An exact formulation and a hybrid heuristic method for a Hop-constrained WSN using a Delay-constrained Mobile Agent

Oberlan Christo Romão
Departamento de Informática – Universidade Federal de Viçosa
Av. P.H. Rolfs, s/n, DPI - Campus UFV - 36570-000, Viçosa – MG
`oberlan.romao@ufv.br`

André Gustavo dos Santos
Departamento de Informática – Universidade Federal de Viçosa
Av. P.H. Rolfs, s/n, DPI - Campus UFV - 36570-000, Viçosa – MG
`andre@dpi.ufv.br`

ABSTRACT

Wireless Sensor Networks (WSNs) have emerged as an attractive and challenging research field. One of the main challenges lies in the constrained energy resources available to sensor nodes. A possible way to save energy is to use a mobile agent to collect the data moving through the WSN, but this approach increases the messages delivery delay. In this work we propose a Mixed-Integer Linear Programming (MILP) formulation to model the problem as a communication forest linked by a mobile agent route, where the roots of the trees are cluster heads sensors visited by the mobile agent; the other sensors transmit their information to the cluster heads using one or more hops. We also introduce a hybrid approach using GRASP to select the cluster heads, and constructive heuristics with local search to define the communication forest and the mobile agent route, in order to minimize the total energy consumption within a limited delay. Results are presented for WSN with up to 80 sensors with different limits for the mobile agent route length, to control the delivery delay of messages. The optimality of instances with 20 and 30 nodes were confirmed by solving the MILP formulation.

KEYWORDS: Wireless Sensor Networks, Mobile Agent, GRASP, Combinatorial Optimization.
Main area: MH - Metaheuristics, OC - Combinatorial Optimization.

1. Introduction

A wireless sensor network (WSN) consists of spatially distributed sensors nodes that are deployed in a region of interest. The sensors are capable of collecting information from the environment, such as temperature, vibration, sound, motion, pressure, etc. and are able to communicate with each other via wireless if they are within their mutual communication range [19]. The data collected should then be transmitted to a *Base Station (BS)*, where all information collected is analyzed.

The major limitation of WSNs is the low energy capacities of the sensors [11]. Since the sensors are usually deployed in inhospitable environments in large quantities, it is difficult or impossible to replace or recharge the batteries. A possible solution to save energy is to allow a *Mobile Agent (MA)* to move through the WSN to collect the data. However, this approach dramatically increases the message delay rates [16]. Therefore, the construction and operation of a WSN should be carefully planned in order to increase the network lifetime and also decrease the information delivery delay. The network lifetime is the period of time that the network is able to perform its intended operations, and may be defined in different ways, for example the time until the first sensor in the WSN runs out of battery [4, 11, 18]. The information delivery delay is the elapsed time between the generation of a information by a sensor and the moment it is received by the *BS* [2].

The network considered in this work contains one mobile agent and several randomly deployed sensor nodes. It is assumed that the mobile agent knows the geographical location of the sensor nodes after their deployment in the sensing field. The mobile agent is rich in energy and physically travels through the network visiting only a subset of sensors, known as *Cluster Heads (CH)*. They can store data temporarily and transmit it when the mobile agent is in their communication range. Only cluster heads communicate with the MA directly and the collected information by other sensors must be forwarded to one of the cluster heads using one or multi hops, i.e., directly or indirectly via other sensors. The cluster head in turn delivers its data and the data received from other sensors to the mobile agent. This sets a balance between the network lifetime and delivery delay: sensors located far from the base station deliver the data collected to the a cluster head, thus decreasing the energy consumption; the mobile agent visits only the cluster heads, not all sensors, thus decreasing the delivery delay.

Sensors may deliver their information to the cluster heads using a limited number H of hops, thus defining a hop-constrained tree whose root is a cluster head. As each cluster head defines a tree, we have a communication forest and a mobile agent that visits the roots of each tree.

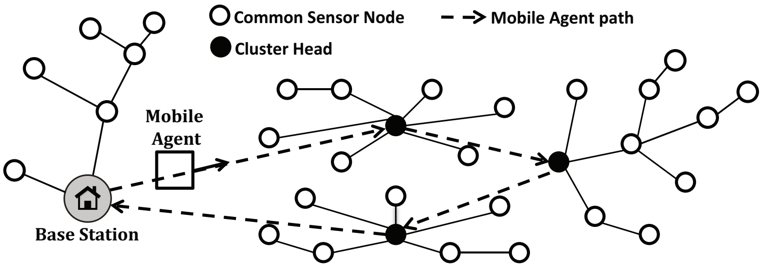

Figure 1. An example of a WSN, with communication limited to three hops

A solution to the problem consists of: (i) the set of cluster heads, (ii) the communication forest, and (iii) the mobile agent route. The objective is to find a hop-and-delay constrained solution with minimal energy consumption: one whose communication forest is H-hop-constrained and whose mobile agent route is within a limited length. Notice that if the MA route length is limited to zero, the problem reduces to the *Hop Constrained Minimal Spanning Tree (HMST)* problem [8], a NP-hard problem. On the other hand, if $H = 0$, the problem reduces to the *Traveling Salesman Problem (TSP)*, which is NP-hard too. In the proposed hybrid algorithm the first part of the solution (set of cluster heads) is defined by a GRASP algorithm, and the others are defined by greedy constructive heuristics followed by local search.

