

# A Unified Framework for Joint Sink Mobility and Routing to Increase the Lifetime of Wireless Sensor Networks

Jun Luo\*

Jean-Pierre Hubaux<sup>†</sup>

Technical Report LCA-REPORT-2008-015, June 2008

**Abstract**—The longevity of wireless sensor networks is a major issue that impacts the application of such networks. While communication protocols are striving to save energy by acting on sensor nodes, recent results show that network lifetime can be prolonged by further involving sink mobility. As most proposals give their evidence of lifetime improvement through either (small-scale) field tests or numerical simulations on rather arbitrary cases, a theoretical understanding of the reason for this improvement and the tractability of the joint optimization problem is still missing. In this paper, we build a unified framework for investigating the joint sink mobility and routing problem. We formally prove the NP-hardness of the problem. We also investigate the induced sub-problems. In particular, we develop an efficient interior point algorithm to solve the sub-problem involving a single sink, then we generalize this algorithm to approximate the original problem. Finally, we apply the algorithm to a set of typical topological graphs; the results demonstrate the benefit of involving sink mobility, and they also suggest the desirable moving traces of a sink.

**Index Terms**—Wireless sensor networks, lifetime, sink mobility, routing.

## I. INTRODUCTION

*Wireless Sensor Networks* (WSNs) are fast emerging as a new networking and sensing paradigm based on a large number of tiny sensor nodes. These networks can be deployed close to or inside the phenomenon under surveillance, and thus have the potential of providing diverse information to numerous applications. However, the small size of the sensor nodes (hence their capacity-limited power sources) is posing a great challenge: the longevity of WSNs under energy constraints should be addressed before we can benefit from their advantages. Communication protocols that strive to save energy in WSNs (e.g., [1]–[12]) mainly focus on the *sensor nodes*<sup>1</sup>, whereas a recent trend indicates a focus shift to the behavior of *sinks*<sup>2</sup> [13]–[20], which can be exploited to further

improve the lifetime of WSNs.

There are two approaches, *fast mobility* and *slow mobility*, for exploiting *sink mobility* to improve network lifetime. They are distinguished by the relationship between the moving *speed* of a sink and the tolerable *delay* of the data delivery. On one hand, a sink can “transport” data with its movements if its speed is **high** enough to produce a tolerable data delivery delay [13]–[15], and hence spare nodes from the traffic forwarding load. This is the fast mobility approach, as the sink should move sufficiently fast. On the other hand, moving the sink, even very **infrequently** (say once a week), may still benefit the network lifetime, because it can lead to a global load balancing in the entire network [16]–[20]. This is the slow mobility approach, because the mobility is several orders lower than the tolerable delay (but it barely affects the delay due to its infrequency). The main reason accounts for the improvement brought by the slow mobility approach is the typical many-to-one traffic pattern in WSNs. Such a pattern imposes on the nodes close to sinks a heavy forwarding load. While no energy conserving protocol alleviates such a load, moving sinks can distribute over time the role of bottleneck nodes and thus even out the load.

The general reason that sink mobility, no matter fast or slow, can improve network lifetime lies in the fact that mobility increases the **dimension** (thus the degree of freedom) of the problem. This follows the principle that optimizing an objective in a high-dimension space always leads to a result no worse than what can be achieved in a subspace of reduced dimension. However, solving problems in high-dimension space incurs a higher complexity. Existing approaches either directly consider the practical implementation issues before developing a theoretical understanding [15] or solve simplified sub-problems using contemporary software without paying attention to the tractability of the problem in general [17], [18], [20]. This prevents us from getting deeper insight on how and why sink mobility brings lifetime improvement.

In this paper, we investigate the problem of maximum lifetime data collection in WSNs by jointly considering sink mobility and routing. We consider a type of continuously monitoring WSNs whose data generation rates of sensors can be estimated accurately. We focus on the slow mobility approach. We build a unified framework to cover most of the joint sink mobility and routing strategies. Our investigation of the *maximizing network lifetime* (MNL) problem is based on a graph model. We show that the MNL problem involving multiple mobile sinks is NP-hard, but that certain induced

\*Jun Luo is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. Email: j7luo@engmail.uwaterloo.ca.

<sup>†</sup>Jean-Pierre Hubaux is the School of Computer and Communication Sciences, EPFL (Swiss Federal Institute of Technology in Lausanne), CH-1015, Lausanne, Switzerland. Email: jean-pierre.hubaux@epfl.ch.

The work presented in this paper was supported (in part) by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322. (<http://www.terminodes.org>)

<sup>1</sup>In this paper, the words *sensor*, *sensor node* and *node* are used interchangeably.

<sup>2</sup>These are the devices that collect data from WSNs; sometimes they are also termed *base stations*.

sub-problems having a practical significance are tractable. Moreover, we show that the MNL problem involving only a single mobile sink can be solved by an efficient interior point algorithm; we further generalize this algorithm to approximate the general MNL problem. Finally, we illustrate the benefit of using a mobile sink by applying our algorithm to a set of typical topological graphs.

Our main contributions are:

- We provide a constructive proof of the NP-hardness of the MNL problem involving multiple mobile sinks.
- We identify the sub-problem that has a potential to guide routing protocol designs in practice.
- We develop an efficient interior point algorithm for the sub-problem involving a single sink; it is then generalized to approximate the general MNL problem.
- We formally prove the superiority of moving the sinks over keeping them static.

The rest of this paper is organized as follows: Section II surveys related work. Section III and IV state our assumptions and formulate the MNL problem. Section V proves the NP-hardness of MNL. Section VI discusses induced sub-problems of MNL and in particular, Section VII investigates in detail the sub-problem that involves only a single mobile sink. Section VIII extends our investigations to the general MNL problem. Section IX reports the numerical experiment results on the algorithm we developed in Section VII. Finally, Section X concludes the paper.

## II. RELATED WORK

We first present the recent work consisting in improving network lifetime with mobile sinks. We then briefly discuss a few topics related to energy conservation in WSNs. Proposals related to sink mobility are also described in [21]–[25], but we will not discuss them because these proposals are either about coping with sink mobility (rather than exploiting it) [21]–[23] or about preventing buffer overflow (rather than extending lifetime) [24], [25]. We are aware that there have been significant efforts in designing online mobility control algorithms (e.g., [26], [27]), but our offline approach does serve as a benchmark. More importantly, our offline solution is applicable and more efficient provided that the data rates can be accurately estimated. Last but not least, we refer to [28] for a theoretical investigation on load-balancing (including using mobile sinks) in WSNs detecting bursty events.

### A. Moving Sinks to Improve Network Lifetime

If a sink moves fast enough to deliver data with a tolerable delay, WSNs may take advantage of *mobility capacity* [29]. In this *mobile relay* approach [13], [14], the mobile sink “picks up” data from nodes (through one-hop transmissions) and transports the data with mechanical movements. This approach trades data delivery latency for the reduction of energy consumption of nodes. While both [13] and [14] leverage only on uncontrollable mobility, [15] investigates the controllable mobility. This proposal is a compromise between the mobile relay approach and the *mobile sink* approach [16]–[20]: the sink relays data with its movements, and nodes

transmit data (through a multi-hop routing if necessary) when the sink moves to the closest point to them. A field study is reported in [15], and a simple theoretical analysis on this hybrid approach is presented later in [30]. In our paper, we will briefly investigate the tractability of the lifetime optimization problem using a controllable mobile relay (see Section V); we will not cover the approaches involving uncontrollable sink mobility [13], [14].

If a sink moves infrequently, its average speed is not high enough to produce tolerable data delivery delay. In fact, the sink mobility may take a **discrete** form: the movement trace consists of several *anchor points* between which sinks move and at which they pause. Consequently, data packets have to be carried from their origin to the sinks through multihop routing. However, it has recently been observed that sink mobility still offers benefits in terms of network lifetime [16], [17], thanks to a consequent load-balancing effect. Unfortunately, the formulations in [16], [17] cannot lead to a global lifetime maximization; only local optimums are obtained for each pause time. In another contribution [18], Wang et al. make a different formulation where the flows are pre-scheduled and the pause times become the variables of the lifetime optimization problem. However, a global lifetime maximization is still not achieved in this proposal because only a subset of possible flows is considered. Luo and Hubaux [19] take a continuum model and obtain some forms of optimality by exploiting the symmetry of the assumed circular networks; it is difficult to apply these results to more general network topologies. Papadimitriou and Georgiadis [20] extend the formulation of [18] by jointly considering sink mobility and routing. The full scale problem is, however, not addressed because of its prohibitive complexity; the sink is hence confined to a limited number of positions in the numerical experiments. An extension of [18] is presented recently in [31], where a more comprehensive formulation is taken and a greedy online algorithm is reported. Note that, apart from [17], the authors of [16], [18]–[20], [31] consider only a single mobile sink. Most importantly, the hardness of the joint sink mobility and routing for lifetime optimization is not evaluated in [16]–[18], [20], [31]. The general framework we propose in this paper encompasses all the formulations in [17], [18], [20], [31].

Recently, Wang, Srinivasan and Chua [32] presented a *mobile node*<sup>3</sup> approach. The idea is that a few powerful mobile nodes are deployed to replace heavily loaded (static) nodes, such that these static nodes can shut down for energy saving. The same authors have further investigated the tradeoff between using mobile node and deploying dense network [33]. Whereas our framework will not cover this problem due to the fundamental difference between moving nodes and sinks, our analysis still sheds light on understanding this approach from a different perspective (see Section VIII-B for details). We also note that, as proved in [32], the mobile node approach can achieve the same order of lifetime as the mobile sink approach only if a sufficient number of mobile nodes ( $O(\sqrt{n})$  for an  $n$ -nodes network) are deployed.

<sup>3</sup>The approach is termed “mobile relay” in [32], but we give it another name in order to be consistent with our terminology (where “mobile relay” is given to another approach [13], [14]).

## B. Energy Conservation Protocols

The mobile sink approach is closely related to existing energy conserving routing (e.g., [1]–[3]). These protocols aim at balancing the energy consumption instead of minimizing the absolute consumed power. The mobile sink approach, by further involving sink mobility, increases the dimension of such optimization problems. Topology control (including transmission power control, e.g., [4]–[6] and node scheduling, e.g., [7]–[9]), clustering (e.g., [10]–[12]), sensor-sink coordination (e.g., [34]), and exploiting limited infrastructures (e.g., [35]) are, among others, important ways of reducing energy consumption. Although the mobile sink approach is orthogonal to these proposals, it serves as a potential complement.

