new α-cover is generated. Lines 17–22 decrease the lifetime of each sensor belonging to the new α-cover by a quantity equal to gf and check whether the *SENSORS* set must be updated.

Algorithm 1 $\alpha-$Greedy algorithm

1: input: wireless network (T, S), granularity factor $gf \in (0, 1]$
2: $SENSORS = S$
3: **for** each $s_i \in SENSORS$ **do**
4: $\quad R_{s_i} = 1$
5: **end for**
6: **while** $|T_{SENSORS}| \geq T_\alpha$ **do**
7: $\quad$ Create a new empty α-cover C_l
8: $\quad TARGETS = T_{SENSORS}$
9: $\quad$ **while** $|T_{C_l}| \leq T_\alpha$ **do**
10: $\quad\quad$ Find a **critical target** $t_r \in TARGETS$
11: $\quad\quad$ Select $s_u \in SENSORS$ s.t. $t_r \in T_{s_u}$ and s_u has the **greatest contribution**
12: $\quad\quad$ **for** each $t_i \in TARGETS$ s.t. $t_i \in T_{s_u}$ **do**
13: $\quad\quad\quad TARGETS = TARGETS \setminus \{t_i\}$
14: $\quad\quad$ **end for**
15: $\quad\quad C_l = C_l \cup \{s_u\}$
16: $\quad$ **end while**
17: $\quad$ **for** each $s_i \in C_l$ **do**
18: $\quad\quad R_{s_i} = R_{s_i} - gf$
19: $\quad\quad$ **if** $R_{s_i} = 0$ **then**
20: $\quad\quad\quad SENSORS = SENSORS \setminus \{s_i\}$
21: $\quad\quad$ **end if**
22: $\quad$ **end for**
23: **end while**
24: solve **[α-MLP]** restricted to the generated covers, find activation time w_l for each C_l
25: **return** the collection of pairs (C_l, w_l)

Critical target

Cardei et al. [4] in their Greedy-MSC propose to select at each iteration the most sparsely covered target as critical (either in terms of number of sensors or their residual energy). In our algorithm instead, a critical target is the target that, among the ones which are covered by sensors s_i with $R_{s_i} > 0$, has been covered for the shortest amount of time so far. Ties are broken by selecting the one whose covering sensors have the smallest amount of residual energy.

Sensor contribution

In [4] the authors propose to use a contribution function that favors the selection of sensors with high residual energy and that cover a large number of uncovered targets. Our contribution function further addresses this second objective, by trying to favor sensors which cover targets that have been covered for small amounts of time so far. During the algorithm execution, let w_t be the coverage time so far for a generic target t and w_{max} be the maximum coverage time so far among all the targets in the network. The contribution of each sensor s with $R_s > 0$ is $Contr(s) = R_s + \sum_{t \in TARGETS | t \in T_s}(1 + w_{max} - w_t)$.

Complexity analysis

We do not consider line 24, where a linear programming model is solved by means of the simplex method. Critical target selection requires to check the residual energy of the covering sensors of each candidate target and sensor contribution evaluation considers for each sensor the coverage time so far of its targets: therefore, each of these operations require $O(nm)$ time at most. Every other operation requires $O(n)$ or $O(m)$ at most. The loop in lines 9–16 is executed $O(n)$ times at most since at each iteration at least a new target is covered. The loop in lines 6–23 is executed $O(\frac{m}{gf})$ times at most since at each iteration we lower the battery life of at least one sensor of a quantity gf. Therefore, the worst case complexity of the algorithm, without considering the computational complexity of the final step, is $O(\frac{n^2m^2}{gf})$.

7 Computational results

In this section we describe our computational experiments and the related results. All tests were performed on a workstation with Intel Core 2 Duo processor at 2.4 Ghz and 3 GB of RAM. All the procedures have been coded in C++, using IBM ILOG CPLEX 11.2 and the Concert Technology library to solve the mathematical formulations. After a preliminary experimental phase we chose a granularity factor equal to 0.25 for the heuristic algorithm and a 1-hour time limit for the exact algorithms.

