Delay constrained placement of mobile data collectors in underwater acoustic sensor networks

Waleed Alsalih , Hossam Hassanein, and Selim Akl.
School of Computing at Queen's University.
{waleed,hossam,akl}@cs.queensu.ca.

***Abstract*—We propose a scheme for routing and placement of mobile data collectors in Underwater Acoustic Sensor Networks (UASNs). The proposed scheme maximizes the lifetime of the network with an upper bound on the maximum delay. We assume a 3D architecture, in which on-the-surface data collectors gather data from underwater sensors and relay them to an on-shore sink. We divide the lifetime of the network into fixed length rounds and move the data collectors to new locations at the beginning of each round.**

This problem is formulated as an Integer Linear Program (ILP), and we use an ILP solver to find the optimal placement of data collectors together with the multi-hop routing paths to deliver data from underwater sensors to data collectors. To the best of our knowledge, this is the first attempt towards the placement of data collectors in a 3D environment with delay constraints. When compared with other schemes, our scheme has shown the capability to achieve longer lifetime and shorter delay.

I. INTRODUCTION

Because of their prominent potential for environmental monitoring and oceanographic data gathering, underwater sensor networks have recently gained the attention of researchers in wired and wireless networks. Most of the work in underwater sensing and data collection is based on wired networks (e.g., the work in [1]), which demand high cost and significant engineering effort. Accordingly, interest has been directed to acoustic networks because of their potential to provide low-cost, distributed underwater monitoring [10] [11].

Underwater acoustic sensor devices are battery-operated and, hence, untethered in terms of both power and communication. This enables a new generation of large-scale, untethered, unattended sensor networks and simplifies the deployment process. However, as a result of the limited energy supply of sensor nodes, prolonging the lifetime of UASNs has become crucial for UASNs to deliver their full potential and to enable this variety of fundamental applications. A few Medium Access Control (MAC) and routing protocols (e.g., the work in [10],[8], and [13]) have been proposed to alleviate the effect of energy limitations at individual sensor nodes. However, they fail to solve topology-related problems; with multi-hop communication, nodes near data collectors become bottlenecks and with single-hop communication, nodes far from data collectors deplete their energy very quickly. According to our work in [5], it has been shown that such topology-related problems are hard to overcome when data collectors are stationary, and we proposed a scheme that exploits data collector mobility to make a uniform energy consumption across the network and to prolong its lifetime. The work in [5], however, does not make any consideration to the delay a data packet may encounter before arriving to the data collector (i.e., a delay tolerant placement problem). Therefore, it suits delay tolerant applications rather than real-time ones, such as a Tsunami warning system. In this paper we extend our work in [5] and propose a scheme that prolongs the lifetime of the network and guarantees an upper bound on the delay (i.e., a delay constrained placement problem).

Our scheme divides the lifetime of the network into fixed length rounds (e.g., hours, days, or weeks) and moves the data collectors, which can be Autonomous Underwater Vehicles (AUVs), to new locations at the beginning of each round. Relocation of data collectors will require new routing paths to deliver data from sensor nodes to data collectors. In this paper the problem of finding the optimal routing and placement of data collectors is formulated as an ILP, and we use an ILP solver with a constant time limit to find near-optimal solutions. The benefits of data collectors' mobility in terrestrial sensor networks have been explored and some promising mobility planning schemes have been proposed [7][6][9]. However, those schemes have aimed at locating the data collectors at predefined spots in the sensing field or at the boundary of the network. These constraints do not apply to UASNs where data collectors are free to go virtually anywhere on the surface of the ocean. Moreover, the three-dimensional nature of UASNs imposes another challenge to the problem.

To the best of our knowledge, this paper presents a pioneering effort for the placement of mobile data collectors in three-dimensional UASNs with delay constraints. The novelty of this work stems from discretizing the infinite search space of data collector locations and from the solid mathematical programming formulation of the problem. Furthermore, we are not aware of any mobile data collector (base station) placement scheme, even in terrestrial sensor networks, that maximizes the lifetime of the network under delay constraints.

The remainder of the paper is organized as follows. Section II describes the model of the system. In Section III, we present our scheme. Section IV shows the experimental results. Finally, in Section V, we conclude by summarizing the contributions and pointing out some related future research directions.