## III. SYSTEM MODEL

We model a WSN as a digraph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  where  $\mathcal{V}$  represents  $|\mathcal{V}| = n$  sensor nodes. There is a cost assignment  $c: \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}^+$ , such that  $\exists(i, j) \in \mathcal{E}$  if and only if the transmission energy (defined later) of  $i \in \mathcal{V}$  is no less than  $c(i, j)$ . Apart from the sensor nodes, there is a set  $\mathcal{S}$  ( $|\mathcal{S}| = m < n$ ) of sinks that harvest data from the WSN. The following properties are specified for the network:

- A) Sensor nodes are stationary, but the sinks change their positions from time to time with a negligible traveling time between two positions.
- B) We only consider the case where the location of each sink coincides with the location of one of the nodes, such that the network topology (thus the graph  $\mathcal{G}$ ) does not change with different locations of the sinks.
- C) Each sink inherits the data collection function of the co-located node. It behaves like a common node for receiving data, but it has long-range (wireless) communication facilities to transmit data out of the considered WSN.
- D) The data traffic flows from each node  $i \in \mathcal{V}$  (i.e., all nodes are *sources*) to one of the *sinks*  $s \in \mathcal{S}$  (through multi-hop relaying if no direct connection exists between  $i$  and  $s$ ), and the control traffic involved (e.g., in a routing protocol) is not considered because it has the same effect to all nodes.
- E) Data transmission and reception are the dominating factors for the energy consumption of a node.
- F) The transceiver of a node is half-duplex, so the (wireless) channel capacity of a node has to be divided for both transmission and reception.

In addition, there are attributes associated with a node  $i \in \mathcal{V}$ :

- H) a value  $E_i$  (Joules) representing the initial energy reserve of the node,
- I) two values  $e_i^{tx}$  and  $e^{rx}$  representing the energies for the node to transmit and receive a unit of data (e.g., Joules/byte),
- J) a quantity  $R_i$  that upper bounds the node's transmission rate (e.g., bytes/second),
- K) a rate  $\lambda_i$  of the information generation.

As shown in [36], most sensor radios have a constant  $e^{rx}$  regardless of the transmission power of a sender. Fig. 1 shows the graph representation of a WSN.

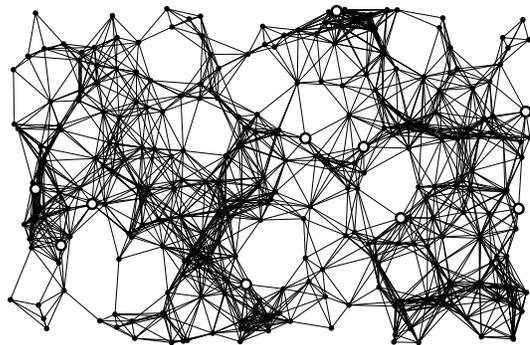


Fig. 1. Network graph model. The WSN has 300 nodes (black points) and 10 sinks (white points). Nodes are uniformly distributed in a rectangular area. Here we take a cost assignment based on the Euclidean distance and a uniform transmission energy assignment.

*Remarks:* First, the assumption A) of a negligible traveling time for a sink comes from the fact that the time during which the sink pauses at a certain location can be long enough to amortize the routing overhead introduced by the sink mobility. We have recently validated this assumption by showing that a significant lifetime improvement can still be achieved when taking the routing overhead into account [37].

Secondly, one might expect that relaxing the location limitation of sinks posed by assumption B) could further improve the lifetime [20]. However, the degree of freedom introduced by sink mobility is confined by the network size; it will not increase unboundedly by adding more potential sink locations. In addition, allowing sink locations to be “off-the-graph” would produce a hard non-linear optimization problem, for which even the bounds are impossible to obtain. In other words, though the solution of such a problem would provide us with the globally optimal sink locations, the non-existence of approximation algorithms renders the problem almost useless.<sup>4</sup> Most importantly, as we will show in Section VIII-B and IX-A, sink mobility does not always benefit lifetime without co-locating sinks with nodes.

Thirdly, we note that one significant difference between our model and those in [1], [3], [20] is that, in our model, the energy consumption of data transmission  $e_i^{tx}$  is associated with a vertex (a node) instead of an edge (a link), as defined by assumption I). We consider this model to be more realistic, because, although nodes have the flexibility to tune their own transmission power, it is not cost-effective to dynamically tune the power for destinations at different distances. In addition, tuning transmission power according to transmission distances is not always feasible either since a node might not know the distances. Therefore, a reasonable scenario, in our opinion, is that each node sets up a transmission power according to certain topology control mechanisms [4]–[6] at the network initialization phase and this power is fixed until some topology changes happen.

<sup>4</sup>One possible simplification is to limit the possible sink locations to a finite set [31]. Our model can cover this case by extending  $\mathcal{G}$  to include these locations in  $\mathcal{V}$ . However, it is not clear if such a formulation would provide better solution.

Finally, assuming a known rate  $\lambda_i$  in assumption  $K$  does not mean our model cannot cover the sensing of physical phenomena that are bursty. As we are only concerned with the long-term average load carried by nodes, our model allows the data generation by node to follow any point process as far as the process has a intensity (or a rate in our case) and this intensity can be estimated.

#### IV. PROBLEM FORMULATION

We begin with the definition of network lifetime for WSNs, and then formulate the problem of *maximizing network lifetime* (MNL) under an optimization framework.

##### A. Network Lifetime

The *network lifetime* can be defined in various ways; these definitions focus on either individual [1] or collective [38] behaviors of nodes. Because the individuality has an implication on the collectiveness (e.g., the death of a node is soon followed by the death of all nodes one-hop away [17]), we define the network lifetime as the time period for the first node to run out of its energy reserve [1].

##### B. The Optimization Problem

We denote the lifetime by  $T$  and use  $t_k$  to indicate the time span for the  $k$ th *epoch*: a new epoch begins when some sinks change their locations. We also define  $q_{ij}^k$  as the total information flow from node  $i$  to node  $j$  during  $t_k$ . By assumption  $I$ ), the total energy consumed by node  $i$  during  $t_k$  is given by  $\sum_j q_{ij}^k e_i^{tx} + e^{rx} \sum_l q_{li}^k$ , where the sum is over all adjacent vertices of  $i$  (the adjacency is implied by the cost assignment  $c$  and the transmission energies of node  $i$  and its neighbors). In order to indicate the location of a sink during  $t_k$ , we use a binary variable  $\delta_{is}^k$  to represent the relation between the location of a sink  $s \in \mathcal{S}$  and that of node  $i \in \mathcal{V}$ , such that  $\delta_{is}^k = 1$  if sink  $s$  is co-located with node  $i$  and  $\delta_{is}^k = 0$  otherwise. We also associate with node  $i$  an outgoing flow  $F_i^k$  for epoch  $t_k$ ; it becomes positive only if  $\exists s \in \mathcal{S} : \delta_{is}^k = 1$ . Hence the mixed-integer nonlinear programming for MNL is as follows:

$$\text{Maximize } T = \sum_k t_k \quad (1)$$

$$\sum_j q_{ij}^k - \sum_j q_{ji}^k - \lambda_i t_k + F_i^k = 0 \quad \forall i, k \quad (2)$$

$$F_i^k - \sum_j \lambda_j \cdot t_k \sum_s \delta_{is}^k \leq 0 \quad \forall i, k \quad (3)$$

$$\sum_s \delta_{is}^k \leq 1, \quad \sum_s \sum_i \delta_{is}^k - m = 0 \quad \forall k \quad (4)$$

$$\sum_j q_{ij}^k + \sum_j q_{ji}^k - R_i t_k \leq 0 \quad \forall i, k \quad (5)$$

$$\sum_k \left( \sum_j q_{ij}^k e_i^{tx} + e^{rx} \sum_j q_{ji}^k \right) - E_i \leq 0 \quad \forall i \quad (6)$$

$$q_{ij}^k, t_k, F_i^k \geq 0 \quad \forall i, j, k \quad (7)$$

$$\delta_{is}^k \in \{0, 1\} \quad \forall i, k, s \quad (8)$$

where  $\sum_i$  means a sum over all possible  $i$  (this **notation** is used throughout the paper). We explain each constraint:

- **FLOW CONSERVATION (2)–(4)**: The outgoing flow exceeds the incoming flow by an amount of  $\lambda_i t_k$ , by assumption  $K$ ), if  $\sum_s \delta_{is}^k = 0$  (i.e., no sink is co-located with  $i$ ; by (3),  $F_i^k = 0$  if  $\sum_s \delta_{is}^k = 0$ ); otherwise the flow  $F_i^k$  should be taken into account (again by (3), this quantity is bounded from above by the total incoming flow  $\sum_j \lambda_j \cdot t_k$ ). Constraint (4) states that no sink is co-located with another sink and the number of sinks is exactly  $m$ .
- **RATE CONSTRAINT (5)**: The average rate of data transmission and reception by node  $i$  should not exceed the channel capacity, due to our assumption  $F$ ).
- **ENERGY CONSTRAINT (6)**: The energy spent by node  $i$  to transmit and receive data during the whole network lifetime is upper bounded by the initial energy reserve  $E_i$ , due to our assumptions  $H$ ) and  $I$ ).

In this paper, we ask the question of what is the maximum lifetime and we consider only WSNs that do not demand high throughput. For such networks, constraint (5) is inactive if the rate set  $\{\lambda_i\}$  is scaled properly. Also, the interferences among links become negligible if the rate set  $\{\lambda_i\}$  is scaled to a very low level. Therefore, we do not take interference into account. However, we could also ask what is the maximum amount of data that can be collected, considering  $\lambda_i, i = 1, 2, \dots, n$  as variables. In such a case, the objective function becomes  $T \sum_i \lambda_i$  and (5) limits the potential choices of  $\lambda_i$ . This requires us to find a good tradeoff between lifetime and throughput. For a detailed treatment of the latter problem, we refer to our recent work [39].