Instances are divided into two groups. The instances of the Group 1 have been created according to the description given in [4]. These are random instances containing a limited number of targets ($m = 15$ targets) and a considerably higher number of sensors (we set $n = 25, 50, 100, 150$). Sensors and targets are randomly disposed on a grid with size $500\,m \times 500\,m$ and the sensing range of each sensor is $200\,m$. We generated five instances for each value of n. For experiments on Group 1, we considered $\alpha = 1, 0.85, 0.75, 0.5$, which means (by rounding up) that each α-cover must contain $T_\alpha = 15, 13, 11, 8$ targets respectively, for a total of 16 scenarios and 80 instances.

Instances in Group 2 were used in [6]. Each of these instances contains 100 targets. Instead of a predefined number of sensors, an additional $d = 3$ depth parameter has been used, representing the minimum number of sensors covering each target. Moreover, these instances are divided into two subgroups. In the *Scattering* subgroup, sensors are considered to be inexpensive and therefore randomly disposed as in the instances of Group 1. In the *Design* subgroup, sensors are installed only where needed to reach the depth. Each subgroup contains 30 instances. For experiments on Group 2, we considered $\alpha = 1, 0.99, 0.97, 0.95, 0.93, 0.85, 0.75, 0.5$ (that is, $T_\alpha = 100, 99, 97, 95, 93, 85, 75, 50$ respectively), for a total of 16 scenarios and 480 instances.

For each instance we fixed the regularity threshold w_{min} for α-RMLP equal to the optimum objective function value of MLP on the same instance (that is, the result of α-MLP when $\alpha = 1$). Therefore each target is guaranteed to be covered for at least as long as it was in the original problem.

The Column Generation applied to solve α-MLP is initialized with the solutions provided by the heuristic described in Sect. 6. On the other hand, the CG for α-RMLP is initialized using the optimal solution of MLP on the same instance. In this way, we are guaranteed to have feasible initializations due to the chosen value for parameter w_{min}.

The results of our experiments are contained in next subsections.

7.1 Heuristic results for α-MLP

We compare the results of α-Greedy with the related optimal solutions of α-MLP. Tables 1 and 2 contain average values for the instances of Group 1 and Group 2. Optimal solutions have been computed using our Column Generation with heuristic initializations.

In Group 2 five instances violated the 1-hour time limit when solved optimally (see Table 2). The averages of the related scenarios are evaluated only on the instances which run to completion for both procedures. Computational times of α-Greedy are negligible, while those of the CG are discussed in the next subsection and are not reported in these tables.

Table 1 Exact and heuristic solution comparisons for Group 1

α	T_α	Number of sensors					
		25			50		
		Heuristic	Exact	Gap %	Heuristic	Exact	Gap %
1	15	3.6	3.6	0	9.2	9.4	2.17
0.85	13	5.1	6.6	29.41	11.33	13.93	22.95
0.75	11	7.83	10.4	32.82	16.1	19.4	20.5
0.5	8	11.6	13.6	17.24	23.2	27.23	17.37
α	T_α	Number of sensors					
		100			150		
		Heuristic	Exact	Gap	Heuristic	Exact	Gap
1	15	15.4	15.4	0	25	25	0
0.85	13	23.29	30.4	30.53	37.22	51.72	38.96
0.75	11	29.95	41.49	38.53	50.59	66.98	32.4
0.5	8	44.35	54.9	23.79	76.03	87.6	15.22

Table 2 Exact and heuristic solution comparisons for Group 2

α	T_α	Instances type					
		Scattering			Design		
		Heuristic	Exact	Gap %	Heuristic	Exact	Gap %
1	100	2.87	3	4.53	2.36	3	27.12
0.99	99	3.28	3.83	16.77	2.45	3	22.45
0.97	97	3.68	5.37	45.92	2.72	3.04	11.76
0.95	95	4.24	6.64	56.60	2.76	3.34	21.01
0.93	93	4.44	7.38[a]	66.22	2.82	3.65	29.43
0.85	85	6.33	10.56[b]	66.82	3.48	4.5	29.31
0.75	75	8.66	13.36	54.27	4.27	5.42	26.93
0.5	50	16.08	20.5	27.49	6.83	8.32	21.82