II. SYSTEM MODEL

We consider an UASN consisting of N sensor nodes and R data collectors. Each sensor node collects data from the

978-1-4244-2413-9/08/$25.00 ©2008 IEEE

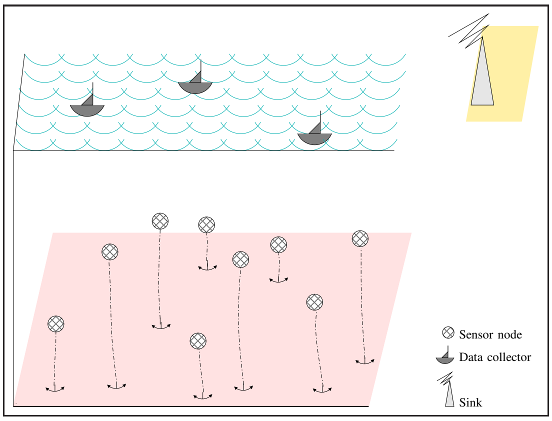

Fig. 1. 3D architecture for UASNs (adapted from [4]).

surrounding environment and sends the collected data to one of the collectors either directly or through other nodes (i.e., multi-hop communication). We follow the three-dimensional architecture proposed in [4]. In this architecture, data collectors are equipped with a radio transceiver to communicate with an on-shore sink, and an acoustic transceiver to communicate with underwater sensors. While data collectors stay on the surface to keep the link with the sink, sensor nodes, which are equipped with acoustic transceivers, float at different depths to carry out the sensing mission. This architecture is shown in Fig. 1.

We assume that the transmission range of an acoustic transceiver can be adjusted to a constant number of levels (i.e., variable, discrete transmission ranges). Thereby, when a sensor node wants to send data to another node, it uses the smallest transmission range that can reach the destination. The topology of the network is modeled as a weighted graph $G = (V, E)$, where $V = \{n_0, n_1, ..., n_{N-1}\}$ is the set of N sensor nodes, and $(i, j) \in E$, if sensor nodes n_i and n_j can communicate with each other directly (i.e., their separation is not more than the maximum transmission range). $W(i, j)$ denotes the weight of the link (i, j) which is the Euclidean distance between sensor nodes n_i and n_j; the energy consumption and the delay over the link (i, j) are functions of $W(i, j)$. Each sensor node n_i has a data generation rate G_i. G_i is the number of data units generated by node n_i per time unit.

For the sake of simplicity, we adopt a discrete delay model which can be tuned to meet any desired accuracy. This delay model is described in details in Section III. Moreover, due to the relatively low speed of sound in water (approximately 1500 m/second), the propagation delay dominates the transmission and queueing delays. Therefore, we assume that transmission and queueing delays are negligible.

Our scheme is independent of the underlying MAC protocol. We assume a capacity limit for each sensor node which limits the number of data units that can be transmitted by a sensor node during one round. The capacity of a sensor node n_i is denoted by C_i. This parameter can be adjusted to comply with any constraints imposed by any MAC protocol. While this capacity constraint is also applicable to communication links, to simplify the presentation of the paper, we will limit ourselves to capacity constraints on individual nodes.

In this paper we assume that data generation rates, residual energy, and locations of all sensor nodes are known in advance. We also assume that data collectors are not energy constrained as they can be easily recharged [12]. Without loss of generality, we define the lifetime of the network as the time until the first sensor node dies. Yet other definitions (e.g., the time until a particular proportion of the sensors die) can be equally used in our scheme.

III. Delay constrained placement and routing

Even though energy is a critical resource in UASNs, some time-sensitive applications demand a balance between energy saving and the provisioned Quality of Service (QoS). Delay is a QoS metric that can be of significant importance in many real-time applications (e.g., intruder detection, Tsunami warning, ...etc). In the scheme we proposed in [5], the sole objective is to maximize the lifetime of the network without any delay constraints, i.e., there is no bound on the length of any path between a sensor node and a data collector. Therefore, a data unit might be sent through a longer path in order to avoid sensor nodes with lower residual energy. The scheme we propose here seeks the same goal of maximizing the lifetime of the network, yet it maintains an upper bound on how much time a data unit may spend in its way to a data collector.

A. Delay model and problem definition

The delay component we are considering in this paper is the propagation delay; the propagation delay between two nodes is the time a bit would require to propagate from the source to the destination. It depends on the speed of the link between the two ends. In UASNs, link speed is the speed of sound in water, which is approximately 1500 m/second.

In this paper, we assume for the sake of simplicity that queueing and transmission delays are negligible. Nevertheless, when these delay components are predictable, they can be easily integrated in our scheme.