#### V. HARDNESS OF THE PROBLEM

The potential number of sink layouts for the MNL problem is  $\binom{n}{m}$ , which, by Stirling's approximation, is exponential in  $n$  for an arbitrary  $m$ . Given the exponential (in  $n$ ) number of columns in the programming, it is not difficult to believe that the MNL problem is "very hard". In order to formally evaluate its hardness, we consider the following decision problem that is derived as a restricted case for the original MNL, which we term MSP (standing for Mobile Sink Positioning):

**INSTANCE**: A graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , a cost assignment  $c$ , a set  $\mathcal{S}$  of sinks with  $|\mathcal{S}| < |\mathcal{V}|$ , and for each  $i \in \mathcal{V}$ , a transmission energy  $e^{tx}$ , an energy reserve  $E$ , a rate  $\lambda$ , and a positive real  $t$ .

**QUESTION**: Is there a *sink layout schedule*  $\{(sl_k, t_k)\}$  ( $sl_k$  is a vector of  $[\delta_{is}^k]$  where  $\delta_{is}^k : \mathcal{V} \rightarrow \{0, 1\}$  and  $\sum_i \sum_s \delta_{is}^k = |\mathcal{S}|$ ) such that the lifetime  $T = \sum_k t_k$  is at least  $t$ ?

*Proposition 1*: The MSP problem is NP-hard.

*Proof*: The NP-hardness of the MSP problem can be shown by giving a polynomial-time reduction from DOMINATING SET [40] to a special case of MSP where a schedule consists of only one element  $(sl, T)$ . Given an undirected graph  $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$  (an instance of DOMINATING SET), we construct a complete graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , let  $e^{tx} = 1$ ,  $E = 1$ ,

and  $\lambda = 1$ , and define the cost assignments as follows:

$$c(e) = \begin{cases} 1 & e \in \mathcal{E}' \\ 2 & e \notin \mathcal{E}' \end{cases}$$

The reduction is straightforward and can be done in polynomial time. Now, if  $\mathcal{G}$  admits a lifetime  $T = 1$ , the sink layout  $sl$  suggests the dominating set in  $\mathcal{G}'$ . Conversely, if  $\mathcal{G}'$  has a dominating set of size  $|\mathcal{S}|$ , taking the dominating set as  $sl$  allows  $\mathcal{G}$  to admit  $T = 1$ .<sup>5</sup> In other words, answering the decision problem of whether  $\mathcal{G}'$  has a dominating set of size  $|\mathcal{S}|$  is equivalent to the decision problem of whether  $\mathcal{G}$  admits a lifetime  $T = 1$ . ■

*Remarks:* The above proof leaves several problems open. First, the proof is not constructive because it would not suggest an algorithm to solve the problem provided that there were an algorithm for the DOMINATING SET problem (a common weakness of applying *restriction* [40] for proving NP-hardness). Secondly, it fails to expose the structure of the problem: for example, whether the complexity lies in the selection of  $sl_k$  or of  $t_k$ . Finally, it does not show if the problem is tractable with a single sink; the DOMINATING SET problem is tractable if the set contains only one element. Therefore, we provide a constructive proof in Appendix I, and we answer other questions in the upcoming sections.

At this point, it is worth noting that the lifetime optimization problem of the joint (fast) mobility and routing strategy proposed in [15] is also intractable, because the corresponding decision problem that we term SMRP (Single Mobile Relay Positioning) is also NP-hard:

INSTANCE: A graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , a cost assignment  $c$ , a set  $\mathcal{S}$  of **sink positions** with  $|\mathcal{S}| < |\mathcal{V}|$ , and for each  $i \in \mathcal{V}$ , a transmission energy  $e^{tx}$ , a energy reserve  $E$ , a rate  $\lambda$ , a constraint that  $i$  sends data to only one  $s \in \mathcal{S}$ , and a positive real  $t$ .

QUESTION: Is there a sink layout schedule  $\{(\delta_{is}, T)\}$ , where  $\delta_{is} : \mathcal{V} \rightarrow \{0, 1\}$ ,  $\sum_i \delta_{is} = 1$  (only one sink at a given moment) and  $\sum_i \sum_s \delta_{is} = |\mathcal{S}|$ , such that the lifetime  $T$  is at least  $t$ ?

A simple proof can be achieved by again applying the reduction from the DOMINATING SET.

## VI. INDUCED SUB-PROBLEMS

As we have shown in Section V, the MSP problem is hard, although it is a simplified version of the original MNL by considering homogenous (in terms of, e.g., energy reserve and consumption) nodes. Therefore, we turn to investigate several relaxed versions of the MNL problem, by decomposing the two variable  $sl_k$  and  $t_k$  in the schedule. We note that a common approach of relaxing the integer constraints, i.e., (8), does not work for the MNL problem, because this relaxation renders every node a sink and thus leads to a trivial solution.

<sup>5</sup>A similar reduction has been shown by Bogdanov et al. [41] for proving the hardness of the BSP (Base Station Positioning) problem. While their problem intends to find the maximum rate admitted by a WSN, we try to maximize the network lifetime for a given rate.

### A. Uniform Epoch

If we assume a uniform epoch span  $\tau$ , the question asked by the resulting MSP-UE (Mobile Sink Positioning with Uniform Epoch) problem becomes:

QUESTION: Is there a sink layout schedule  $\{(sl_k, \tau)\}$  of length  $l$  such that the lifetime  $T = l\tau$  is at least  $t$ ?

*Proposition 2:* The MSP-UE problem is NP-hard.

*Proof:* Similar to the proof for *Proposition 1*, the NP-hardness of MSP-UE can be shown by the reduction from the DOMINATING SET to the MSP-UE with single element schedule, i.e.,  $l = 1$ . ■

*Remarks:* The above result shows that relaxing the constraint on the time schedule does not significantly simplify the MSP (and thus the MNL) problem. It hints that the complexity lies in the selection of sink layouts instead of the time schedule. In fact, a further relaxation was described in [17]: it consists in solving a sequence of MSP-UEs with  $l = 1$ . Such a problem is still NP-hard because the hardness of MSP-UE problem does not depend on  $l$ , as shown by the proof of *Proposition 2*.

### B. Pre-defined Flow Schedule

Given all possible sink layouts  $sl_k: k \leq \binom{n}{m}$ , it is always possible to come up with a flow schedule (feasible but not necessarily optimal<sup>6</sup>)  $fs_k$  (where a  $fs_k$  is a vector of  $[\tilde{q}_{ij}^k] : \mathcal{E} \rightarrow \mathbb{R}_0^+$ ) for each  $sl_k$ . Note that here a flow  $\tilde{q}_{ij}^k$  on an edge  $(i, j)$  is defined as the information **rate** (as opposed to the total information  $q_{ij}^k$  defined in Section IV). Now the MNL problem becomes a search for a time schedule  $\{t_k\}$  such that  $T = \sum_k t_k$  is maximized; we formulate the problem as the following linear program:

$$\text{Maximize } \sum_k t_k \quad (9)$$

$$\sum_{fs_k: i \in fs_k} p_i^k t_k \leq E_i \quad \forall i \quad (10)$$

$$p_i^k, t_k \geq 0 \quad \forall i, j, k \quad (11)$$

where  $p_i^k = \sum_j \tilde{q}_{ij}^k e_i^{tx} + e^{rx} \sum_l \tilde{q}_{li}^k$  is the **power** consumption of node  $i$  during the  $k$ th epoch. This formulation has also been taken by [18], [19].

Although the above linear program could involve an (potentially) exponential number of variables  $t_k$ , the vector  $[t_k]$  for the optimal solution contains no more than  $n$  non-zero elements. Since the matrix  $\mathbf{P} = [p_i^k]$  has a row rank of at most  $n$ , it cannot have more than  $n$  linearly independent columns. Consequently, each basic solution  $[\hat{t}_k]$  contains at most  $n$  non-zero elements. The problem involving only a single sink leads to  $n$  potential layouts, for which the solution becomes straightforward. If multiple sinks are involved, the problem might have a very high complexity due to the possibly exponential number of variables. We can apply an interior

<sup>6</sup>It becomes a GREEDY approximation if optimal flows are required. However, GREEDY does not necessarily perform better than an arbitrary (feasible) flow, because what matters is the complementarity among flows (which leads to the overall optimality) rather than the individual optimality.

point algorithm in this case. It is easy to see that the above linear program is a packing LP (a linear program in the form  $\max\{c^T x | Ax \leq b, x \geq 0\}$ ); its  $(1 + \epsilon)$ -approximation algorithm is proposed by Garg and Könemann [42].

We note that the formulation of (9)–(11) has also a significance in practice. In reality, the flow in a network is shaped by a certain routing protocol instead of being defined by the solutions of some optimization problems. Therefore, once we introduce a set of sink layouts schedule  $sl_k$  into a WSN, the network itself will figure out a flow schedule  $fs_k$ . This “pre-defined” schedule can then be taken as the input to the above program to obtain an optimal time schedule. In a companion work [37], we show how this relaxation can be applied when designing a practical routing protocol for supporting mobile sinks.

## VII. MAXIMIZING NETWORK LIFETIME FOR A SINGLE MOBILE SINK (MNL-SMS)

Although the MNL-SMS problem is again a sub-problem of MNL, we discuss it in a separate section due to its relevance. The problem is polynomially solvable in its original form, because it can be formulated as a linear program with  $O(n^2)$  constraints. Nevertheless, we propose an interior point algorithm that solves the problem efficiently. More importantly, this algorithm will be further generalized to solve the original MNL problem in Section VIII-A.

### A. The MNL-SMS Problem

Based on (1)–(8), the Arc-Flow form of MNL-SMS problem can be formulated as the following linear program:

$$\text{Maximize } \sum_k t_k \quad (12)$$

$$\sum_j q_{ij}^k - \sum_j q_{ji}^k - \lambda_i t_k = 0 \quad \forall k, i \neq k \quad (13)$$

$$\sum_k \left( \sum_j q_{ij}^k e_i^{tx} + e^{rx} \sum_j q_{ji}^k \right) - E_i \leq 0 \quad \forall i \quad (14)$$

$$q_{ij}^k, t_k \geq 0 \quad \forall i, j, k \quad (15)$$

We simplify the formulation by assuming that the sink is co-located with node  $k$  during the  $k$ th epoch.<sup>7</sup> We also deliberately drop the rate constraint (5) because we can scale the set  $\{\lambda_i\}$  anyway to meet this constraint. It can be easily seen that the number of constraints is bounded by  $|\mathcal{V}|^2 + |\mathcal{V}|$  (with (13) introducing the first term and (14) accounting for the second) and is thus polynomial in  $n$ . Although this linear program is polynomially solvable (by, for example, the *ellipsoid* algorithm [43]), directly solving it is practically ineffective on all but very small scale problems (similar to the case of concurrent flow problem [44]). In addition, common techniques such as the simplex algorithm cannot be extended to address the original MNL problem. In Section VII-B, we will discuss an interior point algorithm that solves the problem efficiently. Moreover, we will extend the algorithm to approximate the solution of the original MNL in Section VIII-A.