[a] Scenarios where 3 instances out of 30 violated the time limit
[b] Scenarios where 2 instances out of 30 violated the time limit

In the tables, Column α reports the different values of parameter α, Column T_α is the total number of targets to be covered by each α-cover and it is such that $T_\alpha = n \times \alpha$. Columns *heuristic* and *exact* contain the average heuristic and exact solution value, respectively, computed on the instances of the same scenario. Column *gap* reports the average gap between the optimal solution and the heuristic one. The gap is computed as $\frac{(\text{exact}-\text{heuristic})}{\text{heuristic}} \times 100$. It can be noticed that the average gap can be up to 38.96% for Group 1 and up to 66.82% for Scattering instances in Group 2 (see Table 2 with $\alpha = 0.85$), therefore more complex approaches (e.g., metaheuristics) should be considered for accurate heuristic results on each scenario.

7.2 Exact results for α-MLP and α-RMLP

We compare exact solutions for both problems returned by our CG algorithms.

Tables 3 and 4 contain average solution values and average computational times. Times are expressed in seconds. In each table the row corresponding to $\alpha = 1$ gives the optimal solutions for the original problem MLP on each scenario.

In the tables, for each value of α (and the corresponding T_α) the results for α-MLP and α-RMLP are reported. Average solution values and computational times of both the problems are given in Columns *sol* and *time*. Column *gap* reports the average percentage gaps between the objective function value of the original problem MLP and of the two new problems. The gap is computed as $\frac{(\alpha\text{-MLP value}-\text{MLP value})}{\text{MLP value}} \times 100$ for the α-MLP and similarly for the α-RMLP. Hence, the greater the value of the gap, the greater the increment in the network lifetime. Finally, for each value of α (and the corresponding T_α), the values contained in Row *ratio* show average percentage gaps between α-MLP and α-RMLP, both in terms of solution values and computational times. These values give a measure of the impact of regularity requirement. The ratio is computed as $\frac{(\alpha\text{-RMLP value}-\alpha\text{-MLP value})}{\alpha\text{-MLP value}} \times 100$ to compare the objective function values and similarly for the computational times.

Table 3 Exact solutions comparisons for Group 1

α	T_α	Prob	Number of sensors					
			25			50		
			Sol	Gap %	Time	Sol	Gap %	Time
1	15	MLP	3.6		0.01	9.4		0.01
		α-MLP	6.6	83.33	0.11	13.93	48.19	0.39
0.85	13	α-RMLP	6.6	83.33	0.13	13.93	48.19	0.45
		Ratio (%)	0		18.18	0		15.38
		α-MLP	10.4	188.89	0.44	19.4	106.38	0.68
0.75	11	α-RMLP	10.4	188.89	0.22	19.4	106.38	1.01
		Ratio (%)	0		−50	0		48.53
		α-MLP	13.6	277.78	0.26	27.23	189.68	1.11
0.5	8	α-RMLP	13.6	277.78	0.46	27.23	189.68	1.79
		Ratio (%)	0		76.92	0		61.26
α	T_α	Prob	Number of sensors					
			100			150		
			Sol	Gap %	Time	Sol	Gap %	Time
1	15	MLP	15.4		0.02	25		0.02
		α-MLP	30.4	97.4	2.74	51.72	106.88	9.79
0.85	13	α-RMLP	30.4	97.4	3.33	51.72	106.88	12.31
		Ratio (%)	0		21.53	0		25.74
		α-MLP	41.49	169.42	8.03	66.98	167.92	13.90
0.75	11	α-RMLP	41.49	169.42	8.64	66.98	167.92	22.73
		Ratio (%)	0		7.6	0		63.53
		α-MLP	54.9	256.49	5.95	87.6	250.4	15.24
0.5	8	α-RMLP	54.9	256.49	8.76	87.6	250.4	22.2
		Ratio (%)	0		47.23	0		45.67

A common subset of five instances belonging to two different scenarios of Group 2 was not solved by both procedures in the 1-hour time limit, as reported in Table 4. Average values and gaps for the related scenarios are evaluated only on the instances which run to completion for all the procedures.