Delay has a continuous nature; even if two nodes are placed in a bounded area, there is an infinite number of possible values for the delay between them. That is simply because the delay can take any real value between two real numbers, which is an infinite set. Since we are using linear programming, we adopt a discrete delay model that can be tuned to achieve any desired accuracy. This model is described as follows. We define a delay step Δ which is the distance an acoustic signal would travel in the water in one time unit. Then, the discrete delay over a single-hop link (i, j) is $\lceil \frac{W(i,j)}{\Delta} \rceil$. The discrete delay over a multi-hop path is the sum of the discrete delays of single-hop links that constitute that path. It is obvious that Δ (and the time unit) can be made small enough to meet any desired accuracy. In what follows we use W^*_{ij} to denote the discrete delay over a single-hop link between two sensor nodes n_i and n_j, and D^*_{ij} to denote the discrete delay over a single-hop link between a sensor node n_i and a data collector placed at a point p_j (p_j is an element in the complete set which is described below).

Now, we can define the problem as follows:

The lifetime of the network is divided into equal length rounds. At the beginning of each round, find the optimal locations of R data collectors together with the routing paths to deliver the generated data from all sensor nodes to data collectors, such that the discrete delay of a path from a sensor node to a data collector is at most Γ. The objective is to maximize the minimum residual energy at the end of the round. Data collectors can be placed anywhere on the surface of the water.

B. Finding a complete set

Indeed, a data collector may be placed anywhere on the surface of the water as long as it is within the acoustic transmission range of at least one sensor node. Since each sensor node has a discrete number of transmission levels, its transmission range can be viewed as a set of transmission spheres: one sphere for each transmission level. Moreover, for each sensor node we add $\lfloor \frac{MaxRange}{\Delta} \rfloor$ virtual delay spheres, where $MaxRange$ is the maximum transmission range of a sensor node. When these delay spheres are indexed from 1 to $\lfloor \frac{MaxRange}{\Delta} \rfloor$, the radius of the j^{th} sphere is $j\Delta$. Delay spheres partition the space around a sensor node based on the discrete delay. Thereby, a layer, which is the space between two consecutive transmission or delay spheres (see Fig. 2), has the same delay and energy properties, i.e., sending a data unit to any node inside the same layer would consume the same amount of energy and encounter the same discrete delay. Thus, for a given sensor node, placing a data collector at any point inside the same layer has the same effect; what matters is at which layer the data collector is placed and not where exactly inside that layer.

Further, since data collectors are placed on the surface of the water, we should find the intersection of these spheres and the surface of the water. Without loss of generality, we let the plane $z = 0$ represent the surface of the water (where the $z-$axis represents the depth). Then, the intersection of a sphere centered at a point (a, b, c) with a radius of r and the plane $z = 0$ is:

- ϕ, the empty set, if $r^2 - c^2 < 0$.
- A circle centered at a point $(a, b, 0)$ with a radius of $\sqrt{r^2 - c^2}$, if $r^2 - c^2 \geq 0$. We call such a circle a *surface circle*.

Each sensor node will have at most one surface circle for each transmission or delay sphere. And since the placement will be made on the surface of the water, we can focus on these surface circles rather than the transmission and delay spheres, i.e., the problem becomes two-dimensional. So let us say each sensor node n_i has γ_i surface circles $\{\zeta_i^0, \zeta_i^1, ..., \zeta_i^{\gamma_i - 1}\}$.

In [5] we defined the *complete* set as a set of points on the surface of the water, such that there exists an optimal placement in which each data collector is placed at a point in that set. We also devised an algorithm to find a complete set for the delay tolerant problem. However, a set of points $\mathcal{P}$, which is complete for the delay tolerant problem, is not necessarily complete for the delay constrained problem on the same network. That is because the optimal delay tolerant placement embedded in $\mathcal{P}$ may be infeasible for the delay constrained problem possibly because it involves a path that exceeds the maximum delay limit. Therefore, we need to modify our algorithms in [5] to accommodate the delay constraints.

