MIF: Optimizing Information Freshness in Intermittently Connected Sensor Networks

Howard Luu, Hung Ngo, Bin Tang, Mohsen Beheshti
Department of Computer Science
California State University Dominguez Hills, Carson, CA 90747, USA
Email: {hluu5,hngo10}@toromail.csudh.edu, {btang, mbeheshti}@csudh.edu

ABSTRACT

In this paper, we study information freshness in intermittently connected sensor networks (ICSNs). ICSNs are emerging sensing applications and systems that are deployed in challenging environments (e.g., underwater exploration). Due to the inaccessibility of the environments, the data-collecting base station is usually not available inside the ICSN. Consequently, the newly generated data in ICSNs must be stored inside the network before uploading opportunities (e.g., autonomous underwater vehicles (AUVs)) become available. How to accurately quantify and effectively achieve the freshness of information stored in ICSNs pose a new challenge.

We propose a new algorithmic framework to attain optimal and efficient information freshness in ICSNs. We refer to it as MIF: maximization of information freshness in ICSNs. The goal of MIF is to maximize the freshness of data packets stored in ICSNs while incurring a minimum amount of energy cost in this process. We first consider a special case wherein nodes are not energy-constrained and show that with proper transformation, MIF can be modeled as a minimum cost flow problem, which can then be solved optimally and efficiently. For the general case that nodes are energy-constrained, we formulate it as an integer linear programming (ILP) problem and solve it optimally. We also propose time-efficient greedy algorithms for each case. Simulation results show that our algorithms achieve information freshness for ICSNs under different network parameters while incurring minimum energy consumptions.

KEYWORDS

Information freshness, network flows, integer linear programming, intermittently connected sensor networks

1 INTRODUCTION

Background. Maintaining information or data freshness has been an important task for any computer and information systems [17, 24]. For example, real-time databases maintain fresh views that are derived from data shared from multiple distributed sites [23, 36], and web caching and replication systems must ensure the freshness of its stored data in order to provide a robust and efficient Web infrastructure [3, 34].

In this paper we focus on information freshness for many emerging sensor networks that are deployed in challenging environments. Such sensing systems include underwater or ocean sensor networks [4, 5, 7, 15, 26], volcano and seismic sensor networks [25], underground sensor networks [29], volcano eruption monitoring, and glacial melting monitoring [6, 31]. In such challenging environments, it is not feasible to install base stations with power outlets in or near the network to collect the data or receive its network updates. The data collection and system updates in such networks are usually accomplished by sending autonomous underwater vehicles (AUVs) [4] or mobile robots [28] to visit the sensor field periodically. As the connection between the sensor nodes and the AUVs and robots are intermittent, we refer to such sensor networks as *intermittently connected sensor networks* (ICSNs).

We assume that data generation in an ICSN takes place in a time-slotted manner, starting from time slot 0.[1] At some time slots, some events of interest occur inside the ICSN and the sensor nodes close to such events generate large volumes of sensory data related to the events. If the sensory data cannot be collected and uploaded timely by the AUVs and mobile robots (e.g., due to inclement weather in underwater exploration), it has to be stored inside the ICSN for some unpredictable amount of time. To characterize the freshness of the data packets stored inside the ICSN, we consider three typical stages of data generation and storage in the ICSN.

Stage 1. At time slot $t_1 \geq 0$, when a sensor node detects the events of interest, it generates sensory data packets and stores them at their local storages, as shown in Fig. 1(a). We refer to this stage as the *local storing* of a sensor node. When a data packet is stored locally at a sensor node, it is time-stamped with the current time slot number t_1 as its *age* (we will formally define the ages of packets and the ICSN in Section 3). As events of interest could occur at different locations inside the ICSN and at different time slots, different sensor nodes could store local data packets with different time stamps.

Stage 2. At time slot $t_2 > t_1$, as the large volumes of data packets are continuously generated at some sensor nodes, it depletes their storage spaces. We refer to this stage as *local storage overflow.* Such newly generated data packets that cannot be stored locally are referred to as *overflow data.* Fig. 1(b) shows that to preserve all the information sensed, the overflow data packets are time-stamped with the current time slot t_2 and then *offloaded* to neighboring sensor nodes with available storage to be stored.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
, ,
© 2021 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00

[1]We use time slots for ease of presentation. The time stamps can indeed be in any formats and intervals, which do not affect our problem formulation and solutions.