<sup>7</sup>Note that this procedure does not change the problem. It only requires the numbering of the nodes to coincide with that of the epochs.

### B. The Primal-Dual Interior Point Algorithm

Let us re-formulate the MNL-SMS problem into a Path-Flow form:

$$\text{Maximize } \sum_k t_k \quad (16)$$

$$\sum_{p \in P_{ik}} f(p) - \lambda_i t_k = 0 \quad \forall i, k \quad (17)$$

$$\sum_k \sum_{p \in P_i^k} f(p)(e_i^{tx} + e^{rx}) - E_i \leq 0 \quad \forall i \quad (18)$$

$$f(p), t_k \geq 0 \quad \forall p, k \quad (19)$$

where  $p$  refers to a certain path and  $f(p)$  is the flow that goes through  $p$ . Furthermore,  $P_{ik}$  stands for the set of paths between node  $i$  and the sink location during the  $k$ th epoch<sup>8</sup> and  $P_i^k$  is the set of paths that go through node  $i$  in the  $k$ th epoch.

The dual problem is given by:

$$\text{Minimize } \sum_i E_i w(i) \quad (20)$$

$$\sum_i \lambda_i W(i, k) \geq 1 \quad \forall k \quad (21)$$

$$\sum_{j \in p, p \in P_{ik}} w(j)(e_j^{tx} + e^{rx}) - W(i, k) \geq 0 \quad \forall i, k \quad (22)$$

$$w(j) \geq 0 \quad \forall i, j, k \quad (23)$$

where  $W(i, k)$  is the weight assigned to a “commodity” (data flow injected at a node) from node  $i$  to the sink location during the  $k$ th epoch and  $w(j)$  is the weight assigned to a node  $j$ . The weight of a node  $w(j)$  represents the marginal “cost” of using an additional unit of energy of the node, and the weight of a commodity  $W(i, k)$  represents the marginal “cost” of rejecting a unit of demand of the commodity. Provided that the maximum lifetime is achieved,

- (21) says that the sum of  $\lambda_i$  multiplied by weights  $W(i, k)$  for all  $n$  commodities in any epoch  $k$  is at least 1. This means that the “cost” of increasing the lifetime by one time unit without admitting yet another  $\sum_i \lambda_i$  units of demand exceeds or balances the “revenue”, and
- (22) states that the shortest path between an arbitrary node pair  $i$  and  $k$  (the “cost” of routing a unit of demand) is no less than  $W(i, k)$  (the “cost” of rejecting a unit of demand from  $i$  to  $k$ ).

Otherwise a longer lifetime could have been “profitable” either by rejecting or by admitting (thus routing) more demands. Here the length of a path is computed as the sum (over all nodes along the path) of the product of node weight  $w(i)$  and the node energy consumption  $e_i^{tx} + e^{rx}$ .

Usually, the flow maximization problem involving multiple  $s$ - $t$  flows can be solved by one of the algorithms proposed by Garg and Könemann [42]. However, MNL-SMS is a combination of two problems, namely a *maximum concurrent flow* problem and a *maximum multicommodity flow* problem. MNL-SMS is, on one hand, a maximum concurrent flow problem because each node has a demand  $\lambda_i$  and the objective is to find a maximum multiple  $T$  for all nodes. On the other hand, if the

<sup>8</sup>It is indeed node  $k$  due to the specific numbering taken in Section VII-A.

time schedule  $\{t_k\}$  is considered as a set of “commodities”, the objective is to maximize  $\sum_k t_k$  without caring about any demand (note that some “commodities” can be zero), which indicates a maximum multicommodity problem. Therefore, we need to develop new algorithms to solve MNL-SMS.

Let us denote the objective of the dual problem by  $G(w) = \sum_i E_i w(i)$ . In order to minimize  $G(w)$ ,  $w(i)$  should be as small as possible, but it is bounded from below by  $W(i, k)$  through (21) and (22). Taking an arbitrary assignment  $w$  and  $W(i, k) = \sum_{j \in \min\{p | p \in P_{ik}\}} w(j)(e_j^{tx} + e^{rx})$  (i.e., the length of the shortest path from  $i$  to  $k$ ), we meet (22). Then (21) becomes the following constraints:

$$\sum_i \lambda_i \left[ \sum_{j \in p, p \in P_{ik}} w(j)(e_j^{tx} + e^{rx}) \right] \geq 1 \quad \forall k$$

This assignment is not necessarily feasible because it might violate the above constraints. However, it can be made feasible by finding the most violated constraint and scale the assignment accordingly. In other words, if there is an *oracle* that identifies  $\min_k \rho_k(w) : \rho_k(w) = \sum_i \lambda_i W(i, k) < 1$ , we can scale all assignments  $w(j), W(i, k), \forall i, j, k$  by  $[\min_k \rho_k(w)]^{-1}$  and make a feasible assignment. Therefore, the dual problem is equivalent to finding a weight assignment  $w : \mathcal{V} \rightarrow \mathbb{R}_0^+$  such that  $G(w)/\rho(w) : \rho(w) \equiv \min_k \rho_k(w)$  is minimized. We denote  $\min_w [G(w)/\rho(w)]$  by  $\beta$ . Note that this interpretation of the dual problem already suggests a duality theorem analogous to the *max-flow min-distance ratio* theorem [44] (which is in turn analogous to the *max-flow min-cut* theorem of Ford and Fulkerson [45] for single *s-t* flow). We will discuss this point more in detail in Section VIII-B.

The algorithm proceeds in iterations. Let  $w_{i-1}, W_{i-1}$  be the weight assignment at the beginning of the  $i$ th iteration and let  $\{t_{k,i-1}\}$  be the time schedule after iterations  $1, \dots, i-1$ . In the  $i$ th iteration, we route  $\sum_l \lambda_l$  units of commodity along the paths (and thus to the corresponding sink location) given by an oracle (we will specify it later) that computes  $\min_k \rho_k(w)$  and let  $t_{k,i} = t_{k,i-1} + 1$ . Let  $f_i(l)$  be the flow through node  $l$  and  $p_{l,k}, \forall l$  be the paths suggested by  $\min_k \rho_k(w)$  in this iteration. The new weight assignment to a node  $l$  is given by  $w_i(l) = w_{i-1}(l)(1 + \epsilon f_i(l)(e_l^{tx} + e^{rx})/E_l)$ , and the new weight assignments to a commodity are computed as  $W_i(l, k) = \sum_{j \in p_{l,k}} w_i(j)(e_j^{tx} + e^{rx})$ . Note that  $p_{l,k}$  is indeed the shortest path from  $l$  to  $k$  because it is suggested by the oracle that computes  $\min_k \rho_k(w)$ . Now the dual objective is updated as:

$$\begin{aligned} G(w_i) &= \sum_l E_l w_i(l) \\ &= G(w_{i-1}) + \epsilon \sum_l w_{i-1}(l) f_i(l) (e_l^{tx} + e^{rx}) \\ &= G(w_{i-1}) + \epsilon \cdot \rho(w_{i-1}) \end{aligned} \quad (24)$$

Initially, the weight assignment to a node  $l$  is  $w_0(l) = \delta/E_l$ . The iteration stops when  $G(w_i) \geq 1$  for the first time. We refer to Appendix II for details of setting parameters  $\epsilon$  and  $\delta$ .

The oracle that computes  $\min_k \rho_k(w)$  is simply an extension of the Floyd-Warshall algorithm [46] that computes all-pairs shortest path with a time complexity of  $\Theta(n^3)$ . We organize the results of the Floyd-Warshall algorithm into “clusters”;

each cluster includes paths that have a common end. Then we run a search algorithm in order to find the best “median”  $k$  that achieves  $\min_k \rho_k(w)$ . This oracle has a time complexity of  $\Theta(n^3)$  (because the later clustering and searching both have a time complexity at least one order lower than that of the Floyd-Warshall algorithm). Combining the oracle with the iteration procedure, we have the following proposition:

*Proposition 3:* Given  $\sum_i \lambda_i \leq E_l/(e_l^{tx} + e^{rx}), \forall l$ ,<sup>9</sup> there is an algorithm that computes a  $(1 - \epsilon)^{-2}$ -approximation to the MNL-SMS problem in time  $\Theta(n \log n) \cdot T_{\text{oracle}}$ , where  $T_{\text{oracle}} = \Theta(n^3)$  is the time complexity for the oracle to compute  $\min_k \rho_k(w)$ .

*Proof:* See Appendix II. ■

## VIII. MAXIMIZING NETWORK LIFETIME FOR MULTIPLE MOBILE SINKS

We are now ready to investigate the original MNL problem that may involve multiple mobile sinks. Although it is shown to be NP-hard by *Proposition 1*, we are able to approximate the solution based on the interior point algorithm described in Section VII-B. Moreover, we are able to solve the following crucial decision problem:

**TO MOVE OR NOT TO MOVE (TMNTM):** Is there a sink layout schedule  $\{(sl_k, t_k)\}$  such that the lifetime  $T = \sum_k t_k$  is longer than what is achieved by any fixed layout  $sl$ ?

This was never fully addressed in the previous work [17]–[20].

### A. The Approximation Algorithm

The Path-Flow form and its dual of the original MNL problem is the same as those of the MNL-SMS problem (16)–(19) and (20)–(23), apart from the fact that  $P_{ik}$  now stands for the set of paths between node  $i$  and one of the  $m$  sinks during the  $k$ th epoch and  $W(i, k)$  becomes the weight assigned to the “commodity” from node  $i$  to that sink during the  $k$ th epoch. Such a formulation hides the complexity of the problem behind a seemingly simple formulation, as the size of a set  $P_{ik}$  can be enormous. A formal evaluation of the complexity is given in Appendix I, where we also point out that, if we have an oracle that is able to solve the  $p$ -median problem below, then we are able to solve the original MNL problem.

**INSTANCE:** A graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , a weight assignment  $w : \mathcal{V} \rightarrow \mathbb{R}_0^+$ , a length assignment<sup>10</sup>  $l : \mathcal{V} \rightarrow \mathbb{R}_0^+$ , positive integer  $K \leq |\mathcal{V}|$ , and positive rational number  $B$ .