The remainder of this paper is organized as follows: in Section 2 we review previously proposed solutions that exploit the mobility on the WSNs. In Section 3 we present the energy model of the WSN proposed. In Section 4 an Mixed Integer Linear Programming (MILP) for the problem is proposed while in Section 5 we detail each part of the hybrid algorithm; Section 6 analyzes the results from different scenarios, and finally, Section 7 concludes the paper and points out some future research directions.

2. Related Works

Maximizing the lifetime of wireless sensor network with mobile agent has received a notable attention in research. Some papers study networks where sensors transmit the messages through the network to the base station using multi-hops, without mobile agent ([10, 9, 14, 17]). As the data communication in WSN with multi-hops is the primal responsible for the energy consumption [12], several works from literature, like [1, 3, 4, 5, 11, 12, 15, 16], propose a communication approach using a controlled number of hops. Then, the use of mobile agents becomes necessary, since some sensors may not be able to transmit their data to the base station.

Keskin et al. [11] propose two mathematical programming formulations, one limiting the number of hops and other without limit. But the mobile agent can visit only a small number of predefined locations. An architecture for a traffic surveillance application is proposed in [15], where a police car is the mobile agent which travels the highway for collecting the data. However, it is also assumed that all sensors must communicate using one-hop with the mobile agent.

Aioffi et al. [1] propose the *MHS-*λ (*Multi-Hop Strategy*) consisting in two phases, solved separately. Firstly, the sensors are clustered and a cluster head is chosen for each group; the other sensors send their data to the cluster head of their group, using limited number of hops λ. Then, a mobile agent visits each cluster head by the shortest route. However, the messages delivery delay in is not considered. To address this, our approach solves the two phases together.

Bechelane et al. [3] propose the *Minimum Cost Hop-and-root Constrained Forest Problem* (*MCFP*), a mathematical model to minimize the energy consumption so that the MA route length is at most D_{max}. However, each sensor sends only one data packet, which may be not feasible in some scenario, like military application, in which all information are important. Besides that, they consider energy consumption only on transmission. In our work we assume that the energy consumption is proportional to the amount of data transmitted and received. As demonstrated by [7], limiting the MA tour length may result in less energy efficiency, then we tested different limits to analyze the impact.

3. Energy Model

As mentioned before, in this work the energy consumption is proportional to the amount of data transmitted and received, which is typical for the mathematical optimization models proposed for message routing in WSNs [11]. We also assume that sensors are awake only while transmitting or receiving data sensing and remain in the standby mode otherwise (in this case we do not regard the energy consumption); sensor 1 is also the base station (BS), as shown in Figure 1, which is assumed not to be energy constrained. Furthermore, a sensor can communicate with another one if the distance between them is within their transmissions ranges (R^c), which is the same for all sensors. It is assumed that the MA can move freely from one point to another using the minimal distance, like a flying device traveling above the sensing field at a fixed height (D_{MA}), which is the transmission distance used by the cluster heads.

Generally, the energy metric is composed of two parts: data transfer energy (Et) and data receive energy (Er).The communication cost between two sensors usually depends on the distance d between them, the message size k, and a factor f which varies according to the characteristics of the environment. Each sensor (except the base station) collects k bits from the sensing field and sends data directly to one other sensor or to the mobile agent, as is the case of the cluster heads. Thus, whether a sensor i transmits Q_i packets, it has received $Q_i - 1$ packets from other sensors, and adds its own data before passing ahead the received data. Then, the estimate the total E_c of energy consumed, based on [9], can be calculated with the following set of equations:

$$Et_i = Q_i k (E_{elec} + \epsilon_{amp} d_{ij}^f) \tag{1}$$

$$Er_i = (Q_i - 1) k E_{elec} \tag{2}$$

$$E_c = \sum_{i \in \mathcal{S} \setminus \{1\}} (Et_i + Er_i) \tag{3}$$

where: Et_i is the energy necessary to sensor i transmit Q_i packets with k bits to sensor j at a distance d_{ij}; Er_i is the energy consumed by sensor i to receive Q_i packets with k bits; E_{elec} is a constant related to radio electronics to send or receive data; ϵ_{amp} is a constant associated with the characteristics of the environment; and $\mathcal{S}$ is the set of sensors.