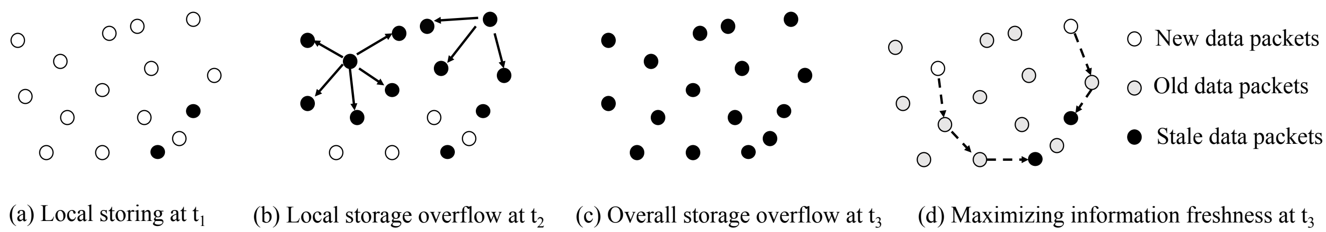


Figure 1: Illustrating the information freshness in the ICSN.

Stage 3. Finally, at time slot $t_3 > t_2$, with the continuous data offloading while the uploading opportunity is still unavailable, the offloaded data overflows the available storage in the entire network, shown in Fig. 1(c). Such *overall storage overflow*, wherein the entire storage spaces of the ICSN are full while new data packets are still being generated, cannot be alleviated by data offloading in Stage 2.

A Motivating Example. Let's consider an underwater exploration application wherein underwater camera sensors take pictures of the underwater environments while the AUVs are dispatched periodically to collect the pictures [4]. Consider 100 underwater camera sensors, each has 4 GB storage space. If 10 of them take one 640×480 JPEG color image per second with 3 bytes per pixel, it will just take around 10 hours to exhaust the storage spaces of all the 100 camera sensors and reach the overall storage overflow. In the events of inclement and stormy weather, which is not uncommon for underwater exploration, the AUVs cannot be dispatched and as such, the overall storage overflow could most possibly take place.

Contributions. We focus on how to maximize information freshness when overall storage overflow occurs in the ICSN.[2] As the newest information provides the latest development of the monitored sensor field, it thus needs to remove some existing old data packets (referred to as *stale data packets*) in order to empty spaces for the newly generated data packets, as shown in Fig. 1(d). The challenge is how to replace existing stale data packets with new data packets in an energy efficient manner in order to maximize the information freshness in ICSN.

To tackle this challenge we propose a new algorithmic framework called **MIF**: maximization of information freshness in ICSNs. The goal of MIF is to maximize the information freshness while incurring minimum energy consumption in this process in a dynamic ICSN environment, wherein events of interest could occur anywhere in the ICSN and at any time slot. We consider two scenarios viz. without and with energy constraints for sensor nodes. For the former, we show that MIF can be modeled as a minimum cost flow problem, which can then be solved optimally and efficiently. For the latter, we formulate an integer linear program (ILP) to solve MIF optimally. In either case, we encode all the necessary network information (i.e., data, storage, and energy) into a flow network transformed from the ICSN graph to achieve information freshness in the ICSN. We prove that our algorithms indeed achieve maximum freshness of information while using minimum energy consumption in the ICSN. We also propose greedy heuristics for the purpose of comparison. Simulation results show that all our algorithms achieve good information freshness in ICSNs under different network parameters.

[2] In this paper we are not concerned with Stage 1 and 2, as all the generated data can be collected by AUVs and mobile robots when they arrive at the field.

Paper Organization. The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents our system models, formulates MIF, and illustrates it with an example. In Section 4 we propose our solutions to MIF. Section 5 compares all the algorithms under different network dynamics and discuss the results. Section 6 concludes the paper and outline a few future research directions with some preliminary thoughts.

2 RELATED WORK

Data freshness has been an active research area for ICSNs such as underwater sensor networks [4, 11, 16, 35]. Basagni [4] et al. determined the collection path for the AUV so that the Value of Information (VoI) of the data delivered to the sink is maximized. Here VoI is defined as the difference between the time it is detected and the time it is delivered to the sink. Hollinger [11] studied how to plan the AUV path to maximize the information collected while minimizing travel time of the AUV. They proposed variants of the traveling salesperson problem and solution. Khan et al. [16] improved the end-to-end data freshness by focusing on the long propagation delay of acoustic data together with slow AUV speed, and designed AUV path of traversal to deliver data to the sink in timely manner. Wang [35] considered that surfacing and diving of AUVs cost huge energy consumption and found a trajectory schedule to minimize number of surfacing for AUVs.

However, all above work assume the AUVs are always available to collect the data from the ICSNs and focus on how to plan the paths for AUVs. Therefore they are not applicable to the scenarios where the AUV cannot be dispatched due to inclement and stormy weather, which is often the case for underwater explorations. Our work instead focus on such extreme scenario and study how to maintain information freshness within the network itself when AUVs are not available to collect the data.

In recent years, the Age of Information (AoI) was proposed as a new performance metric to measure the information freshness in systems wherein timeliness of status updates is critical to their functions and operations [9, 30]. We review a few related work.