**QUESTION:** Is there a set  $\mathcal{P}$  of  $K$  “points on  $\mathcal{G}$ ” such that if  $d(v)$  is the length of the shortest path (i.e., the

<sup>9</sup>This assumption is reasonable because each sensor node should be equipped with an energy source that is at least enough for the node to forward data for all nodes in one time unit. Otherwise if a node  $l : E_l/(e_l^{tx} + e^{rx}) < \sum_i \lambda_i$  is deployed close to a static sink (assuming a randomly deployed WSN), the network lifetime can be even less than one time unit. In addition, it can be proved that an approximation ratio of  $(1 - \epsilon)^{-3}$  is still achievable without this assumption.

<sup>10</sup>Usually, a length assignment is associated with edges. However, we can always convert our node-capacity based problem to a link-capacity based version by replacing a node with two nodes and a link having the same capacity.

sum of all length assignments along the path) from  $v$  to the closest point in  $\mathcal{P}$ , then  $\sum_{v \in \mathcal{V}} \omega(v) \cdot d(v) \leq B$ ?

However, the p-median problem is NP-complete [40]. Yet, the following proposition provides us with an approximation algorithm for the original MNL problem.

*Proposition 4:* If the p-median oracle can be approximated within a ratio of  $\alpha > 1$  (i.e., the oracle has an  $\alpha$ -approximation), then the interior point algorithm given in Section VII-B along with this oracle provides an  $\alpha \cdot (1 - \epsilon)^{-2}$ -approximation to the original MNL problem.

*Proof:* See Appendix III. ■

*Remarks:* In fact, Arya et al. [47] gave a  $(3 + \omega)$ -approximation algorithm for the p-median problem. Therefore, we have an algorithm to approximate the original MNL problem with a factor of  $(3 + \omega)(1 - \epsilon)^{-2}$ .

### B. Duality Theory for MNL and the Answer to TMNTM

There is another benefit coming with the primal-dual interpretation provided in Section VII-B: it helps us to build the related dual theory and allows us to easily address the TMNTM decision problem. We recapitulate the observation that we make on the dual problem of MNL-SMS in the following theorem:

*Theorem 1:* [MAX-LIFETIME MIN-POTENTIAL RATIO THEOREM] Given the maximization lifetime problem formulated in (16)–(19), the optimal lifetime  $T$  is such that

$$T = \min_w \left[ \frac{G(w)}{\rho(w)} \right]$$

where  $G(w) = \sum_i E_i w(i)$  is a linear combination of the energy reserves of all nodes with coefficients  $w(i)$ , and  $\rho(w) \equiv \min_k \rho_k(w) = \min_k \left( \sum_i \lambda_i \sum_{j \in \min\{p|p \in P_{i,k}\}} w(j)(e_j^{tx} + e^{rx}) \right)$  is the minimum “potential” (computed as the sum of the minimum “cost”, given  $w(i)$ , to route  $\lambda_i$  from node  $i$  to one of the  $m$  centers) achieved among all possible center layouts (or sink layouts).

We omit the detailed proof of this theorem; see Section VII-B for a sketch of the proof. We also quote the theorem given in [44] and improved in [42] as follows:

*Theorem 2:* [MAX-FLOW MIN-DISTANCE RATIO THEOREM] Given the maximization lifetime problem formulated in (16)–(19) but with a fixed schedule consisting of only one element  $(sl, t)$ , the optimal lifetime  $T'_{sl}$  is such that

$$T'_{sl} = \min_w \left[ \frac{G(w)}{\rho_k(w)} \right]$$

where  $G(w)$  and  $\rho_k(w)$  is exactly the same as what are defined in the previous theorem, and the center layout is defined by  $sl$ .

*Proposition 5:*  $T > \hat{T}^*$ , where  $\hat{T}^* = \max_{sl} \hat{T}_{sl}$ . Literally, the answer to the TMNTM decision problem is positive.

*Proof:* Assume that  $\hat{T}$  is the maximal one among all possible  $T'$ s, and  $\{\hat{w}\}$  is the corresponding weight assignment. By plugging  $\hat{w}$  into the dual problem of MNL (20)–(23), we can always identify a violated constraint with the oracle that

computes  $\min_k \rho_k(w)$ . For instance, assume that the current sink location is  $i$  and its most loaded neighbor is  $j$  (for which (18) is active). Applying complementary slackness, we have  $\rho_i(\hat{w}) = 1$  (by  $\hat{T} > 0$ ),  $\hat{w}(i) = 0$  (by the fact that (18) is inactive for  $i$  due to assumption C), and  $\hat{w}(j) > 0$  (by the fact that (18) is active for  $j$ ). The potential  $\rho_j(\hat{w})$  is bound to be less than 1, because, by moving the sink from  $i$  to  $j$ , we shorten the length of some paths by  $\hat{w}(j)$  without increasing the length of other paths going through  $i$ . Therefore, we identify that  $\{\hat{w}\}$ , as the dual solution, is infeasible. Consequently, according to the principle of *certificate of optimality*, we know that  $\hat{T}$ , as the primal solution, is not optimal and thus  $T > \hat{T}$ . Let  $\hat{T}^* = \max_{sl} \hat{T}_{sl}$ , we also have  $T > \hat{T}^*$ . ■

*Remarks:* The proof implicitly assumes that  $\min_k \rho_k(w)$  and  $\hat{T}^*$  are computable. As we show in Appendix I, the oracle that computes the minimum “potential”  $\min_k \rho_k(w)$  is NP-complete. At the same time, results in [41] suggest that computing  $\hat{T}^*$  is NP-hard. Therefore, *Proposition 5* serves only for a pure theoretical purpose. Nevertheless, in any practical implementation, the load-balancing effect almost always makes mobile sinks more advantageous (in terms of lifetime) than static sinks.

Another interesting point to be pointed out is that, besides the load-balancing effect that we discuss in Section I, there is another “hidden” benefit from moving the sink: it inherits the data forwarding load from the co-located node (assumption C) and thus saves the energy consumption of that node. We call this *substitution effect*. The mobile node approach [32] has indeed exploited this effect to improve lifetime. While the load balancing effect is the driving force behind a significant lifetime improvement (as we will show in Section IX-C), the substitution effect, as presented in the above proof, makes moving sinks universally superior to keeping them static. As we will explain in Section IX-A, the substitution effect is the only reason that leads to a lifetime improvement in certain (albeit not quite realistic) scenarios.

## IX. NUMERICAL EXPERIMENTS

In this section, we test our interior point algorithm by positioning a single mobile sink in several WSNs of typical topologies. We always assign a homogeneous  $\lambda$ ,  $e^{tx}$ , and  $E$  to all nodes in order to facilitate the interpretation of the results. As a consequence, we can use  $e = e^{tx} + e^{rx}$  to represent the energy spent by each node to forward one byte of data. Without loss of generality, we assume  $\lambda = 1$ ,  $e = 1$ , and  $E = |\mathcal{V}| = n$ . We set  $\epsilon = 0.01$ . We only investigate two metrics, namely lifetime and pause time distribution, in this section and refer to [37] for the evaluation of other metrics.

### A. Line Network

For the line network shown in Fig. 2, it is easy to see that the best (static) sink location is at node 0, which achieves a lifetime of  $2 + 1/m$ . Note that, according to assumption C), the sink inherits the data collection function of the co-located node. Otherwise, if the sink used the co-located node as its gateway to the network, the lifetime would not change with different sink locations because the co-located node would

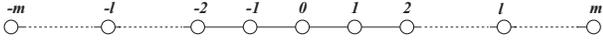


Fig. 2. A line network with  $n = 2m + 1$  nodes. Each node (except nodes  $m$  and  $-m$  that only have one link) has two links with its left and right neighbors.

always take the forwarding load from all the  $2m + 1$  nodes and would thus always “die” first. Now we run our algorithm to show how a mobile sink should be positioned.

As we show in Table I, using a mobile sink can always achieve a longer lifetime than using a static one. However, the relative improvement decreases with the size of a network. The reason comes from the fact that the substitution effect (see

$ \mathcal{V} $	Network Lifetime $T$		
	mobile sink	static sink (optimal)	improvement (%)
11	2.765	2.200	25.67
21	2.578	2.100	22.77
41	2.408	2.050	17.47
81	2.285	2.025	12.82

TABLE I

COMPARING THE ACHIEVABLE LIFETIME BETWEEN USING A MOBILE SINK AND A STATIC SINK (AT ITS OPTIMAL POSITION) IN LINE NETWORKS.

Section VIII-B) is the only cause of the lifetime improvement. In a line network, moving a sink does not lead to load balancing, because it can be easily seen that moving the sink only results in an increase of load for some nodes without lightening others’. This is mainly due to the lack of alternative routing paths between an  $s$ - $t$  pair. Therefore, the lifetime improvement is only brought in by the substitution effect, whose absolute quantity grows only sub-linearly with the network size. This experiment further supports our statement in Section VIII-B: in scenarios where the number of alternative paths between an  $s$ - $t$  pair can be small, it is the substitution effect that makes moving sinks universally superior to keeping them static.

In Fig. 3 (a), we also show the trace of the mobile sink. It

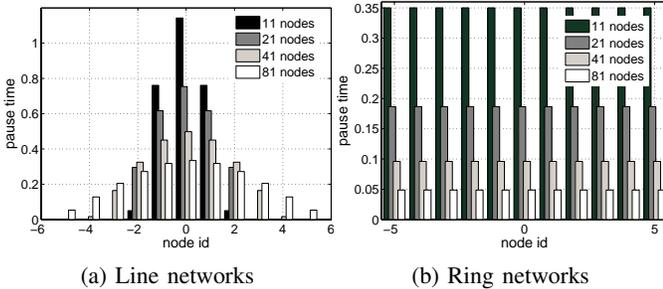


Fig. 3. The pause time distribution of a mobile sink in line and ring networks. For the line networks, only non-zero values are shown. For the ring networks, we zoom to the section between nodes -5 and 5.

can be immediately seen that the larger the network size is, the shorter the pause times near node 0 (and thus the longer the pause times far from node 0). The reason is that, when the network grows in size, it appears (to a centered node)

more and more like a ring. For ring networks (Section IX-B), the sink pauses at every node for the same amount of time. Therefore, a larger line network tends to have a more “spread” pause time distribution.