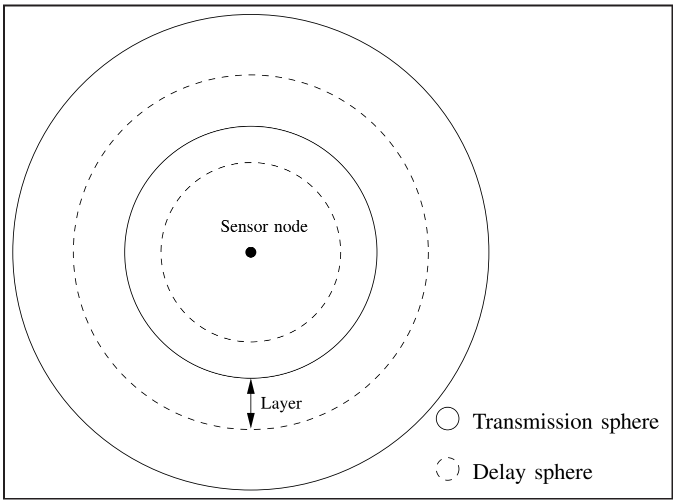

Fig. 2. A sensor node with four layers.

In [5] we found a complete set using the notion of *Maximal Overlapping Regions* (MOR). An *overlapping region* is a region where a nonempty subset of surface circles overlap. For an overlapping region $\mathcal{O}$, let $\mathcal{S}(\mathcal{O})$ denote the subset of surface circles that overlap at $\mathcal{O}$. An overlapping region $\mathcal{O}$ is *maximal* if there is no overlapping region $\mathcal{F}$ where $\mathcal{S}(\mathcal{O}) \subset \mathcal{S}(\mathcal{F})$. Fig. 3 shows six surface circles with their MORs being shaded.

We have shown that a set $\mathcal{P}$ that contains one point from every MOR is complete. For the delay constrained problem, we need to find all MORs of both transmission and delay spheres. We can use our algorithms presented in [5] with a minor modification to take into account the delay spheres. For the sake of completeness, those algorithms are shown here in Algorithms 1,2, and 3. Intersection points (i.e., points where the boundaries of two surface circles intersect) are classified into *entry points*, *exit points*, and *tangent points*; and Algorithm 2 uses this classification with the notion of *defect points* to identify MORs. By visiting intersection points incident to a surface circle ζ_i^k in a clockwise order, an entry point is one at which we enter another surface circle, an exit point is one at which we leave another surface circle, and a tangent point occurs if two surface circles intersect at exactly one point. Defect points are added to the arrangement of circles to check whether some circle lies entirely inside an overlapping region. A circle may add at most one defect point. To find the defect point of a circle ζ_i^k centered at a point (x_i, y_i), we find points that:

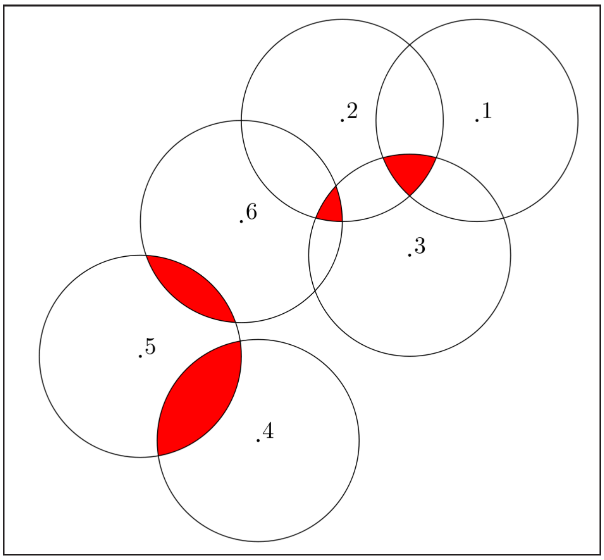

Fig. 3. Maximal overlapping regions.

Algorithm 1: Arrangement of surface circles.

Procedure FindArrangement( )
foreach *sensor node* n_i **do**
  for $j = 1..\lfloor \frac{MaxRange}{\Delta} \rfloor$ **do**
    Add a delay sphere centered at n_i with a radius of $j\Delta$;
  end
  Find all surface circles derived from all delay and transmission spheres of n_i;
end
Find all points where the boundaries of two surface circles intersect;
Find all defect points;
foreach *surface circle* ζ_i^k **do**
  Sort all intersection points and defect points incident to ζ_i^k in a clockwise order;
end

1) lie at the intersection of the line $y = y_i$ and another circle ζ_j^l.
2) are outside and to the right of ζ_i^k.

Amongst those points that satisfy these two conditions, we take the point p which is the closest to (x_i, y_i). Let us say p lies at the boundary of a circle ζ_j^l. Then, if (x_i, y_i) is inside ζ_j^l, p is added as a defect point on ζ_j^l. Correctness proof and more details about these algorithms are provided in [5].