Note that total consumption does not include the energy consumption of the base station, as its energy is not constrained, i.e., technically unbounded. Like [9], we consider $E_{elec} = 50$nJ/bit, $\epsilon_{amp} = 100$pJ/bit/m^2 and $f = 2$.

4. Problem Formulation

We propose a Mixed Integer Linear Programming (MILP) formulation, based on [1] and [3], in which the data is sent through paths with lowest energy consumption from sensors to the MA.

In order to model the WSN, consider a graph $D = (V, E)$, where the vertices in $V = \{1, 2, \dots, n\}$ represent the sensors nodes, and E is a set of edges where $(i, j) \in E$ if sensors i and j can communicate directly, i.e., if they are located within their communication range. To formulate the problem as a MILP, let $\overline{D} = (V \cup 0, \overline{E})$ be a graph obtained from $D = (V, E)$ by adding (i) an artificial vertex 0 and (ii) artificial edges $\{\{i, 0\} : i \in V\}$, with $d_{i0} = D_{MA}$. The H-hop-constrained forest in D can be mapped into a (*H*+*1*)-hop-constrained tree in $\overline{D}$, rooted at 0. The graph $\overline{D}$ together with $Et_{ij}, i, j \in V$ and $Er_i, i \in V$ defines the *energy network*. Graph $D = (V, E)$, together with distances $\{d_{ij} : i, j \in V\}$, defines the *translation network*.

Let $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ be the set of sensors and $\mathcal{S}' = \mathcal{S} \cup \{0\}$, where 0 is an artificial sensor. The proposed formulation is a mixed integer formulation, and uses the following variables:

- y_i: binary variable, indicating whether the sensor $i \in \mathcal{S}$ is or not a cluster head;
- w_{ij}: binary variable, indicating whether the cluster head j is visited right after cluster head i or not;
- z_{ij}: binary variable, indicating if the arc $(i,j) \in \overline{E}$ belongs to the (*H*+*1*)-hop-constrained tree or not;
- x^s_{ij}: amount of a commodity $s \in \mathcal{S}$ flowing in the translation network through arc $(i,j) \in \mathcal{S}$;
- f^s_{ij}: quantity of a commodity $s \in \mathcal{S}$ flowing in the energy network through arc $(i,j) \in \mathcal{S}'$;
- Q_i: Number of packets transmitted by sensor $i \in \mathcal{S}$;

As the estimate of energy consumption, the formulation uses equations (1)-(3), but variable z_{ij} is included in (1) resulting in (4):

$$Et_i = Q_i k(E_{elec} + \epsilon_{amp} z_{ij} d^f_{ij}), \qquad \forall i \in \mathcal{S}\backslash\{1\} \tag{4}$$

Then, the complete formulation is given by equations (2)-(4) and the following objective function (5) and set of constraints (6)-(27).

$$\min E_c \tag{5}$$

$$\sum_{i \in \mathcal{S}\backslash\{1\}} x^s_{1i} = y_s, \qquad \forall s \in \mathcal{S}\backslash\{1\} \tag{6}$$

$$\sum_{j \in \mathcal{S}} x^i_{ij} - \sum_{j \in \mathcal{S}} x^i_{ji} = -y_i, \qquad \forall i \in \mathcal{S}\backslash\{1\} \tag{7}$$

$$\sum_{j \in \mathcal{S}} x^v_{ij} - \sum_{j \in \mathcal{S}} x^v_{ji} = 0, \qquad \forall i, v \in \mathcal{S}\backslash\{1\}, i \neq v \tag{8}$$

$$x^s_{ij} \leq w_{ij}, \qquad \forall i, j \in \mathcal{S}, s \in \mathcal{S}\backslash\{1\} \tag{9}$$

$$w_{ij} \leq y_i, \qquad \forall i, j \in \mathcal{S} \tag{10}$$

$$w_{ij} \leq y_j, \qquad \forall i, j \in \mathcal{S} \tag{11}$$

$$\sum_{i,j \in \mathcal{S}} w_{ij} d_{ij} \leq D_{max} \tag{12}$$

$$\sum_{j \in \mathcal{S}\backslash\{1\}} w_{ij} \leq 1 \tag{13}$$

$$\sum_{j \in \mathcal{S}\backslash\{1\}} w_{1j} = \sum_{j \in \mathcal{S}\backslash\{1\}} w_{j1} \tag{14}$$

$$y_1 = 1 \tag{15}$$

$$\sum_{j \in \mathcal{S}, j \neq i} w_{ij} \leq y_i \qquad \forall i \in \mathcal{S}\backslash\{1\} \tag{16}$$

$$\sum_{j \in \mathcal{S}, j \neq i} w_{ji} = y_i \qquad \forall i \in \mathcal{S}\backslash\{1\} \tag{17}$$

$$\sum_{i,j \in \mathcal{S}'} z_{ij} = n \tag{18}$$

$$\sum_{j \in \mathcal{S}} f^q_{0j} = 1 \qquad \forall q \in \mathcal{S} \tag{19}$$