A few works explored using unmanned aerial vehicles (UAVs) to timely deliver status updates and to improve the AoIs in IoT or sensor networks. For example, Hu et al. [14] studied a UAV-assisted wireless powered IoT network. In their scenario, the UAV starting from a data center visits sensor nodes to transfer energy and collect data. In a multi-hop network, the average AoI of the collected data thus depends on the UAV's trajectory. They designed dynamic programming and ant colony heuristic algorithms that achieved the optimal or near-optimal average AoI of the system. Abd-Elmagid et al. [2] used UAVs to minimize the average peak age-of-information for a source-destination pair. In particular. they jointly optimized the UAV's flight trajectory as well as energy and service time allocations for packet transmissions.

Liu et al. [19, 20] considered multi-hop network wherein a sender sends a flow at fixed rate to a receiver using multiple paths. They studied how to minimize the maximum delay, the average delay, as well as how to find the Nash equilibrium. They designed two polynomial-time solutions that delivers $1-\epsilon$ of the flow with maximum delay and average delay simultaneously within $\frac{1}{\epsilon}$ to the optimums. In their followup work [21], they considered a batch of data transmission and minimized peak/average AoI subject to throughput requirements. They showed the problem is NP-hard and developed an optimal algorithm with a pseudo-polynomial time complexity. In [22], they again considered the single sender and single receiver, and proposed to minimize network transmission under both a maximum delay constraint and a throughput requirement and designed an approximation and heuristic algorithms.

However, all above existing AoI research assumes a designated receiver or base station that is always available in the system to receive the system update data. Therefore their techniques cannot be applied to the ICSNs studied in this paper, wherein the base station is not available. However, we do not claim that our work studied AoI in ICSNs, as many unique features of AoI are not considered in our paper. Instead, our work focused on how to characterize and maximize the information freshness in a multi-hop ICSN by replacing the stale information with the fresh information. In our previous work, we proposed different data offloading techniques to achieve various objectives in the ICSN [12, 13, 18, 32, 33, 37]. However, none of them considered the information freshness in ICSNs, which is the topic of this paper.

3 PROBLEM FORMULATION OF MIF

Network Model. We model an ICSN as an undirected graph $G(V,E)$, where $V=\{1,2,...,|V|\}$ is the set of $|V|$ nodes and E is the set of $|E|$ edges. All the sensor nodes have the same transmission range; two nodes are connected if their distance is less than or equal to the transmission range. Sensor node $i \in V$ has storage capacity of m_i, indicating it can store m_i data packets. Let's assume all the packets have the same uniform size of k bits. We consider the overall storage overflow of ICSN sensor nodes' storage spaces are all full. That is, m_i old data packets are already stored at sensor node i. Let $q=\sum_{i=1}^{|V|} m_i$ denote the total number of old data packets in the network, and $O=\{o_1,o_2,...,o_q\}$ the set of the old data packets. Denote the time stamp on old data packet $j \in O$ as $t_j \geq 0$. Recall that time stamp of a packet is the time slot at which packet was generated and stored inside the ICSN. We assume that each time slot is long enough that the generation, offloading and storing of any data packet can all take place within one time slot. We denote the largest (i.e., latest) time stamp of all the old data packet as t_m; $t_m=\max\{t_j\}, j \in O$.

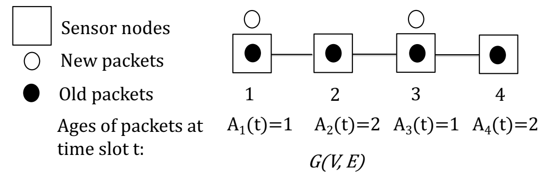


Figure 2: An illustrative example for MIF.

WLOG, let's assume that the current time slot is $t \geq t_m$, at which some new events of interest occur in the ICSN. WLOG let's assume that these events are detected by l sensor nodes $V_d=\{1,2,...,l\} \subset V$, which are referred to as *data nodes*. Data node $i \in V_d$ generates b_i data packets at the beginning of time slot t. Let $a=\sum_{i=1}^{l} b_i$, and let $N=\{n_1,n_2,...,n_a\}$ denote these newly generated data packets. We assume that $a \leq q$; otherwise, some new data packets must be discarded. Let the data node of data packet $n_j \in N$ be $d(j) \in V_d$. We first define the ages of data packets in $O \cup N$ and the age of the ICSN at time slot t as follows.