C. ILP formulation

Once we obtain the set $\mathcal{P} = \{p_0, p_1, ..., p_{M-1}\}$, which contains a point from each MOR, the problem of finding the optimal locations of R data collectors and the routing paths from sensor nodes to data collectors can be formulated as an ILP. We define the following constants and variables.

Constants:

Algorithm 2: Testing whether an overlapping region is maximal or not.

Function Maximal(ζ_i^k: a surface circle, p: an intersection point incident to ζ_i^k)
Input: A surface circle ζ_i^k and a point p, which could be an intersection point or a defect point, incident to ζ_i^k.
Output: True if a MOR is found, and False otherwise.
if *Flag(p,i)* $= 1$ *OR p is an entry point with respect to* ζ_i^k *OR p is a defect point* **then**
  return False ;
end
if *p is a tangent point* **then**
  return True ;
end
Flag(p,i) := 1;
q:=p;
(j, l):=$other_i^k(p)$;
p:=$succ_j^l(p)$;
i:=j;
k:=l;
while $p \neq q$ **do**
  if *Flag(p,i)* $= 1$ *OR p is an entry point with respect to* ζ_i^k *OR p is a defect point OR p is a tangent point* **then**
    return False ;
  end
  Flag(p,i) := 1;
  (j, l):=$other_i^k(p)$;
  p:=$succ_j^l(p)$;
  i:=j;
  k:=l;
end
return True ;

Algorithm 3: Finding all MORs.

Procedure FindMORs( )
Output: A set $\mathcal{P}$ that contains one point from every MOR.
$\mathcal{P}$:=ϕ ;
FindArrangement( );
foreach *intersection point p which is incident to two surface circles* ζ_i^k *and* ζ_j^l **do**
  Flag(p,i):=0;
  Flag(p,j):=0;
end
foreach *surface circle* ζ_i^k **do**
  foreach *intersection point p incident to* ζ_i^k **do**
    if *Maximal(*ζ_i^k*,p)* **then**
      $\mathcal{P}$:=$\mathcal{P} \cup \{p\}$;
    end
  end
end

Maximize $\alpha E_{min} - \beta E_{total}$
s.t.,

$$\sum_{j \in N(i)} \sum_{W^*_{ij} \leq d < \Gamma} f^d_{ij} + \sum_{j \in M(i)} \sum_{D^*_{ij} \leq d \leq \Gamma} h^d_{ij} \leq C_i, 0 \leq i < N \quad (1)$$

$$\sum_{\substack{j \in N(i), \\ d \geq W^*_{ji}}} f^d_{ji} = \sum_{\substack{j \in N(i), \\ d < \Gamma - W^*_{ij}}} f^{d+W^*_{ij}}_{ij} + \sum_{\substack{j \in M(i), \\ d \leq \Gamma - D^*_{ij}}} h^{d+D^*_{ij}}_{ij}, 0 \leq i < N, 1 \leq d < \Gamma \quad (2)$$

$$G_i = \sum_{\substack{j \in N(i), \\ W^*_{ij} < \Gamma}} f^{W^*_{ij}}_{ij} + \sum_{\substack{j \in M(i), \\ D^*_{ij} \leq \Gamma}} h^{D^*_{ij}}_{ij}, 0 \leq i < N \quad (3)$$

$$\begin{aligned} E_i \quad &- \sum_{j \in N(i)} \sum_{W^*_{ij} \leq d < \Gamma} E_{Tr}(W_{ij})\, f^d_{ij} - \sum_{j \in M(i)} \sum_{D^*_{ij} \leq d \leq \Gamma} E_{Tr}(D_{ij})\, h^d_{ij} \\ &- E_{Rc} \sum_{j \in N(i)} \sum_{W^*_{ji} \leq d < \Gamma} f^d_{ji} \geq E_{min} \quad , 0 \leq i < N \end{aligned} \quad (4)$$

$$\begin{aligned} \sum_{0 \leq i < N} \quad (&\sum_{j \in N(i)} \sum_{W^*_{ji} \leq d < \Gamma} E_{Rc} f^d_{ji} + \sum_{j \in N(i)} \sum_{W^*_{ij} \leq d < \Gamma} E_{Tr}(W_{ij})\, f^d_{ij} \\ + &\sum_{j \in M(i)} \sum_{D^*_{ij} \leq d \leq \Gamma} E_{Tr}(D_{ij}) h^d_{ij}) = E_{total} \end{aligned} \quad (5)$$

$$E_{min} \geq 0 \quad (6)$$

$$\sum_{i \in B(j)} \sum_{D^*_{ij} \leq d \leq \Gamma} h^d_{ij} \leq l_j \sum_{0 \leq i < N} G_i, 0 \leq j < M \quad (7)$$

$$\sum_{0 \leq j < M} l_j = R \quad (8)$$

$$l_i \in \{0, 1\}, 0 \leq i < M$$

Fig. 4. ILP formulation.