$$\sum_{j \in \mathcal{S}'} f^q_{ij} - \sum_{j \in \mathcal{S}'} f^q_{ji} = 0 \qquad \forall i, q \in \mathcal{S}, i \neq q \tag{20}$$

$$\sum_{j \in \mathcal{S}'} f^i_{ij} - \sum_{j \in \mathcal{S}'} f^i_{ji} = -1 \qquad \forall i \in \mathcal{S} \tag{21}$$

$$f^q_{ij} \leq z_{ij} \qquad \forall q \in \mathcal{S}, \forall i, j \in \mathcal{S}' \tag{22}$$

$$\sum_{i,j \in \mathcal{S}'} f^q_{ij} \leq H + 1 \qquad \forall q \in \mathcal{S} \tag{23}$$

$$\sum_{i \in \mathcal{S}} z_{ij} + y_j = 1 \qquad \forall j \in \mathcal{S} \tag{24}$$

$$z_{0i} = y_i \qquad \forall i \in \mathcal{S} \tag{25}$$

$$Q_s = \sum_{i,j \in \mathcal{S}} f^j_{si} + 1 \qquad \forall s \in \mathcal{S} \tag{26}$$

$$\begin{array}{lr} x_{ij} \in \mathbb{R}^+, & \forall i, j \in \mathcal{S} \\ z_{ij} \in \mathbb{B}, & \forall i, j \in \mathcal{S}' \\ y_i \in \mathbb{B}, & \forall i \in \mathcal{S} \\ f^q_{ij} \in \mathbb{R}^+, & \forall i, j \in \mathcal{S}', \forall q \in \mathcal{S} \\ w_{ij} \in \mathbb{B}, & \forall i, j \in \mathcal{S} \\ Q_i \in \mathbb{N}, & \forall i \in \mathcal{S} \end{array} \tag{27}$$

In the above formulation, the problem of defining the MA route is represented by constraints (6)-(17). These constraints represent the *Minimum Cost Flow Problem* defined on the network $D = (V, E)$ and distance $\{d_{ij} : i, j \in V\}$. The set of constraints (6) ensure a flow from the vertex 1 to each cluster head s. Constraints (7) and (8) ensure flow conservation until the cluster head is reached. The constraints (9)-(11) ensures that the trajectory of the mobile agent visits each cluster head. Constraint (12) limit the mobile agent route length. The constraints (13)-(15) guarantee that the route, if it exists, starts and ends at vertex 1, the *BS*. Finally, the constraints (16) and (17) force the MA to visit only the cluster heads.

The (*H*+*1*)-hop-constrained tree is defined by constraints (18)-(25), based on [8]. The equality (18) defines the total number of edges used in the energy network. Constraints (19) guarantee that the tree is rooted in artificial node 0. The constraints (20) and (21) assure flow balance. Inequalities (22) link variables z and f, while the constraints (23) ensure there is no more than *H*+*1* hops between any vertex $i \in \mathcal{S}$ and the vertex 0 in the communication tree. From constraints (24) and (25) no arc can be incident to cluster heads, except arcs coming from the artificial node 0. Finally, the constraints (26) define the number of packets transmitted by sensor $s \in \mathcal{S}$ and the set of constraints (27) define the type of the decision variables.

The formulation is in fact nonlinear because binary variables z_{ij} are multiplied by continuous variables Q_i in (4). The model was then linearized by defining variables $\delta_{ij} = Q_i z_{ij}$ and assuring their value properly according to z_{ij}.

5. Proposed hybrid method

As mentioned before, a good solution for the problem is composed by (i) a good set of cluster heads, (ii) a good communication forest, and (iii) a good mobile agent route. The first two are related to minimizing the energy consumption and the last (together with (i)) is related to minimizing the delivery delay. It is easy to see that all three are related to each other. The mobile agent route visits the cluster heads and the communication forest is built using cluster heads as root of each subtree. Then, to define the route and the forest, one must know which sensors are selected as cluster heads. On the other hand, cluster heads should be carefully selected in order to have good communication forest (for matters of energy consumption) and good mobile agent route (for matters of delivery delay).

We propose methods to solve each subproblem heuristically, but in a cooperative way. The first one, the selection of cluster heads, is solved by a *GRASP* metaheuristic. Both construction phase and local search phase is focused on the selection of cluster heads. For each of the other two subproblems, definition of communication forest and mobile agent route, a greedy constructive heuristic with local search is used. They work cooperatively in the following way: each potential solution (set of cluster heads) generated during the iterations of GRASP must be evaluated; the value of a solution depends on the energy consumed and on the delivery delay, and these values are calculated by the greedy heuristics.

The methods for the subproblems are to published in [13] and are detailed below.