Let us compare the optimal value of the original MLP and of the α-MLP. Introducing α-coverage has a very high impact on the objective function value. On Group 1 (Table 3), neglecting two targets for each cover ($\alpha = 0.85$) increases the average objective function value from 48.19% (for 50 sensors) to 106.88% (for 150 sensors) with respect to the original problem. The average gap increases up to 277.78% with $\alpha = 0.5$ and 25 sensors. For every instance in Group 2 (Table 4), the MLP solution is 3. When α-coverage is taken into account, the behavior on Scattering instances is very similar to the one observed for Group 1. Neglecting one target out of 100 for each cover ($\alpha = 0.99$) already brings an average optimal solution improvement of 27.67% and the average gap increases up to 583.33% for $\alpha = 0.5$. As expected, the impact of α-coverage is not so high for Design instances but even in this case large improvements can be noticed for low enough values of α (e.g., about 50% improvement when neglecting 15 targets). Let us now compare the objective function values for α-MLP and α-RMLP. On all instances of Group 1 and Scattering instances of Group 2, the

Table 4 Exact solutions comparisons for Group 2

α	T_α	Prob.	Instances type					
			Scattering			Design		
			Sol	Gap %	Time	Sol	Gap %	Time
1	100	MLP	3		0.05	3		0.21
		α-MLP	3.83	27.67	0.56	3	0	0.59
0.99	99	α-RMLP	3.83	27.67	0.59	3	0	0.1
		Ratio (%)	0		5.36	0		−83.05
		α-MLP	5.37	79	2.01	3.04	1.33	1.43
0.97	97	α-RMLP	5.37	79	2.03	3.03	1	2.13
		Ratio (%)	0		1	−0.33		48.95
		α-MLP	6.64	121.33	8.02	3.34	11.33	2.68
0.95	95	α-RMLP	6.64	121.33	8.88	3.31	10.33	3.45
		Ratio (%)	0		10.72	−0.9		28.73
		α-MLP	7.38[a]	146	36.18[a]	3.65	21.67	7.03
0.93	93	α-RMLP	7.38[a]	146	44.1[a]	3.61	20.33	6.51
		Ratio (%)	0		21.89	−1.1		−7.4
		α-MLP	10.56[b]	252	302.91[b]	4.5	50	11.46
0.85	85	α-RMLP	10.56[b]	252	304.12[b]	4.48	49.33	15.58
		Ratio (%)	0		0.4	−0.44		35.95
		α-MLP	13.36	345.33	216.98	5.42	80.67	13.94
0.75	75	α-RMLP	13.36	345.33	222.93	5.41	80.33	12.89
		Ratio (%)	0		2.74	−0.18		−7.53
		α-MLP	20.5	583.33	11.13	8.32	177.33	3.2
0.5	50	α-RMLP	20.5	583.33	23.85	8.32	177.33	4.7
		Ratio (%)	0		114.29	0		46.88

[a] Scenarios where 3 instances out of 30 violated the time limit
[b] Scenarios where 2 instances out of 30 violated the time limit

regularity requirement has no effect on the objective function. On Design instances, slight decrements can be noticed in 5 out of 7 scenarios when regularity is imposed (highest average gap 1.1%). Overall, on the considered datasets, regularity proves to be highly acceptable.