DEFINITION 1. (**Age of a Data Packet and age of the ICSN at the Beginning of Time Slot** $t \geq t_m$.) For any data packet, its age at time slot $t \geq t_m$, denoted as $A_j(t)$, is the time elapse between the time slot at which it is generated and the current time slot t; $A_j(t)=t-t_j$. Therefore for any old data packet $j \in O$, $A_j(t)>0$ whereas for any newly generated data packet $j \in N$, $A_j(t)=0$. The age of the ICSN $G(V,E)$ at *the beginning of* time slot t, denoted as $A(G,t)$, is the sum of the ages of all its data packets at this time slot. That is, $A(G,t)=\sum_{j \in O \cup N} A_j(t)$. □

Note that we distinguish the ages of the ICSN at the beginning and at the end of the time slot t. As maximizing information freshness at time slot t is achieved by replacing some existing old data packets with the a newly generated data packets, the ages of the ICSN at the beginning and the end of t are different. The goal of MIF is to replace the least fresh data packets among the q old data packets O to achieve maximum data freshness in a most energy-efficient manner. We refer to the least fresh data packets as *stale packets*, which are defined as below.

DEFINITION 2. (**Stale Packets and Stale Nodes at the Beginning of Time Slot** $t \geq t_m$.) Stale packets, denoted as S, are a old data packets in ICSN that have the largest ages at the beginning of time slot t. Let $S=\{s_1,s_2,...,s_a\} \subseteq O$. Denote the sensor node where old packet $s_j \in S$ is stored as $s(j)$. The set of sensor nodes storing at least one stale packet are called *stale nodes*, denoted as $V_s=\{S_1,S_2...,S_{|V_s|}\}=\bigcup\{s(j)|j \in S\}$. Given any stale node $i \in V_s$, denote the number of its stored stale packets as ξ_i; that is, $\xi_i=|\{j|s(j)=S_i, j \in S\}|$. □

Note that if a sensor node i is both a data node and a stale node (i.e. $i \in V_d \cup V_s$), it can replace its old data packets locally with its new data packets without incurring any energy cost. Thus if $b_i \geq \xi_i$, i is considered as a data node with $b_i-\xi_i$ new data packets; otherwise, i is a stale node with ξ_i-b_i stale packets. For ease of presentation, we still use b_i and ξ_i to denote the number of new data packets at i (if i is a data node) and the number of stale packets at i (if i is a stale node), respectively.

Table 1: Notation Summary

Notation	Description
$G(V,E)$	ICSN graph with $\lvert V\rvert$ sensor nodes and $\lvert E\rvert$ edges
m_i	Storage capacity of sensor node $i \in V$
q	Total number of old data packets in G, $q = \sum_{i \in V} m_i$
O	$O = \{o_1, o_2, ..., o_q\}$ is the set of q old data packets
V_d	$V_d = \{1, ..., l\}$ is the set of l data nodes
b_i	Number of new data packets generated at $i \in V_d$
a	$a = \sum_{i \in V_d} b_i$ is the total number of new packets
N	$N = \{n_1, n_2, ..., n_a\}$ is the set of a new packets
$d(j)$	The data node of a data packet $n_j \in N$
S	$S = \{s_1, s_2, ..., s_a\} \subseteq O$ is the set of a stale packets
V_s	$V_s = \{S_1, S_2 ..., S_{\lvert V_s\rvert}\}$ is the set of stale nodes
$s(j)$	The stale node of stale data packet $o_j \in O$
ξ_i	Number of stale data packets at stale node $S_i \in V_s$
$A_j(t)$	The age of data packet $j \in N \cup O$ at time slot t
$A(G,t)$	The age of the G at the beginning of time slot t
$A'(G,t)$	The age of the G at the end of time slot t
E_i	Initial energy level of sensor node i
$E_i^{'}$	Remaining energy of sensor node i after MIF
$E_u^t(v)$	Transmission energy of u to send one packet to v
E_v^r	Receiving energy of v to receive one data packet
r	Data offloading function for MIF
P_j	The *offloading path* of new data packet $n_j \in N$
$y_{i,j}$	Node i's energy cost of offloading data packet D_j
G'	Flow network for MIF without energy constraint
G''	Flow network for MIF with energy constraint

After replacing the stale packets in stale nodes, the age of $G(V,E)$ *at the end of* time slot t, denoted as $A'(G,t)$, becomes $A(G,t) - \sum_{j \in S} A_j(t)$. This is the minimum age G can achieve at the end of t.

Energy Model. Sensor node $i \in V$ has initial and finite energy power of E_i. We adopt the well-known first order radio model [10] for wireless energy consumption. When node u sends a k-bit data packet to its one hop neighbor node v over their distance $l_{u,v}$ meters, the *transmission energy* spent by u is $E_u^t(v) = \epsilon_{elec} * k + \epsilon_{amp} * k * l_{u,v}^2$, the *receiving energy* spent by v is $E_v^r = \epsilon_{elec} * k$. Here $\epsilon_{elec} = 100nJ/bit$ is the energy consumption per bit on the transmitter circuit and receiver circuit, and $\epsilon_{amp} = 100pJ/bit/m^2$ is the energy consumption per bit on the transmit amplifier. Let $E_{u,v} = E_u^t(v) + E_v^r$; we have $E_{v,u} = E_{u,v}$. Denote the minimum energy consumption sending a data packet from sensor node i to sensor node j as $c(i,j)$. We assume that the sensor nodes in ICSN have enough energy such that all data packets in N can be offloaded to the stale nodes, and leave the case wherein not all the packets can be offloaded as a future work. Table 1 shows all the notations.