5.1. GRASP

The meta-heuristic *GRASP (Greedy Randomized Adaptive Search Procedure)*, proposed in [6], is a multi-start iterative process, whose each iteration consists of two phases: the construction of a greedy randomized solution and a local search procedure that aims to improve the solution generated in the previous phase. Each GRASP iteration works independently and the final result is the best solution found among all iterations. The pseudocode in Algorthm 1 illustrates the main blocks of a basic GRASP, in which $maxIteration$ iterations are executed. The construction phase, including the parameter α, is detailed in the following. In the sequence, the local search procedure is detailed.

5.1.1. Construction phase

The GRASP construction phase proposed is an iterative process in which each sensor which is not cluster head is evaluated by constructive heuristics (detailed in the following) that estimates the gain to turn the sensor a cluster head. Whether any sensor is disconnected of the communication trees the solution receives a penalty, that is the number

Algorithm 1 Basic GRASP pseudo-code

```
procedure GRASP($\alpha, maxIteration$)
    for $i \leftarrow 1, maxIteration$ do
        $Solution \leftarrow$ GreedyRandomizedConstruction($\alpha$);
        $Solution \leftarrow$ LocalSearch($Solution$);
        UpdateSolution($Solution$,$BestSolution$);
    end for
    return $BestSolution$
end procedure
```

of disconnected sensors multiplied by big constant. The sensors are sorted in descending order according to the value obtained in the previous step and put into a *candidate list (CL)*. Through a factor α ($0 < \alpha \leq 1$) a *restricted candidate list (RCL)* is made that has the $max(\alpha \times |CL|, 1)$ best elements of *CL*. Note that if α is close to 1 the solution will be totally random. On the other hand, whether α is close to 0 the solution will be greedy. The next sensor to be cluster head in the partial solution is randomly selected from those in the *RCL*. This phase is done while the route length among the cluster heads selected is less than D_{max}.

Generally this solution is not a local minima, so a local search phase tries to improve the constructed solution until no better solution is found.

5.1.2. Local Search for Cluster Heads (LS_{CH})

The *Local Search (LS)* goal is to improve the solution. Local search algorithms start from a given solution and then iteratively moves to a neighbor solution. In this work we introduce, in fact, three specific neighborhoods, one for the cluster heads (LS_{CH}), another for the mobile agent route (LS_{MA}), and one for the communication trees (LS_{CT}). Only the first, LS_{CH}, is used explicitly in GRASP, the others are used only to better evaluate a solution, in the constructive heuristic for the other subproblems.

This local search is divided in two parts: (i) swap and (ii) change. The first checks whether the exchange of the cluster head status of two sensors (one becoming cluster head an other becoming a normal sensor) improves the quality of the solution. In order to reduce the neighborhood size, a cluster head $c \in \mathcal{S}$ is exchanged with a sensor $s \in \mathcal{S}$ only if $d_{cs} \leq R^c$. If no improvement can be made anymore with the first, the second tries to change the status of a single sensor i.e., whether a sensor is cluster head it becomes an ordinary sensor (non cluster head) and vice-versa. Both enumerates the whole neighborhood and the best neighbor updates the solution. Once no improvement can be made by a single movement the current solution is returned.

5.2. Constructive heuristic for solution evaluation

Every solution generated during the GRASP metaheuristic must be evaluated, being it a partial solution generated in the Constructive phase or a complete solution generated during the Local Search phase. The evaluation is done in two phases: the first calculates the energy consumption, the other one calculates the mobile agent route length, both based on the set cluster heads choice. For both we propose a constructive greedy heuristic followed by a local search.

In order to build the communication trees to measure the energy consumption with up to H hops, the following procedure is used: the non cluster head sensors that can be connected to a cluster head are connected to the closest cluster head and receive label 1. Then the still disconnected sensors which can be connected to any sensor with label 1 receive label 2. The process continues until label H. If all sensors are connected to a tree we have a feasible communication forest. Otherwise, the solution is unfeasible and is penalized by the number of disconnected sensors.

Having built the communication trees the energy consumption of the network can be calculated using the equations (1)-(3). However, the greedy constructive heuristic proposed may not build the best tree in terms of energy consumption, so a local search (detailed in the following) is proposed to try to improve the energy consumption.

The second phase computes the mobile agent route length, which is just the shortest route that visits all cluster heads, i.e., a solution for a *TSP* problem. As the number of cluster heads, compared to the number of sensors in the network, is usually small, this phase can be solved exactly. However, it must be calculated for ever solution generated during GRASP. To reduce the time to repeatedly solve this part, a TSP tour considering all sensors as cluster heads is pre-computed. For a given set of cluster heads, one possible route is the pre-computed TSP route without the common sensors (which are not cluster heads). For example, if the solution of TSP for a WSN with 5 sensors is 1→5→3→4→2→1 and the cluster heads are 1, 3 and 5, firstly we consider the route 1→5→3→1. As there is no guarantee this process generates the shortest route for every subset, a simple and fast local search (detailed in the following) is applied. If the final route length is greater than D_{max}, the solution is marked as unfeasible and penalized.