In Tables 5 and 6, averages of the minimum and maximum coverage time for the targets are reported. Columns *gap* report the percentage gaps of these values with respect to the corresponding value of the original problem obtained when $\alpha = 1$ (where every target has the same coverage). It can be seen that regularity is always reached without being required in 10 out of 12 scenarios for Group 1 (see the values corresponding to α-MLP), and the maximum average decrement in the minimum coverage level for the remaining 2 is 1.17%. Regularity is also always reached without being imposed in 3 out of 7 scenarios for Scattering instances in Group 2, and the maximum average decrement for the remaining 4 is 3.33%. This shows that besides being a desirable condition regularity is also often a valid criterion to effectively use the sensors energy, giving numerical justification to the regularity-oriented choices we introduced in our heuristic. There are no Design scenarios where regularity is reached without being imposed, however average gaps are never higher than 21% (see $\alpha = 0.97$). Moreover, a comparison of the maximum coverage levels and the related objective function values for every scenario of every group shows that targets in fortunate conditions

Table 5 Max/Min target coverage time comparisons for Group 1

α	T_α	Prob.	Number of sensors							
			25				50			
			Min	Gap %	Max	Gap %	Min	Gap %	Max	Gap %
1	15	MLP	3.6		3.6		9.4		9.4	
0.85	13	α-MLP	3.6	0	6.6	83.33	9.29	−1.17	13.93	48.19
		α-RMLP	3.6	0	6.6	83.33	9.4	0	13.93	48.19
0.75	11	α-MLP	3.6	0	10.3	186.11	9.4	0	19.15	103.72
		α-RMLP	3.6	0	10.4	188.89	9.4	0	19.09	103.09
0.5	8	α-MLP	3.6	0	12.3	241.67	9.4	0	23.23	147.13
		α-RMLP	3.6	0	12.1	236.11	9.4	0	23.2	146.81
α	T_α	Prob.	Number of sensors							
			100				150			
			Min	Gap %	Max	Gap %	Min	Gap %	Max	Gap %
1	15	MLP	15.4		15.4		25		25	
0.85	13	α-MLP	15.4	0	30.4	97.4	24.94	−0.24	51.72	106.88
		α-RMLP	15.4	0	30.4	97.4	25	0	51.72	106.88
0.75	11	α-MLP	15.4	0	40.46	162.73	25	0	66.17	164.68
		α-RMLP	15.4	0	40.71	164.35	25	0	65.78	163.12
0.5	8	α-MLP	15.4	0	46.19	199.94	25	0	76.1	204.4
		α-RMLP	15.4	0	45.92	198.18	25	0	75.6	202.4

Table 6 Max/Min target coverage time comparisons for Group 2

α	T_α	Prob.	Instances type							
			Scattering				Design			
			Min	Gap %	Max	Gap %	Min	Gap %	Max	Gap %
1	100	MLP	3		3		3		3	
0.99	99	α-MLP	2.9	−3.33	3.83	27.67	2.53	−15.67	3	0
		α-RMLP	3	0	3.83	27.67	3	0	3	0
0.97	97	α-MLP	2.91	−3	5.37	79	2.37	−21	3.04	1.33
		α-RMLP	3	0	5.37	79	3	0	3.03	1
0.95	95	α-MLP	2.99	−0.33	6.64	121.33	2.62	−12.67	3.34	11.33
		α-RMLP	3	0	6.64	121.33	3	0	3.31	10.33
0.93	93	α-MLP	2.99[a]	−0.33	7.38[a]	146	2.69	−10.33	3.65	21.67
		α-RMLP	3[a]	0	7.38[a]	146	3	0	3.61	20.33
0.85	85	α-MLP	3[b]	0	10.56[b]	252	2.9	−3.33	4.5	50
		α-RMLP	3[b]	0	10.56[b]	252	3	0	4.48	49.33
0.75	75	α-MLP	3	0	13.36	345.33	2.86	−4.67	5.42	80.67
		α-RMLP	3	0	13.36	345.33	3	0	5.41	80.33
0.50	50	α-MLP	3	0	18.15	505	2.91	−3	7.14	138
		α-RMLP	3	0	18.21	507	3	0	7.13	137.67