Problem Formulation of MIF. We define *offloading function* as $r : N \longrightarrow V_s$, indicating that new data packet $n_j \in N$ is offloaded from its data node $d(j) \in V_d$ to a stale node $r(j) \in V_s$ to replace one of its stale packets. Let $P_j : d(j), ..., r(j)$, referred to as the *offloading path* of n_j, be the sequence of distinct sensor nodes along which n_j is offloaded from $d(j)$ to $r(j)$. Let $y_{i,j}$ be node i's energy cost of offloading n_j. Let $E_i^{'}$ be node i's remaining energy level at the end of time slot t after all the a new data packets are offloaded to some stale nodes. Then, $E_i' = E_i - \sum_{j=1}^{a} y_{i,j}, \forall\ i \in V$. The goal of MIF is to offload the a new packets to the stale nodes to replace their a stale packets while a) minimizing the total energy consumption $\sum_{i \in V} \sum_{j \in N} y_{i,j}$ during the data offloading and b) satisfying the energy constraint of sensor nodes $E_i' \geq 0$.

EXAMPLE 1. Fig. 2 shows a linear ICSN with four sensor nodes 1-4, each has one storage capacity with one old packet being stored. Each edge has 1 unit of energy cost. At the beginning of time slot t, the ages of the old packet from left to right is 1, 2, 1, 2, respectively. Therefore, $A(G,t) = 6$. During time slot t, nodes 1 and 3 each generates one new data packet and become data nodes. The optimal solution of MIF is to replace node 2's old data packet with node 1's new packet, and node 4's old packet with 3's new packet, costing minimum energy of 2 and resulting in $A'(G,t) = 2$ at the end of time slot t. Other data offloading solutions are not optimal. For example, although offloading node 3's new data packet to node 2 and node 1's to node 4 resulting in $A'(G,t) = 2$ as well, it incurs 4 amount of energy. □

4 ALGORITHMIC SOLUTIONS TO MIF

4.1 The Case Without Energy Constraint MIF-W

We first consider that sensor nodes do not have energy constraint, and refer to the problem as MIF-W. Recall that the minimum energy consumption of sending a data packet from sensor node i to sensor node j is $c(i,j)$. In MIF-W, $c(i,j)$ is thus the cost of the shortest path between i and j where the weight of an edge $(u,v) \in E$ is $E_{u,v} = E_u^t(v) + E_v^r$. We show that MIF-W can be modeled and solved as a minimum cost flow (MCF) problem on properly converted graphs. We thus briefly review MCF and its algorithms first.

Given a directed graph $G' = (V', E')$ with a source node s and a sink node t, each edge $(u,v) \in E'$ has a capacity $c_{u,v}$ as well as a cost $d_{u,v}$. Let $f(u,v)$ be the flow on an edge $(u,v) \in E'$. The goal of MCF is to find a flow function f to minimize the total cost of transmitting y amount of flow from s to t, i.e. $\Sigma_{(u,v) \in E'} (d_{u,v} \cdot f(u,v))$, subject to (a) capacity constraint: $f(u,v) \leq c_{u,v}, \forall (u,v) \in E'$, (b) flow conservation constraint: $\sum_{u \in V'} f(u,v) = \sum_{u \in V'} f(v,u)$, for each $v \in V' - \{s,t\}$, and (c) the net flow out of s and the net flow into t are both y. MCF can be solved efficiently by many combinatorial algorithms [27]. We adopt the algorithm by Goldberg [8], which is the authoritative implementation of the scaling push-relabel algorithm and has the highest performance codes available for such network optimization. It has the time complexity of $O(l^2 \cdot m \cdot \log(l \cdot n))$, where l, m, and n are the number of nodes, number of edges, and maximum edge capacity of $G'(V', E')$.

Transforming an ICSN to a Flow Network. We then convert the ICSN graph $G(V,E)$ in Fig. 2 to a flow network $G'(V', E')$ in Fig. 3 following below five steps.

First, $V' = \{s\} \cup \{t\} \cup V_d \cup V_s$, where s is the source node and t is the sink node in the flow network. $V_s = \{1, 2, ..., l\}$ is the set of l data nodes and $S = \{S_1, S_2, ..., S_{\lvert V_s\rvert}\}$ is the set of $\lvert V_s\rvert$ stale nodes.