5.2.1. Local Search for the Communication Trees (LS_{CT})

This local search tries to reduce the energy consumption of the hop-and-root constrained forest found. A movement (neighbor) is the exchange of one existing link (u, v) by (u, w), with $w \neq u, d_{u,w} \leq R^c$, i.e., a sensor u is linked to another sensor within its transmission range. This exchange is considered only if sensor u is still connected to a cluster head and the resulting forest does not violate the hop constraint.

The neighborhood is then composed of all feasible forests attained by exchanging one of existing links. The best feasible neighbor is chosen, and the process continues until no improvement can be made by a single movement.

5.2.2. Local Search for the Mobile Agent (LS_{MA})

With the purpose of reducing the length of the MA route we use the *2-Opt* neighborhood, that exchanges two of the edges in the route by two other edges. More precisely, in each improving step two edges $[u, v]$ and $[w, x]$ are selected from the route, and exchanged by $[u, w]$ and $[v, x]$. For example, the route $\cdots \to u \to v \to \cdots \to w \to x \to \cdots$, such that u, v, w, x appear in this order in the route and are distinct is changed to $\cdots \to u \to w \to \cdots \to v \to x \to \cdots$, provided that this change decreases the length of the route. Notice that the sensors between v and w would now be visited in opposite direction.

5.3. Integration of the proposed methods

The proposed hybrid GRASP procedure is summarized as pseudo-code in Algorithm 2. The stopping criterion is the maximum number of iterations ($maxIteration$). The Algorithm 3 illustrates the composition of constructive heuristics and local searches to evaluate a solution.

6. Computational Results

In this section we discuss the results of the methods proposed. All algorithms were implemented in C++ and compiled with GNU g++ version 4.3.4, and all experiments were

Algorithm 2 Final GRASP pseudo-code

```
procedure GRASP($n, d_{n \times n}, H, D_{max}, maxIteration, \alpha$)
    for $i \leftarrow 1, maxItetation$ do
        $CHSelected \leftarrow$ GreedyRandomizedConstruction($\alpha$);
        $CHSelected \leftarrow$ LS$_{CH}$($CHSelected$);
        if Evaluation($CHSelected, d_{n \times n}, H, D_{max}$) < Evaluation($BestSolution, d_{n \times n}, H, D_{max}$) then
            $BestSolution \leftarrow CHSelected$
        end if
    end for
    return $BestSolution$
end procedure
```

Algorithm 3 Evaluation pseudo-code

```
procedure EVALUATION($CHSelected, d_{n \times n}, H, D_{max}$)
    $energyTree \leftarrow$ ConstructiveHeuristic($CHSelected$);
    $energyTree \leftarrow$ LS$_{CT}$($energyTree$);
    $energyTree \leftarrow$ LS$_{MA}$($energyTree$);
    return energyConsumed($energyTree$)
end procedure
```

performed on a Intel Xeon machine, with 24 GB of RAM, running at 2.67GHz, under Linux Operating System. The MILP formulation was solved using ILOG CPLEX 12.5.

We could not find results for this problem in the literature, then new instances were generated. Table 1 lists the parameters used. The position of the sensors nodes, including the base station, were randomly generated over the network area. Seven instances, with different number of sensors, were generated. Each one was tested with three different route length limits, totalizing 21 test cases.

Table 1. Instance parameters

Description	**Value**
Network area	$500 \times 500m^2$
Number of sensor nodes	{20, 30, 40, 50, 60, 70, 80}
MA route length limit (D_{max})	{1200, 1700, 2500}
Radio transmission range (R^c)	150m
Maximum number of hops (H)	3
Distance of the MA over the field (D_{MA})	50m
Message size (K)	80bits

In order to define the GRASP parameters (α and $maxIteration$) we performed some statistical tests, using the ANOVA, with several instances. The results are summarized in Figure 2, that shows the average energy consumption for the set of instances for several combination of the parameters. Here, `Ixxx_Ayy` means $maxIteration$ =`xxx` and α =`yy` (in percent). As can be seen, although not statistically different, regardless the value of $maxIteration$, the best results were obtained with α (in percent) equals to 20%. The difference among `I150_A20` and `I200_A20` is quite subtle, but the time to run 150 iterations is smaller than 200 iterations. For these reasons we choose $\alpha = 20\%$ and $maxIteration = 150$ to be used in the GRASP algorithm.

The MILP formulation was able to find optimal solutions only for the smaller instances, with 20 sensors. For instances with more than 30 sensors not even a feasible solution was found within 12 hours set for time limit. However, when providing the solution found by the method proposed as an initial solution to MILP, we could prove the optmality for instances with 30 sensors.