G_i is the data generation rate of sensor node n_i.
$N(i)$ is a set of indices such that $j \in N(i)$ if n_j is within the transmission range of n_i (i.e., n_j is a neighbor of n_i).
$M(i)$ is a set of indices such that $j \in M(i)$ if the point p_j, which belongs to a MOR, is within the transmission range of n_i.
$B(i)$ is a set of indices such that $j \in B(i)$ if the point p_i, which belongs to a MOR, is within the transmission range of n_j.
N is the number of sensor nodes.
R is the number of data collectors.
M is the number of MORs.
E_i is the residual energy of sensor node n_i.
$E_{Tr}(d)$ is the energy consumed to send one data unit to a destination which is d units away from the source.
E_{Rc} is the energy consumed to receive one data unit.
W_{ij} is the Euclidean distance between sensor nodes n_i and n_j.
D_{ij} is the Euclidean distance between a sensor node n_i and a point p_j, which belongs to a MOR.
C_i is the traffic capacity of sensor node n_i (i.e., the maximum number of data units that can be relayed by n_i per round).
α is the weight assigned to the minimum residual energy.
β is the weight assigned to the total consumed energy.
Variables:
$l_i = 1$ if a data collector is placed at p_i, and $l_i = 0$ otherwise.
If $j \in N(i)$, f_{ij} is the flow from sensor node n_i to sensor node n_j (i.e., the number of data units to be sent from n_i to n_j per round). This flow is decomposed into several subflows based on the encountered delay as explained below.
If $j \in M(i)$, h_{ij} is the flow from sensor node n_i to the data collector at p_j (if no data collector is placed at p_j, h_{ij} will be set to 0). This flow is decomposed into several subflows based on the encountered delay as explained below.
E_{min} is the minimum residual energy over all sensor nodes at the end of the round.
E_{total} is the total consumed energy during the round.

Our policy of maximizing the lifetime is to maximize the minimum residual energy (i.e., E_{min}) at the end of each round. However, it is easy to see that more than one solution may have the same optimal value for E_{min} but possibly

different values for E_{total}. Amongst those solutions for which E_{min} is maximized, we want to pick the one with the minimum E_{total}. In order to do so, we have the following objective function:
$\alpha\ E_{min} - \beta\ E_{total}$,
which is composed of a linear combination of E_{min} and E_{total} with a much higher weight given to E_{min} (i.e., $\alpha \ggg \beta$).
In order to accommodate the delay in the ILP formulation, the flow from a node n_i to another node n_j, i.e., f_{ij}, is decomposed into several subflows according to the encountered delay. f_{ij}^d denotes the flow arriving to node n_j through node n_i, for which the discrete delay from the source of the flow to node n_j is d. In other words, by the time the flow f_{ij}^d arrives to node n_j, it will have encountered d delay steps. Similarly, h_{ij}^d denotes the flow arriving to a data collector placed at a point p_j through node n_i, for which the discrete delay from the source of the flow to the data collector is d.
When Γ is the upper bound on the discrete delay, the flow from a sensor node n_i to another node n_j is decomposed into $\Gamma - W_{ij}^*$ subflows: $\{f_{ij}^{W_{ij}^*}$, $f_{ij}^{W_{ij}^*+1}$, ... ,$f_{ij}^{\Gamma-1}\}$. And the flow from a sensor node n_i to a data collector placed at a point p_j is decomposed into $\Gamma - D_{ij}^* + 1$ subflows: $\{h_{ij}^{D_{ij}^*}$, $h_{ij}^{D_{ij}^*+1}$, ... ,$h_{ij}^{\Gamma}\}$.
The ILP is shown in Fig. 4. Equation (1) satisfies the traffic capacity constraints. Equation (2) guarantees the flow balance and makes the appropriate flow decomposition for different delay values; if some data arrived to a node n_i over the flow f_{ki}^d and is to be relayed to a node n_j, it should go on the flow $f_{ij}^{d+W_{ij}^*}$; and if it is to be relayed to a data collector at a point p_j, it should go on the flow $h_{ij}^{d+D_{ij}^*}$. Equation (3) guarantees that each sensor node pushes its own data to its neighbors on the appropriate subflows; data generated at node n_i and sent to node n_j will encounter W_{ij}^* delay steps, and data generated at node n_i and sent to a data collector at a point p_j will encounter D_{ij}^* delay steps. Equation (4) makes E_{min} the minimum residual energy over all sensor nodes (note that we maximize E_{min}). Equation (5) sets E_{total} to the total energy consumption. Equation (6) guarantees that the energy expenditure of any sensor node is not more than its current residual energy. Equation (7) guarantees that if no data collector is placed at p_j (i.e., l_j = 0), no flow is sent to p_j. Equation (8) satisfies the constraint that only R data collectors are available.