### B. Ring Network

For the ring network shown in Fig. 4, the achievable lifetime by a static sink is again  $2 + 1/m$ , but it can be obtained by putting the sink at any node, due to the symmetry of such a network. The relative improvement is converging to 100%

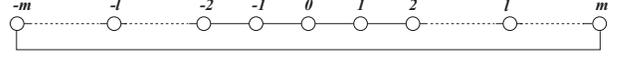


Fig. 4. A ring network with  $n = 2m + 1$  nodes. Each node has two links with its left and right neighbors.

with an increasing network size (Table II). There is no surprise here because the traffic load is fully averaged among all nodes. This averaging effect can be also seen in Fig. 3 (b) (where the pause time distribution is illustrated); the sink pauses at every node for the same amount of time  $t = T/(2m + 1)$ .

$ \mathcal{V} $	Network Lifetime $T$		
	mobile sink	static sink	improvement (%)
11	3.851	2.200	75.05
21	3.920	2.100	86.66
41	3.940	2.050	92.18
81	3.945	2.025	94.81

TABLE II

COMPARING THE ACHIEVABLE LIFETIME BETWEEN USING A MOBILE SINK AND A STATIC SINK IN RING NETWORKS.

### C. Grid Network

For grid networks on  $\sqrt{n} \times \sqrt{n}$  lattices, the maximum achievable lifetime by a static sink is  $n/(\lceil(n - 5)/4\rceil + 1)$ , because the lifetime is maximized if the forwarding load is balanced among the 4 neighbors of the sink. This lifetime can be obtained by putting the sink at the network center (if  $\sqrt{n}$  is odd) or at any of the four nodes close to the center (if  $\sqrt{n}$  is even). While this lifetime is converging to 4 when  $n \rightarrow \infty$ , the lifetime achieved by a mobile sink increases dramatically with the network size (Table III). For small-size networks (e.g.,  $|\mathcal{V}| = 9$  in Table III), the substitution effect dominates the load balancing effect, so the relative improvement is small. With an increasing network size, the number of alternative paths between an  $s$ - $t$  pair is also increasing. Consequently, the load balancing effect becomes increasingly remarkable and thus produces significant improvements on the lifetime.

Compared with the results of [18] (in particular Table I in that paper), our results in Table III exhibit a further increase of 10% to 75% in the lifetime.<sup>11</sup> A straightforward comparison

<sup>11</sup>The only counterexample is when  $|\mathcal{V}| = 16$ . However, this seems to be an outlier case in [18], as it is not compliant with the monotonic increase in the lifetime improvements, which can be easily observed in Table III and Table I in [18].

$ \mathcal{V} $	Network Lifetime $T$		
	mobile sink	static sink (optimal)	improvement (%)
9	5.331	4.500	18.47
16	6.509	4.000	62.72
25	8.146	4.167	95.51
49	11.09	4.084	171.7
81	14.08	4.050	247.6
121	17.07	4.033	323.2
144	18.71	4.000	376.8
169	20.26	4.024	403.6
196	21.75	4.000	443.6
225	23.29	4.018	479.7
289	26.33	4.014	555.9

TABLE III

COMPARING THE ACHIEVABLE LIFETIME BETWEEN USING A MOBILE SINK AND A STATIC SINK IN GRID NETWORKS.

with [20] is not possible because random networks on a square lattice are considered there. However, it might make sense to note the significant difference between the achievable improvements: the approach in [20] achieves an improvement of 24.5% for  $|\mathcal{V}| = 100$ , whereas ours already achieves 10 times of that value for  $|\mathcal{V}| = 81$ . These comparisons confirm our statement in Section II that, in both [18] and [20], further optimizations are still possible.

We illustrate the pause time distribution in four networks in Fig. 5. Our observation is that the sink tends to move toward the periphery of a network with an increasing  $n$ . The intuition

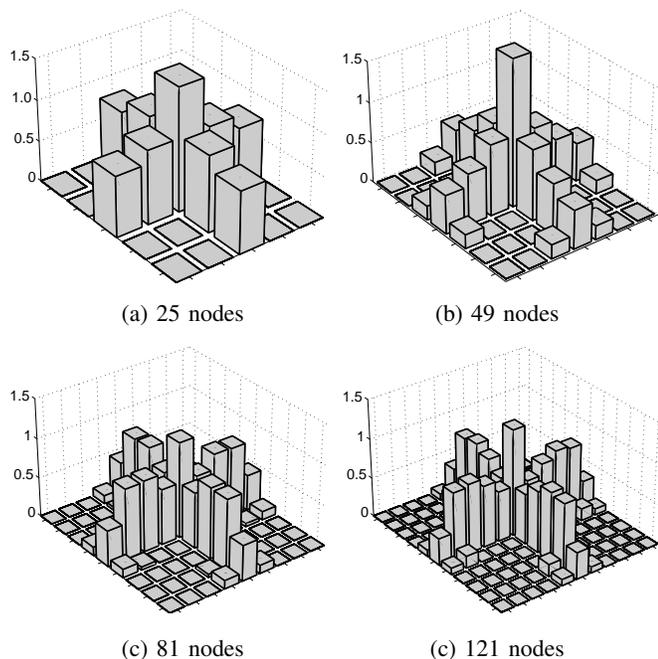


Fig. 5. The pause time distribution of a mobile sink in grid networks. The  $z$ -axis represents the pause time.

is that, for a 3D grid on a sphere, the sink should pause everywhere with same time period (analogous to a ring in 2D). Therefore, the pause times spread out when the network grows in size and thus appears to a centered node more and more

like a sphere grid (analogous to a line in 1D). This observation also corroborates the result in [19]: the network periphery, as a sink moving trace, is asymptotically optimal. Note that we investigate in [19] the asymptotical case where the node density is large enough to make the necessary radio ranges infinitely small. In that case, the shortest paths between any  $s$ - $t$  pair happen to be straight lines.

#### D. Random Network

We also perform experiments on random networks (nodes uniformly distributed within a square). Fig. 6 illustrates the pause time distribution in two such networks. A direct observa-

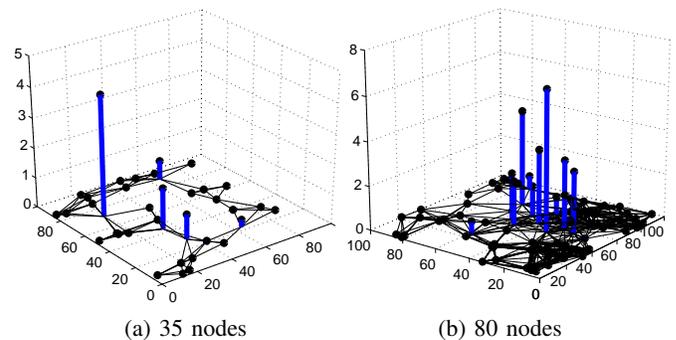


Fig. 6. The pause time distribution of a mobile sink in random networks. The  $z$ -axis represents the pause time.

tion is that the sink tends to pause at the nodes whose degrees are high. This is intuitive because the more neighbors a node has the more balanced load can be achieved by co-locating the sink with it. The relative improvements are 30.52% and 24.83% in networks of 35 and 80 nodes, respectively. As we fix the area of the square, increasing network size leads to higher connectivity. Therefore, these experiments also show that higher connectivity disfavors sink mobility. This is again intuitive because, in the extreme case, the network with a fully connected topology needs no sink mobility.

## X. CONCLUSION

In this paper, we have built a unified framework to analyze the *maximizing network lifetime* (MNL) problem in WSNs. Our investigation, based on a graph model, jointly considers sink mobility and routing for lifetime maximization. We have formally proved the NP-hardness of the MNL involving multiple mobile sinks. We have then identified the sub-problem that has a potential to guide routing protocol designs in practice. In particular, we have developed an efficient algorithm to solve the MNL problem involving only a single mobile sink; we have further generalized the algorithm to approximate the general MNL problem. In addition, using the duality theory, we have proved that, theoretically, moving the sinks is always better than keeping them static. Finally, we have illustrated the benefit of using a mobile sink by applying our algorithm to a set of typical topological graphs.

As for future directions, we are in the process of engineering the routing protocol that we proposed to support sink mobility [37] in order to approach the upper bound characterized in this

paper. We are also working on an online algorithm, derived from the approximation algorithm, to guide sink mobility in the face of network dynamics.