Second, $E' = \{(s,i)\} \cup \{(i, S_j)\} \cup \{(S_j, t)\}$, where $i \in V_d$ and $S_j \in V_s$. Note it is a complete bipartite graph between V_d and V_s.

Third, for each edge (s,i), set its capacity as b_i, the number of new data packets at data node $i \in V_d$, and cost as 0. For each edge

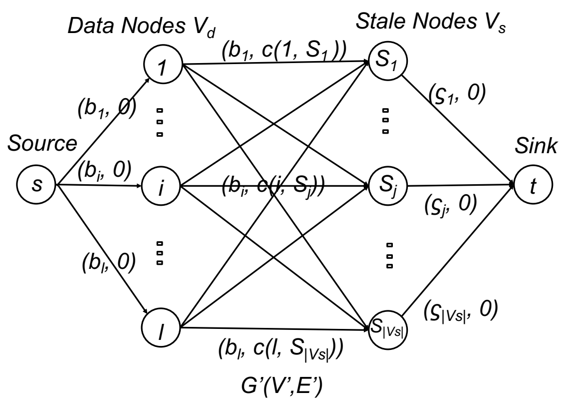


Figure 3: Flow network $G'(V', E')$ transformed from ICSN graph $G(V, E)$ for IoA^2 without energy constraint.

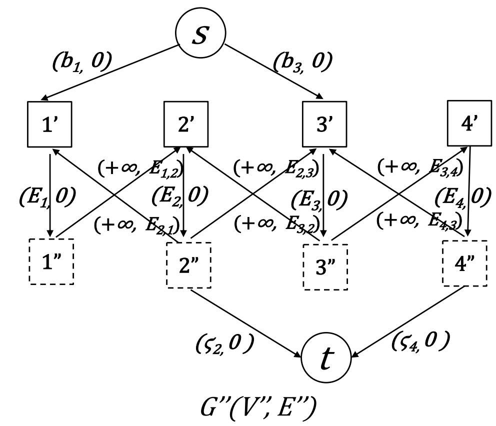


Figure 4: Flow network $G''(V'', E'')$ transformed from ICSN graph $G(V, E)$ for IoA^2 with energy constraint.

(S_j, t), set its capacity as ξ_j, number of stale packets at stale node $S_j \in V_s$, and cost as 0.

Fourth, for each edge (i, S_j), set its capacity as b_i and cost as $c(i, S_j)$, the minimum energy cost sending one packet from data node $i \in V_d$ to stale node $S_j \in V_s$.

Finally, set the supply at s and the demand at t as a, the number of new data packets generated at time slot t in the ICSN. With above transformation, $|V'| = |V| + 2$ and $|E'| = l + |V_s| + l \cdot |V_s|$.

THEOREM 1. *MIF-W in $G(V, E)$ is equivalent to MCF on $G'(V', E')$.*

Proof: We show that by applying MCF algorithm upon $G'(V', E')$, it guarantees that all the a overflow new data packets are offloaded to stale nodes V_s to achieve the minimum age $A'(G, t)$ in $G(V, E)$ while incurring minimum energy cost and respecting the storage constraints of sensor nodes.

First, each of the a new data packets are offloaded to exactly one stale node. As the amount of supply at s is $a = \sum_{i \in V_d} b_i$ while the capacity of each edge (s, i) is b_i, a valid flow of a amount from s to t must have b_i amount on edge (s, i). Next, as the capacity on each edge (i, S_j) is b_i, then b_i amount of flow must come out of any node $i \in V_d$ and go to the stale nodes in V_s. As each flow is unsplittable, each of the new data packet must arrive at one stale node.

Second, that ξ_j stale packets are replaced at stale node $j \in V_s$ is enforced by setting the capacity of edge (S_j, t) as ξ_j. Consequently, G achieves minimum age of $A(G, t) - \sum_{j \in S} A_j(t)$, as the a old data packets in S have the largest ages among all the old packets in G.

Third, MIF-W achieves the minimum energy cost in offloading the a new data packets in G. When applying MCF algorithm upon G', in which the cost on edge (i, S_j) viz. $c(i, j)$ is the energy cost of offloading one packet from data node i to stale node S_j, it thus gives the minimum total energy cost. ∎

4.2 A Greedy Algorithm for MIF-W

We also propose a more time-efficient greedy algorithm for MIF-W. In Algo. 1, each data node i offloads its b_i new packets to its closest stale nodes, until all the new data packets are offloaded. Finding the shortest stale node for any data node takes $O(|V|^2 \cdot \log(|V|))$. Therefore, the time complexity of Algo. 1 is $O(l \cdot V|^2 \cdot \log(|V|))$.