IV. Experimental results

We conducted some experiments to compare our Delay Constrained (DC) scheme with two other schemes. The first one is the FIXED scheme in which data collectors are static and are placed randomly on the surface of the water. In the FIXED scheme, we use a similar ILP to find near-optimal routing for a given placement. The second scheme we compare with is the Delay Tolerant (DT) scheme we proposed in [5]. The comparisons are made in terms of network lifetime and the maximum encountered delay.
In our simulations we assume three transmission levels: 5 km, 2.5 km, and 1 km. The WHOI modem [2] transmits over 5 km at 10W with a data rate of 220 bits/sec. Other acoustic modems have slightly different energy consumption settings [12]. For simulation purposes, we set our energy consumption parameters to be compatible with those of the WHOI modem and with the general energy consumption model described in [11]. So we set $E_{Tr}(5\text{ km}) = 20$ mJ/bit, $E_{Tr}(2.5\text{ km}) = 5$ mJ/bit, $E_{Tr}(1\text{ km}) = 2$ mJ/bit, and $E_{Rc} = 1$ mJ/bit. Every sensor node has an initial energy of 2x10^5 J, and generates 64 bytes/hour. The round length is one day. The capacity of sensor nodes is uniformly distributed between 1000 and 2000 bytes/hour. Our simulations are applied to networks of 100 sensor nodes randomly deployed in a 20 x 20 x 2 km^3 volume (the maximum depth is 2 km). In each network, we tested different scenarios of one, two, three, four, and five data collectors. Δ is set to 1000 m and Γ is set to 22. For each scenario, we generate 20 random samples and take the average. α and β are set to 10^6 and 1, respectively. To solve the ILP, we use *lp_solve* 5.5 [3] with a timeout of 20 minutes.
Our scheme achieves longer lifetime and shorter delay as compared with the FIXED scheme. And when Γ is small enough, it is expected that the DT scheme will have longer lifetime but longer delay as compared with the DC scheme, and that is what our experiments have verified. Fig. 5 shows the lifetime comparison between the three schemes. It is obvious that the DT provides a longer lifetime due to the relatively more freedom it has in choosing paths of any length. However, the DC scheme has a longer lifetime as compared with that in the FIXED scheme. In terms of delay, while the maximum delay encountered in the DC scheme is bounded by Γ (i.e., 22), the maximum delays in the DT and FIXED schemes are more than Γ as shown in Fig. 6. Apparently, the disparity between the maximum delay encountered in the DT scheme and that in the DC scheme dwindles as more data collectors become available. That is due to the fact that having more data collectors reduces the chances of having sensor nodes for which the nearest data collector is too faraway.

V. Conclusion

This paper presents a scheme for finding near-optimal routing and placement of mobile data collectors in UASNs under delay constraints. The novelty of the proposed scheme stems from:

1) Solving three-dimensional placement problems in which a data collector can be placed anywhere on the surface of the water, and
2) Solving the problem of prolonging the lifetime with a delay-based QoS guarantees.

The problem is formulated as an ILP with an objective function that takes into account both the current residual energy and future energy expenditure of each sensor node. Experimental results show that our scheme has the potential

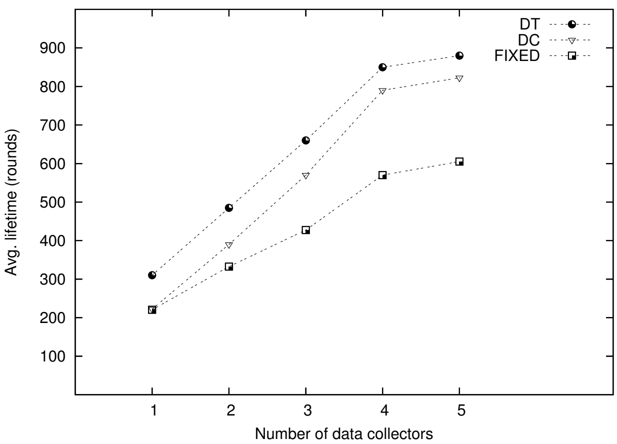

Fig. 5. Lifetime comparison.