## REFERENCES

- [1] J.-H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks," in *Proc. of the 19th IEEE INFOCOM*, 2000.
- [2] K. Kar, M. Kodialam, T. Lakshman, and L. Tassiulas, "Routing for Network Capacity Maximization in Energy-constrained Ad-hoc Networks," in *Proc. of the 22nd IEEE INFOCOM*, 2003.
- [3] A. Sankar and Z. Liu, "Maximum Lifetime Routing in Wireless Ad-hoc Networks," in *Proc. of the 23rd IEEE INFOCOM*, 2004.
- [4] L. Li, J. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer, "Analysis of A Cone-based Distributed Topology Control Algorithm for Wireless Multi-hop Networks," in *Proc. of the 20th ACM PODC*, 2001.
- [5] N. Li and J. Hou, "Topology Control in Heterogeneous Wireless Networks: Problems and Solutions," in *Proc. of the 23rd IEEE INFOCOM*, 2004.
- [6] X.-Y. L. W.-Z. Song and W. Wang, "A Unified Energy-Efficient Topology for Unicast and Broadcast," in *Proc. of the 11th ACM MobiCom*, 2005.
- [7] C.-F. Chiasserini and M. Garetto, "Modeling the Performance of Wireless Sensor Networks," in *Proc. of the 23rd IEEE INFOCOM*, 2004.
- [8] B. Deb and B. Nath, "On the Node-Scheduling Approach to Topology Control in Ad Hoc Networks," in *Proc. of the 6th ACM MobiHoc*, 2005.
- [9] R. Iyengar, K. Kar, and S. Banerjee, "Low-coordination Topologies for Redundancy in Sensor Networks," in *Proc. of the 6th ACM MobiHoc*, 2005.
- [10] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An Application-specific Protocol Architecture for Wireless Microsensor Networks," *IEEE Trans. on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [11] L. Bao and J. Garcia-Luna-Aceves, "Topology Management in Ad Hoc Networks," in *Proc. of the 4th ACM MobiHoc*, 2003.
- [12] T. Shu, M. Krunz, and S. Vrudhula, "Power Balanced Coverage-Time Optimization for Clustered Wireless Sensor Networks," in *Proc. of the 6th ACM MobiHoc*, 2005.
- [13] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: Mobeling a Three-tier Architecture for Sparse Sensor Networks," in *Proc. of the 1st IEEE SNPA*, 2003.
- [14] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks," in *Proc. of the 2nd IEEE IPSN*, 2003.
- [15] A. Kansal, A. Somasundara, D. Jea, M. Srivastava, and D. Estrin, "Intelligent Fluid Infrastructure for Embedded Networks," in *Proc. of the 2nd ACM/USENIX MobiSys*, 2004.
- [16] M. Younis, M. Bangad, and K. Akkaya, "Base-Station Repositioning for Optimized Performance of Sensor Networks," in *Proc. of the 58th IEEE VTC*, 2003.
- [17] S. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy Efficient Schemes for Wireless Sensor Networks with Multiple Mobile Base Stations," in *Proc. of IEEE Globecom*, 2003.
- [18] Z. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli, "Exploiting Sink Mobility for Maximizing Sensor Networks Lifetime," in *Proc. of the 38th HICSS*, 2005.
- [19] J. Luo and J.-P. Hubaux, "Joint Mobility and Routing for Lifetime Elongation in Wireless Sensor Networks," in *Proc. of the 24th IEEE INFOCOM*, 2005.
- [20] I. Papadimitriou and L. Georgiadis, "Maximum Lifetime Routing to Mobile Sink in Wireless Sensor Networks," in *Proc. of the 13th IEEE SofiCom*, 2005.
- [21] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A Two-tier Data Dissemination Model for Large Scale Wireless Sensor Networks," in *Proc. of the 8th ACM MobiCom*, 2005.
- [22] H. Kim, T. Abdelzaher, and W. Kwon, "Minimum Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks," in *Proc. of the 1st ACM SenSys*, 2003.
- [23] P. Baruah, R. Ugaonkar, and B. Drishnamachari, "Learning Enforced Time Domain Routing to Mobile Sinks in Wireless Sensor Fields," in *Proc. of the 1st IEEE EmNets*, 2004.
- [24] A. Somasundara, A. Ramamoorthy, and M. Srivastava, "Mobile Element Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic Deadlines," in *Proc. of the 25th IEEE RTSS*, 2004.
- [25] Y. Gu, D. Bozdağ, E. Ekici, F. Ozguner, and C. Lee, "Partitioning-Based Mobile Element Scheduling in Wireless Sensor Networks," in *Proc. of the 2nd IEEE SECON*, 2005.
- [26] M. Ma and Y. Yang, "SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1478–1488, 2007.
- [27] I. Chatzigiannakis, A. Kinalis, S. Nikolettseas, and J. Rolim, "Fast and Energy Efficient Sensor Data Collection by Multiple Mobile Sinks," in *Proc. of the 5th ACM MobiWAC*, 2007.
- [28] S. Baek and G. de Veciana, "Spatial Model for Energy Burden Balancing and Data Fusion in Sensor Networks Detecting Bursty Events," *IEEE Trans. on Information Theory*, vol. 53, no. 10, pp. 3615–3628, 2007.
- [29] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. on Networking*, vol. 10, no. 4, pp. 477–486, 2002.
- [30] A. Somasundara, A. Kansal, D. Jea, D. Estrin, and M. Srivastava, "Controllably Mobile Infrastructure for Low Energy Embedded Networks," *IEEE Trans. on Mobile Computing*, vol. 5, no. 8, pp. 958–973, 2006.
- [31] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, and Z. Wang, "Controlled Sink Mobility for Prolonging Wireless Sensor Networks Lifetime," *Springer Wireless Networks (to appear)*, 2007.
- [32] W. Wang, V. Srinivasan, and K.-C. Chua, "Using Mobile Relays to Prolong the Lifetime of Wireless Sensor Networks," in *Proc. of the 11th ACM MobiCom*, 2005.
- [33] —, "Trade-offs Between Mobility and Density for Coverage in Wireless Sensor Networks," in *Proc. of the 13th ACM MobiCom*, 2007.
- [34] T. Melodia, D. Pompili, V. Gungor, and I. Akyildiz, "A Distributed Coordination Framework for Wireless Sensor and Actor Networks," in *Proc. of the 6th ACM MobiHoc*, 2005.
- [35] G. Sharma and R. Mazumdar, "Hybrid Sensor Networks: A Small World," in *Proc. of the 6th ACM MobiHoc*, 2005.
- [36] V. Shnayder, M. Hempstead, B. Chen, G. Allen, and M. Welsh, "Simulating the Power Consumption of Large-Scale Sensor Network Applications," in *Proc. of the 2nd ACM SenSys*, 2004.
- [37] J. Luo, J. Panchar, M. Piórkowski, M. Grossglauser, and J.-P. Hubaux, "MobiRoute: Routing towards a Mobile Sink for Improving Lifetime in Sensor Networks," in *Proc. of the 2nd IEEE/ACM DCOSS*, 2006.
- [38] D. Blough and P. Santi, "Investigating Upper Bounds on Network Lifetime Extension for Cell-based Energy Conservation Techniques in Stationary Ad Hoc Networks," in *Proc. of the 8th ACM MobiCom*, 2002.
- [39] J. Luo, A. Iyer, and C. Rosenberg, "Throughput-Lifetime Tradeoffs in Multihop Wireless Networks under a Realistic Interference Model," in *Proc. of the 45th Allerton Conference*, 2007.
- [40] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: Freeman, 1979.
- [41] A. Bogdanov, E. Maneva, and S. Riesenfeld, "Power-aware Base Station Positioning for Sensor Networks," in *Proc. of the 23rd IEEE INFOCOM*, 2004.
- [42] N. Garg and J. Könemann, "Faster and Simpler Algorithms for Multi-commodity Flow and other Fractional Packing Problems," in *Proc. of the 38th IEEE FOCS*, 1997.
- [43] R. Bland, D. Goldfarb, and M. Todd, "The Ellipsoid Method: A Survey," *Operations Research*, vol. 29, no. 6, pp. 1039–1091, 1981.
- [44] F. Shahrokhi and D. Matula, "The Maximum Concurrent Flow Problem," *J. ACM*, vol. 37, no. 2, pp. 318–334, 1990.
- [45] L. Ford and D. Fulkerson, *Flows in Networks*. Princeton, N.J.: Princeton University Press, 1962.
- [46] R. Floyd, "Algorithm 97: Shortest Path," *Commun. ACM*, vol. 5, no. 6, p. 345, 1962.
- [47] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit, "Local Search Heuristics for k-Median and Facility Location Problems," *SIAM J. on Computing*, vol. 33, no. 3, pp. 544–562, 2004.
- [48] G. Nemhauser and L. Wolsey, *Integer and Combinatorial Optimization*. New York: John Wiley & Sons, 1988.

## APPENDIX I

### AN ALTERNATIVE PROOF OF PROPOSITION 1

Our proof of *Proposition 1* is based on the principle of *equivalence of separation and optimization* [48]. *Theorem 3.3* of [48] states that a linear programming problem is solvable in polynomial time (by the *ellipsoid* algorithm [43]) if and only if there exists a *separation oracle* that has a polynomial complexity. The oracle identifies a violated constraint or verifies

that there is no such a constraint. The theorem also implies that the linear programming problem is NP-hard if the separation oracle is NP-complete, because there is a polynomial-time reduction of the separation oracle to the linear programming problem.

Let us re-formulate the MNL problem into a Path-Flow form by dropping the non-essential constraint (5):

$$\text{Maximize } \sum_k t_k \quad (25)$$

$$\sum_{p \in P_{ik}} f(p) - \lambda_i t_k = 0 \quad \forall i, k \quad (26)$$

$$\sum_k \sum_{p \in P_i^k} f(p)(e_i^{tx} + e^{rx}) - E_i \leq 0 \quad \forall i \quad (27)$$

$$f(p), t_k \geq 0 \quad \forall p, k \quad (28)$$

where  $p$  refers to a certain path and  $f(p)$  is the flow that goes through  $p$ . Furthermore,  $P_{ik}$  stands for the set of paths between node  $i$  and one of the  $m$  sinks during the  $k$ th epoch and  $P_i^k$  represents the set of paths that go through node  $i$  in the  $k$ th epoch.

The dual problem is given by:

$$\text{Minimize } \sum_i E_i w(i) \quad (29)$$

$$\sum_i \lambda_i W(i, k) \geq 1 \quad \forall k \quad (30)$$

$$\sum_{j \in p, p \in P_{ik}} w(j)(e_j^{tx} + e^{rx}) - W(i, k) \geq 0 \quad \forall i, k \quad (31)$$

$$w(j) \geq 0 \quad \forall i, j, k \quad (32)$$

where  $W(i, k)$  is the weight assigned to a ‘‘commodity’’ (data flow injected at a node in our case) from node  $i$  to one of the  $m$  sinks during the  $k$ th epoch and  $w(j)$  is the weight assigned to a node  $j$ .

The separation oracle for the dual problem checks if the constraints in (30) and (31) are violated. It is equivalent to verify if the following constraint is violated:

$$\min_k \left[ \sum_i \lambda_i \sum_{j \in \min\{p | p \in P_{ik}\}} w(j)(e_j^{tx} + e^{rx}) \right] \geq 1$$

Now, by taking  $\omega(i) = \lambda_i$ ,  $l(i) = w(i)(e_i^{tx} + e^{rx})$ ,  $K = |\mathcal{S}|$ , and  $d(i) = \sum_{j \in \min\{p | p \in P_{ik}\}} w(j)(e_j^{tx} + e^{rx})$ , the minimization problem leads to the following decision problem:

INSTANCE: A graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , a weight assignment  $\omega(i): \mathcal{V} \rightarrow \mathbb{R}_0^+$ , a length assignment  $l(i): \mathcal{V} \rightarrow \mathbb{R}_0^+$ , positive integer  $K \leq |\mathcal{V}|$ , and positive rational number  $B$ .

QUESTION: Is there a set  $\mathcal{P}$  of  $K$  ‘‘points on  $\mathcal{G}$ ’’ such that  $\sum_{i \in \mathcal{V}} \omega(i) \cdot d(i) \leq B$ ?

The problem, which is known as the p-median problem [40], is NP-complete. Therefore, our arguments at the beginning of this section suggest that MNL is NP-hard. Q.E.D.