ALGORITHM 1. A Greedy Algorithm for MIF-W.
Input: An ICSN graph $G(V, E)$ with a set of new data packets N;
Output: $r : N \to V_s$;
1. **for** $(1 \le i \le l)$ // each data node
2. **while** (not all of its b_i new packets are offloaded)
3. Find the stale node in V_s closest to i that still has
4. stale packets not being replaced, say it is node j;
5. Offload $\min(b_i, \xi_j)$ new packets to j along the
the shortest path between i and j;
6. Update the energy levels of all the nodes involved;
7. **end while;**
8. **end for;**
9. **RETURN** $r : N \to V_s$.

4.3 The General Case of MIF

When sensor nodes have finite energy levels, it is possible that some of them deplete their energy power, making the data offloading process to maximize information freshness more challenging. We first convert the ICSN $G(V, E)$ to another flow network $G''(V'', E'')$ (shown in Fig. 4) by representing node i's energy level as capacities of some edges in G''. The conversion has the following six steps.

First, it replaces each undirected edge $(i, j) \in E$ with two directed edges (i, j) and (j, i) of capacities $+\infty$.

Second, it splits node $i \in V$ into two nodes *in-node* i' and *out-node* i'' and adds a directed edge (i', i'') with capacity of E_i, the initial energy level of node i.

Third, all incoming directed edges of node i are incident on i' and all outgoing directed edges of node i emanate from i''. Therefore the two directed edges (i, j) and (j, i) are now changed to (i'', j') and (j'', i'). we assign the costs of all directed edge (i'', j') as $E_{i,j} = E^t_i(j) + E^r_j$, the sum of node i's transmission energy to j and node j's receiving energy, the costs of directed edges (j'', i') as $E_{j,i} = E^t_j(i) + E^r_i$, the sum of node j's transmission energy to i and node i's receiving energy, and the costs of all other edges as zeros.

Fourth, it connects a super source node s to the in-node i' of the data node $i \in V_d$ with an edge of capacity b_i, the number of data packets at data node i.

Fifth, it connects the out-node i'' of the stale node $S_i \in S$ to a super sink node t with an edge of capacity ξ_i, the number of stale packets at S_i. Finally, it sets the supply at s and the demand at t as a, the number of overflow data packets in the ICSN. Therefore $|V''| = 2 \cdot |V| + 2$ and $|E''| = l + |V| + |V_s| + 2 \cdot |E|$.

ILP Formulation. Next, we apply below ILP (A) on $G''(V'', E'')$ and prove it minimizes the total cost in MIF. Here, $x_{i,j}$ and $c_{i,j}$ are the amount of flows and cost on edge $(i, j) \in E''$, respectively.

$$(A) \qquad \min \sum_{(i,j)\in E''} x_{i,j} \times c_{i,j} \tag{1}$$

s.t.

$$x_{s,i'} = b_i, \quad \forall i \in V_d \tag{2}$$

$$x_{i'',t} = \xi_i, \quad \forall i \in V_s \tag{3}$$

$$x_{s,i'} + \sum_{j:(i,j)\in E} x_{j'',i'} = \sum_{j:(i,j)\in E} x_{i'',j'}, \quad \forall i \in V_d \tag{4}$$

$$\sum_{j:(i,j)\in E} x_{j'',i'} = \sum_{j:(i,j)\in E} x_{i'',j'} + x_{i'',t}, \quad \forall i \in V_s \tag{5}$$

$$E_i^r \times \sum_{j:(i,j)\in E} x_{j'',i'} + \sum_{j:(i,j)\in E} (E_i^t(j) \times x_{i'',j'}) \le E_i, \quad \forall i \in V_d \tag{6}$$

$$E_i^r \times \sum_{j:(i,j)\in E} x_{j'',i'} + \sum_{j:(i,j)\in E} E_i^t(j) \times x_{i'',j'} \le E_i, \quad \forall i \in V_s \tag{7}$$

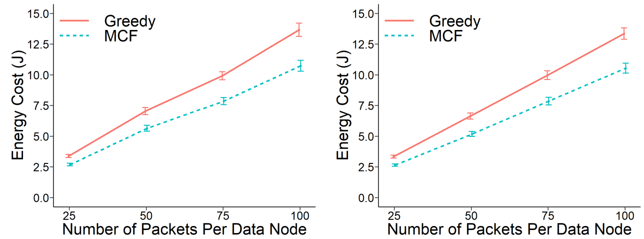
Eqn. 2 and 3 combined require all the $a = \sum_{i \in V_d} b_i$ new data packets in the ICSN will replace all the $a = \sum_{i \in V_s} \xi_i$ stale packets in the ICSN. Eqn. 4 shows the flow conservation for data nodes, wherein the total number of new packets transmitted by a data node equals the number of its own offloaded new packets plus the number of new data packets it relays for other data nodes. Eqn. 5 shows the flow conservation for stale nodes, wherein the total number of new packets it receives equals the number of packets it relays plus its stored stale packets. Inequalities 6 and 7 represent the energy constraint of data nodes and stale nodes respectively. Above graph conversion technique and ILP were used in our previous work [13] to solve a related data resilience problem in ICSN.