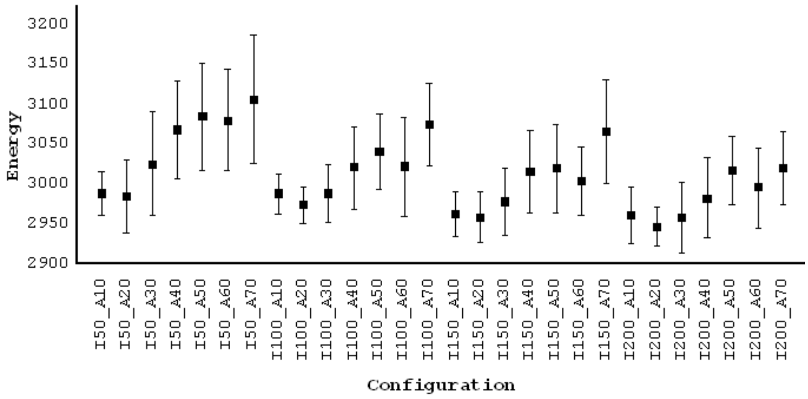

Figure 2. Result of ANOVA test for different configurations

In Table 2 we show the results attained by the proposed hybrid GRASP algorithm, and the improvement MILP could reach upon this solution. The first three columns lists the number of sensors, the MA route length limit (D_{max}) and the name of the instance, which is named using the data from the previous columns. The next two columns shows the results of the algorithm: running time in seconds and energy consumption of the best solution found. Solutions marked with '*' are proven optimal. The last four columns show the MILP results: running time (wall clock) in seconds, best solution found (using the GRASP solution as start point), percentage improvement over the start solution provided and dualilty gap reported by CPLEX.

Firstly notice that for the set of smaller instances (20 and 30 sensors), the hybrid GRASP algorithm found 5 out of 6 optimal solutions. The optimality in these cases were proven by MILP, reaching a 0% duality gap without any improvement on the solution. This suggests that the proposed GRASP and constructive heuristics are able to identify good (and optimal) solutions, the local searches used are able to find local (or global) minimum if the search space is not huge. Note that for instance `N30-D1700` the optimal solution was found in less than 3 seconds by GRASP, but MILP used more than 11 hours to prove its optimality. And for instance `N30-D1200` a solution was found by GRASP in less than 2 seconds, and no improvement was made by MILP in 12 hours.

For medium-sized instances (40 and 50 sensors) no optimal was proven. Nevertheless the improvement of MILP of the solution found by GRASP was very small, reaching a maximum of 2.4%. For large-sized instances (from 60 to 80 sensors), we cannot attest the quality of the GRASP solutions, as the MIP duality gap was generally above 60%, but the method are still very useful. The running time was very low, ranging from 1.5 to 13 minutes, while MILP could not make any improvement after 12 hours. In fact, without the solution found by GRASP, no solution would be found by the MILP formulation.

7. Conclusions and Future Work

Here we deal with a wireless sensor network problem with a mobile agent where the total energy consumption must be minimized, while the communication is hop-constrained and the MA route is length constrained. A solution has three parts: the set of cluster heads, the communication forest and the MA route. For the first we propose a GRASP metaheuristic and for the other two we propose constructive with local search. The three methods works together, cooperatively. We also propose a MILP to model the problem.

We test the proposed approach in several test cases. The computational results show

Table 2. Computational results

#Sensors	D_{max}	Instance	GRASP + heuristics		MILP Solution			
			Time (seg)	Energy (μJ)	Time (seg)	Energy (μJ)	improv.	Gap
20	1200	`N20-D1200`	0.28	1513.20 *	234.19	1513.20	–	–
	1700	`N20-D1700`	0.44	650.86 *	144.07	650.86	–	–
	2500	`N20-D2500`	0.50	441.54 *	20.91	441.54	–	–
30	1200	`N30-D1200`	1.71	1577.47	†	1577.47	–	15.89%
	1700	`N30-D1700`	2.91	1007.50 *	38890.64	1007.50	–	–
	2500	`N30-D2500`	3.63	696.00 *	2957.03	696.00	–	–
40	1200	`N40-D1200`	6.63	1964.63	†	1918.61	2.40%	30.77%
	1700	`N40-D1700`	11.95	1406.50	†	1406.50	–	24.66%
	2500	`N40-D2500`	19.22	940.78	†	940.18	0.06%	7.40%
50	1200	`N50-D1200`	40.87	2950.23	†	2922.10	0.96%	51.53%
	1700	`N50-D1700`	69.50	2081.23	†	2042.13	1.91%	39.01%
	2500	`N50-D2500`	216.63	1337.35	†	1337.35	–	45.08%
60	1200	`N60-D1200`	93.82	3527.59	†	3527.59	–	54.68%
	1700	`N60-D1700`	168.53	2498.25	†	2498.25	–	66.43%
	2500	`N60-D2500`	313.21	1744.98	†	1744.98	–	53.49%
70	1200	`N70-D1200`	130.91	4304.18	†	4304.18	–	77.46%
	1700	`N70-D1700`	238.74	3185.66	†	3185.66	–	74.37%
	2500	`N70-D2500`	405.12	2240.70	†	2240.70	–	62.46%
80	1200	`N80-D1200`	226.21	5779.98	†	5779.98	–	83.67%
	1700	`N80-D1700`	369.13	4021.15	†	4021.15	–	77.23%
	2500	`N80-D2500`	776.20	2804.75	†	2804.75	–	83.75%