Since we explicitly show a polynomial reduction of the p-median problem to our MNL problem in the above proof, any solution (or approximation) to the p-median problem can be directly applied to MNL.

## APPENDIX II PROOF OF PROPOSITION 3

Our proof of *Proposition 3* is based on the proof of Garg and Könemann [42]. We have the following relation from (24):

$$G(w_i) = G(w_{i-1}) + \epsilon \cdot \rho(w_{i-1})$$

Since  $\frac{G(w)}{\rho(w)} > \beta$  and  $G(w_0) = n\delta$ , for the  $i$ th iteration:

$$G(w_i) \leq G(w_{i-1})(1 + \epsilon/\beta) \leq G(w_{i-1})e^{\epsilon/\beta} \leq n\delta e^{i\epsilon/\beta}$$

Suppose that the procedure stops at  $t$ th iteration for which  $G(w_t) \geq 1$ , we have:

$$1 \leq G(w_t) \leq n\delta e^{t\epsilon/\beta} \Rightarrow \frac{\beta}{t} \leq \frac{\epsilon}{\ln(n\delta)^{-1}} \quad (33)$$

Note that  $t$  is not necessarily the lifetime we achieve, because it is possible that the flow already violates the constraints, namely (18), of the primal problem.<sup>12</sup> Let us consider a certain node  $l$ . For every  $E_l/(e_l^{tx} + e^{rx})$  units of flow routed through  $l$ , the weight  $w(l)$  is increased by at least  $1 + \epsilon$ . In addition, the weight  $w(l)$  is increased by at most  $1 + \epsilon$  each iteration due to the assumption that  $\sum_i \lambda_i \leq E_l/(e_l^{tx} + e^{rx})$ . Since  $w_0(l) = \delta/E_l$  and  $w_{t-1}(l) < 1/E_l$  (due to the fact that  $G(w_{t-1}) < 1$ ), the total amount of flow through  $l$  during the first  $t$  iteration is strictly less than  $\frac{E_l}{e_l^{tx} + e^{rx}} \log_{1+\epsilon} \frac{1+\epsilon}{\delta}$ . Therefore, (18) can be violated by at most a multiple of  $\log_{1+\epsilon} \frac{1+\epsilon}{\delta}$ , and thus  $t \cdot \log_{1+\epsilon}^{-1} \frac{1+\epsilon}{\delta}$  gives a feasible primal solution. If we denote the ratio of the dual and the primal solutions by  $\gamma$  and apply the bound on  $\beta/t$  given in (33), we have:

$$\gamma = \frac{\beta}{t} \log_{1+\epsilon} \frac{1+\epsilon}{\delta} \leq \frac{\epsilon \log_{1+\epsilon} \frac{1+\epsilon}{\delta}}{\ln(n\delta)^{-1}} = \frac{\epsilon}{\ln(1+\epsilon)} \cdot \frac{\ln \frac{1+\epsilon}{\delta}}{\ln(n\delta)^{-1}}$$

For  $\delta = (1 + \epsilon)[(1 + \epsilon)n]^{-1/\epsilon}$ , we have:

$$\gamma \leq \frac{\epsilon}{(1 - \epsilon) \ln(1 + \epsilon)} \leq \frac{\epsilon}{(1 - \epsilon)(\epsilon - \epsilon^2/2)} < (1 - \epsilon)^{-2}$$

which means that, if the maximal lifetime is  $T$  ( $= \beta$  according to strong duality), the algorithm achieves a lifetime  $\hat{T} = t \cdot \log_{1+\epsilon}^{-1} \frac{1+\epsilon}{\delta} > (1 - \epsilon)^2 T \geq (1 - 2\epsilon)T$ .

The time complexity can be obtained by weak duality:

$$1 \leq \gamma = \frac{\beta}{t} \log_{1+\epsilon} \frac{1+\epsilon}{\delta}$$

and thus the number of iterations in our algorithm  $t$  is less than  $\beta \log_{1+\epsilon} \frac{1+\epsilon}{\delta}$ . Given  $\delta = (1 + \epsilon)[(1 + \epsilon)n]^{-1/\epsilon}$ , we have  $t \leq \lceil \frac{\beta}{\epsilon} \log_{1+\epsilon}(1 + \epsilon)n \rceil$ . Since each iteration involves one call to the oracle, the actual time complexity is  $\lceil \frac{\beta}{\epsilon} \log_{1+\epsilon}(1 + \epsilon)n \rceil \cdot T_{\text{oracle}}$ .

Notice that the run time depends on  $\beta$ , which can be very large if  $E_l \gg e_l^{tx} + e^{rx}$  for each node  $l$  (since  $\beta \leq \min_l [E_l/(e_l^{tx} + e^{rx})\lambda_l]$ ). Fortunately, we do not need to solve a full-scale problem. If we scale down every  $E_l$  by a factor of  $\phi$ , we actually scale down the dual objective  $\sum_l E_l w_l$  (and thus the lifetime  $\sum_k t_k$ ) by  $\phi$ . Therefore, we choose the largest  $\phi$  without violating  $\sum_i \lambda_i \leq E_l/(e_l^{tx} + e^{rx})$ . So  $\phi =$

<sup>12</sup>Constraints (17) are always satisfied because the iteration procedure increases  $t_k$  by one only if  $\sum_i \lambda_i$  units of commodities from all nodes are admitted.

$E_{\hat{l}}/[(e_j^{tx} + e^{rx}) \sum_i \lambda_i]$  where  $\hat{l} = \arg \min_l E_l/[(e_l^{tx} + e^{rx})]$ . Now  $\beta = \Theta(n)$  and thus the time complexity becomes  $\Theta(n \log n) \cdot T_{\text{oracle}}$ . Note that the solution of this reduced-scale problem should be scaled up by  $\phi$  to obtain the real solution. Q.E.D.

### APPENDIX III PROOF OF PROPOSITION 4

We have already shown in Appendix I that a p-median oracle is indeed an oracle that computes

$$\rho(w) \equiv \min_k \rho_k(w) = \min_k \left( \sum_i \lambda_i \sum_{j \in \min\{p | p \in P_{ik}\}} w(j)(e_j^{tx} + e^{rx}) \right)$$

Now, suppose we have an  $\alpha$ -approximation for the oracle, it means that the oracle always returns  $\tilde{\rho}(w) \leq \alpha \rho(w)$  for  $\alpha > 1$ . Again, we have the following relation from (24):

$$G(w_i) = G(w_{i-1}) + \epsilon \cdot \tilde{\rho}(w_{i-1})$$

Since  $\frac{G(w)}{\rho(w)} > \beta$  (hence  $\frac{G(w)}{\tilde{\rho}(w)} > \tilde{\beta}$  where  $\tilde{\beta} = \frac{\beta}{\alpha}$ ) and  $G(w_0) = n\delta$ , for the  $i$ th iteration:

$$G(w_i) \leq G(w_{i-1})(1 + \epsilon/\tilde{\beta}) \leq G(w_{i-1})e^{\epsilon/\tilde{\beta}} \leq n\delta e^{i\epsilon/\tilde{\beta}}$$

Suppose that the procedure stops at  $t$ th iteration for which  $G(w_t) \geq 1$ , we have:

$$1 \leq G(w_t) \leq n\delta e^{t\epsilon/\tilde{\beta}} \Rightarrow \frac{\tilde{\beta}}{t} \leq \frac{\epsilon}{\ln(n\delta)^{-1}}$$

The derivation of the lower bound for the primal solution is not affected by using the  $\alpha$ -approximation oracle, it follows the same line as the proof in Appendix II and the lower bound is again  $t \cdot \log_{1+\epsilon}^{-1} \frac{1+\epsilon}{\delta}$ . Finally, we have

$$\gamma = \frac{\beta}{t} \log_{1+\epsilon} \frac{1+\epsilon}{\delta} = \frac{\alpha\tilde{\beta}}{t} \log_{1+\epsilon} \frac{1+\epsilon}{\delta} \leq \frac{\alpha}{(1-\epsilon)^2}$$

This gives an upper bound for the gap between primal and dual solutions, which is the required result: if the maximal lifetime is  $T$  ( $= \beta$  according to strong duality), the algorithm achieves a lifetime  $\hat{T} = t \cdot \log_{1+\epsilon}^{-1} \frac{1+\epsilon}{\delta} > \frac{(1-\epsilon)^2}{\alpha} T$ . Q.E.D.

PLACE  
PHOTO  
HERE

**Jun Luo** received his B.S. and M.S. degrees both in electrical engineering from Tsinghua University, Beijing, PRC, in 1997 and 2000, respectively, and his PhD degree in computer science from EPFL (Swiss Federal Institute of Technology in Lausanne), Lausanne, Switzerland, in 2006.

Currently, he is a post-doctoral fellow of the Department of Electrical and Computer Engineering, University of Waterloo, in Waterloo, Canada. His research interests include wireless networking and distributed systems, network modeling and performance analysis, as well as network security. He is a member of ACM and IEEE. For more information, check <http://www.eng.uwaterloo.ca/~j7luo/>.

PLACE  
PHOTO  
HERE

**Jean-Pierre Hubaux** joined the faculty of EPFL (Swiss Federal Institute of Technology in Lausanne) in 1990; he was promoted to full professor in 1996. His research activity is focused on mobile networking and computing, with a special interest in fully self-organized wireless ad hoc networks. In particular, he has performed research on co-operation aspects, security, power efficiency, and distributed algorithms for ad hoc and sensor networks.

During the last few years, he has been strongly involved in the definition and launching phases of a new National Competence Center in Research named "Mobile Information and Communication Systems" (NCCR/MICS), see <http://www.terminodes.org>. He served as the general chair for the Third ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002), held in June on the EPFL campus. He is an Associate Editor of IEEE Transactions on Mobile Computing and of the Elsevier Journal on Ad Hoc Networks. Within his first year at EPFL, he defined the first curriculum in Communication Systems. From October 1999 until September 2001, he was the first chairman of the Communication Systems Department. He has held visiting positions at the IBM T.J. Watson Research Center and at the University of California at Berkeley. He has published more than 60 papers in the area of networking.

In a former life, he spent 10 years in France with Alcatel, where he was involved in R&D activities, mostly in the area of switching systems architecture and software. He is a senior member of the IEEE. For more information, check <http://people.epfl.ch/jean-pierre.hubaux>.