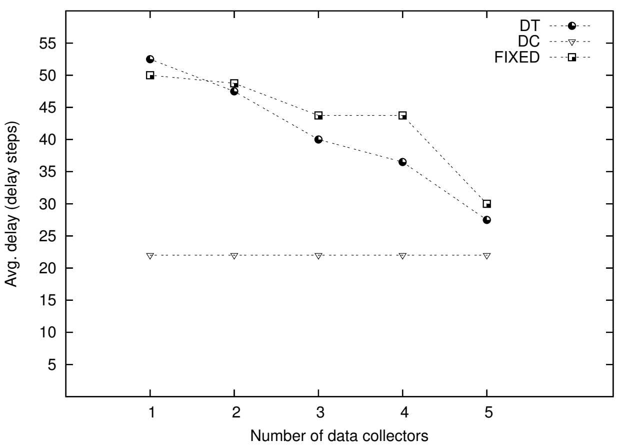

Fig. 6. Delay comparison.

to prolong the lifetime of a UASN significantly as compared with another random, static scheme. We also compare this delay constrained scheme with our delay tolerant scheme. This comparison reveals that when the delay bound is tight enough, the delay constrained scheme has a slightly shorter lifetime as compared to the delay tolerant scheme, yet it has the ability to meet any delay-based QoS guarantees.

We are currently extending our scheme to a more general UASN architecture where data collectors can go below the surface of the water to get closer to sensor nodes and to avoid having bottlenecks among sensor nodes near the surface. Such an architecture demands a schedule for data collectors that need to go up to the surface periodically in order to report data to the sink.

ACKNOWLEDGMENT

This research was supported by the Natural Sciences and Engineering Research Council of Canada and by the Ministry of Higher Education in Saudi Arabia.

REFERENCES

[1] http://web.uvic.ca/ neptune/.
[2] http://acomms.whoi.edu/micromodem/.
[3] http://lpsolve.sourceforge.net/5.5/.
[4] I. Akyildiz, D. Pompili, and T. Melodia. Underwater acoustic sensor networks: Research challenges. *Ad Hoc Networks*, 03(3):257–279, 2005.
[5] W. Alsalih, S. Akl, and H. Hassanein. Placement of multiple mobile data collectors in underwater acoustic sensor networks. In *Proc. IEEE International Conference on Communications (ICC)*, May 2008.
[6] A. Azad and A. Chockalingam. Mobile base stations placement and energy aware routing in wireless sensor networks. In *Proc. The IEEE Wireless Communications and Networking Conference (WCNC)*, April 2006.
[7] S. Gandham, M. Dawande, R. Prakash, and S. Venkatesan. Energy efficient schemes for wireless sensor networks with multiple mobile base stations. In *Proc. The IEEE Global Telecommunications Conference (GLOBECOM)*, December 2003.
[8] A. Harris and M. Zorzi. On the design of energy-efficient routing protocols in underwater networks. In *Proc. IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, June 2007.
[9] S. Jain, R. Shah, W. Brunette, G. Borriello, and S. Roy. Exploiting mobility for energy efficient data collection in wireless sensor networks. *Mobile Networks and Applications*, 11(3):327–339, 2006.
[10] V. Rodoplu and M. Park. An energy-efficient MAC protocol for underwater wireless acoustic networks. In *Proc. MTS/IEEE OCEANS*, September 2005.
[11] E. Sozer, M. Stojanovic, and J. Proakis. Underwater acoustic networks. *IEEE Journal of Oceanic Engineering*, 25(1):72–83, 2000.
[12] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke. Data collection, storage, and retrieval with an underwater sensor network. In *Proc. International Conference on Embedded Networked Sensor Systems (SenSys)*, November 2005.
[13] P. Xie and J. Cui. R-MAC: An energy-efficient MAC protocol for underwater sensor networks. In *Proc. International Conference on Wireless Algorithms, Systems and Applications (WASA)*, August 2007.