† *Time limit of 12 hours exceeded*

that the approach is able to find optimal solutions for instances with 20 and 30 sensors in few seconds. For larger instances (up to 80 sensors), the quality of the solutions could not be confirmed, as the optimal is not known, but a feasible solution is found in at most 13 minutes, while a MILP formulation lasts for 12 hours without improving the given solution.

Future works include solve the multiobjective version of the problem: minimizing energy consumption and delivery delay (MA route length); and the use of multiple MAs.

Acknowledgment

This project was partially supported by CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico and FAPEMIG - Fundação de Amparo à Pesquisa do estado de Minas Gerais. The first author was supported by CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

References

[1] W. Aioffi, G. R. Mateus, and F. Quintão. Optimization issues and algorithms for wireless sensor networks with mobile sink. In *International Network Optimization Conference*, 2007.

[2] W. M. Aioffi, C. A. Valle, G. R. Mateus, and A. S. da Cunha. Balancing message delivery latency and network lifetime through an integrated model for clustering and routing in wireless sensor networks. *Computer Networks*, 55(13):2803–2820, 2011.

[3] C. Bechelane, A. S. Cunha, and G. R. Mateus. The minimum cost hop-and-root constrained forest in wireless sensor networks. *Electronic Notes in Discrete Mathematics*, 35(0):139–144, 2009.

[4] B. Behdani, Y. Yun, J. C. Smith, and Y. Xia. Decomposition algorithms for maximizing the lifetime of wireless sensor networks with mobile sinks. *Computers & Operations Research*, 39(5):1054–1061, 2012.

[5] W. Cai, M. Chen, and T. Hara. A Genetic Algorithm Approach to Multi-Agent Itinerary Planning in Wireless Sensor Networks. *Mobile Networks and Applications*, 16(6):782–793, 2010.

[6] T. A. Feo and M. G. C. Resende. A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*, 8(2):67–71, 1989.

[7] S. Gao, H. Zhang, and S. K. Das. Efficient data collection in wireless sensor networks with path-constrained mobile sinks. *World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a*, pages 1–9, 2009.

[8] L. Gouveia. Multicommodity flow models for spanning trees with hop constraints. *European Journal of Operational Research*, 95(1):178–190, 1996.

[9] W. B. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Hawaii International Conference on System Sciences - Volume 8*, HICSS '00, pages 8020–8029, Washington, DC, USA, 2000. IEEE.

[10] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, 1(4):660–670, oct 2002.

[11] M. E. Keskin, I. K. Altinel, N. Aras, and Cem Ersoy. Lifetime Maximization in Wireless Sensor Networks Using a Mobile Sink with Nonzero Traveling Time. *The Computer Journal*, 54(12):1987–1999, 2011.

[12] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon. Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks. In *Proc. of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, pages 193–204, New York, NY, USA, 2003. ACM.

[13] O. C. Romão, A. G. Santos, and G. R. Mateus. Minimizing energy consumption on a hop-constrained wsn using a delay-constrained mobile agent. In *Proc. of the 10th Metaheuristics International Conference*, MIC'13. accepted for presentation, 2013.

[14] I. G. Siqueira, C. M. S. Figueiredo, A. A. F. Loureiro, J. M. Nogueira, and L. B. Ruiz. An integrated approach for density control and routing in wireless sensor networks. In *20th Intern. Parallel and Distributed Processing Symposium*, pages 10–19, 2006.

[15] L. Song and D. Hatzinakos. Architecture of wireless sensor networks with mobile sinks: Sparsely deployed sensors. *Vehicular Technology, IEEE Transactions on*, 56(4):1826–1836, July 2007.

[16] W. Wang, V. Srinivasan, and K. C. Chua. Using mobile relays to prolong the lifetime of wireless sensor networks. In *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 270–283, New York, NY, USA, 2005. ACM.

[17] O. Younis and S. Fahmy. Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3(4):366–379, oct.-dec. 2004.

[18] Y. Yun and Y. Xia. Maximizing the Lifetime of Wireless Sensor Networks with Mobile Sink in Delay-Tolerant Applications. *IEEE Transactions on Mobile Computing*, 9:1308–1318, 2010.

[19] J. Zheng and A. Jamalipour. *Wireless Sensor Networks: A Networking Perspective*. Wiley-IEEE Press, Hudson, New Jersey, USA, 2009.