[a] Scenarios where 3 instances out of 30 violated the time limit
[b] Scenarios where 2 instances out of 30 violated the time limit

are generally covered for most of the network lifetime. Finally, go back to Tables 3 and 4 to evaluate the computational times of the algorithms. The α-RMLP procedure performs on average worse than the α-MLP algorithm in 11 out of 12 scenarios in Group 1 and in 11 out of 14 scenarios in Group 2. This can be explained with the generally weaker initialization, the additional complexity of the problem and the smaller number of α-covers which can be discarded during the execution. However, on more difficult instances (those where computational times increase) the gap decreases. Indeed, considering the two scenarios which require on average more than 200 s (see the Scattering instances with $\alpha = 0.75$ in Table 4) the maximum gap is 2.74%, and in the single scenario which requires on average more than 300 s (see the Scattering instances with $\alpha = 0.85$) it is 0.4%. Overall, once the five instances which proved to be difficult to solve are removed, the highest average computational time is 302.91 s.

8 Conclusions

In this paper we introduced two variants of the well known MLP where partial coverage of the targets is allowed. The two resulting problems are the Maximum Network α-Lifetime Problem and the Regular Maximum Network α-Lifetime Problem. We analyzed their complexity and provided a Column Generation exact approach and a heuristic algorithm. We tested our approaches on a wide set of benchmark instances to better understand the trade-off between the network lifetime and the coverage percentage. Our experiments show that, with a very small percentage of neglected targets not only the network lifetime is hugely improved, but also, the total coverage time of the targets is increased. That is, the overall performance of the monitoring activity of the sensor network is largely improved by considering α-covers instead of covers. Further steps of research are focused on the analysis of the performance of the sensor network when different measures of regularity are considered. Moreover, more efficient metaheuristic approaches will also be investigated to solve the problem on larger instances in reasonable time.

References

1. Berman, P., Calinescu, G., Shah, C., Zelikovsky, A.: Power Efficient Monitoring Management in Sensor Networks. In: Proceedings of the Wireless Communications and Networking Conference ’04, pp. 2329–2334 (2004)
2. Boginski, V.L., Commander, C., Pardalos, P.M., Ye, Y. (ed.): Sensors: Theory, Algorithms, and Applications. Springer Optimization and Its Applications, vol. **61** (2011)
3. Cardei, M., Du, D.-Z.: Improving wireless sensor network lifetime through power-aware organization. ACM Wirel. Netw. **11**(3), 333–340 (2005)
4. Cardei, M., Thai, M.T., Li, Y., Wu, W.: Energy-Efficient Target Coverage in Wireless Sensor Networks. In: Proceedings of the 24th conference of the IEEE Communications Society (INFOCOM), vol. **3**, pp. 1976–1984 (2005)
5. Cheng, M.X., Ruan, L., Wu, W.: Achieving Minimum Coverage Breach under Bandwidth Constraints in Wireless Sensor Networks. In: Proceedings of the 24th conference of the IEEE Communications Society (INFOCOM), vol. **4**, pp. 2638–2645 (2005)
6. Lopes, L., Gentili, M., Efrat, A., Ramasubramanian, S.: Scheduling Redundant Sensors Optimally for Maximum Lifetime, Technical Report n.11-2010, Department of Mathematics and Computer Science, University of Salerno (2010)

7. Slijepcevic, S., Potkonjak, M.: Power efficient organization of wireless sensor networks. IEEE Int. Conf. Commun. **2**, 472–476 (2001)
8. Wang, C., Thai, M.T., Li, Y., Wang, F., Wu, W.: Optimization scheme for sensor coverage scheduling with bandwidth constraints. Optim. Lett. **3**(1), 63–75 (2009)
9. Wu, W., Gao, X., Pardalos, P.M., Du, D.-Z.: Wireless networking, dominating and packing. Optim. Lett. **4**(3), 347–358 (2010)
10. Zhang, H., Hou, J.C.: Maximizing α-lifetime for wireless sensor networks. Int. J. Sens. Netw. **1**(1), 64–71 (2006)