THEOREM 2. *MIF in $G(V, E)$ with energy constraint is equivalent to solving above ILP (A) on $G''(V'', E'')$.*

Proof: Besides the proof in Theorem 1, we need to show that the energy constraint is satisfied in ILP (A) in MIF. That is, node i does not spend more energy than its initial energy level E_i. This is accomplished by Inequalities 6 for data nodes and 7 for stale nodes. For data node i, the r.h.s of Inequality 6 has two terms: i's receiving energy and its transmission energy, the sum of which is less than E_i. Inequality 7 works similarly for a stale node i. ∎

4.4 A Greedy Algorithm for MIF

DEFINITION 3. (**Bottleneck Capacity.**) Given a data node i and a stale node j in ICSN graph $G(V, E)$, let $P(i, j)$ be the shortest path between i and j in terms of energy cost. For each node k on $P(i, j)$, including i and j, let the energy consumption of sending (for i), receiving (for j), and relaying (for other nodes on $P(i, j)$) one data packet be e_k. Denote the *bottleneck capacity* between i and j in G as $B(i, j, G)$. $B(i, j, G) = \max\{\lfloor \frac{E_k}{e_k} \rfloor\}, k \in P(i, j)$. □



(a) Time stamp = 100. (b) Time stamp = 1000.

Figure 5: Comparing algorithms for MIF-W by varying number of packets per data node b_i. Number of data nodes $l = 10$.

That is, $B(i, j, G)$ is the maximum number of packets that can be delivered from i to j following the energy constraints of all the nodes on $P(i, j)$. Algo. 2 below then works as follows. For each data node i, it first finds the closest stale node that still has stale packets not being replaced, say j, and computes $B(i, j, G)$. It then compares it with number of new packets at i (i.e., new_i) and number of stale packets at j (i.e., $stale_j$), and offloads the smallest (among these three numbers) numbers of packets from i to j. Note that $c(i, x)$ is the energy cost on the shortest path from node i to node x in the induced graph $G[V']$, where V' excludes the energy-depleted nodes from G. As different from Algo. 1, it now needs to deal with energy depletions of sensor nodes that could occur in the freshness maximizing process.

ALGORITHM 2. A Greedy Algorithm for MIF.

Input: An ICSN graph $G(V, E)$ with a set of new data packets N;
Output: $r : N \to V_s$;
new_i: number of new packets to be offloaded from data node i, initially equals b_i;
$stale_j$: number of stale packets to be replaced at stale node j, initially equals ξ_j;
$G[V']$: induced subgraph of $G(V, E)$ on V', initially $V' = V$

for $(1 \le i \le l)$ // each data node
  while $(new_i > 0)$
    $j = \text{argmin}_x c(i, x)$ and $stale_x > 0$;
    Let $q = \min\{new_i, stale_j, B(i, j, G[V'])\}$;
    Offload q new packets from i to replace q stale packets at j following its shortest path;
    Update the energy levels of all the nodes involved, let the set of nodes whose energy levels become zeros be V_0;
    $V' = V' - V_0$, $new_i = new_i - q$, $stale_j = stale_j - q$;
  end while;
end for;
RETURN $r : N \to V_s$.

5 PERFORMANCE EVALUATION

We compare the performance of different algorithms in ICSNs. For MIF-W, we compare MCF-based optimal (referred to as **MCF**) with greedy algorithm viz. Algo. 2 (referred to as **Greedy**). For MIF, we implement ILP-based optimal (referred to as **ILP**) using CPLEX [1]; we also implement the other greedy algorithm viz. Algo. 1 (referred to as **Greedy** too as the context is clear). We write our own simulator in Java on a Linux workstation (Ubuntu 20.04 LTS) with AMD processor (Ryzen 2600X) and 16GB of memory. In our simulations, 100 nodes that are uniformly distributed in a ICSN region of 1000m × 1000m. The transmission range of sensor nodes is 250m. In all the cases we set the storage capacity of sensor nodes m_j as 20. That is, initially each node has 20 old data packets with various time stamps. At some time slot t, some nodes generate new data packets and become data nodes. Each of the new and old data packet is of 512B. The timestamp of each packet is a random number in [1, 100] or [1, 1000]. For the case of MIF, each node has 0.8 Joule of initial battery capacity. Each data point in the plots is an average of 500 trials.

Algorithms for MIF-W. We first compare algorithms for MIF-W. Fig. 5(a) and (b) vary the number of new data packets per data node b_i while setting the time stamp (i.e., age) of each old data packet as a random number in [1, 100] and [1, 1000], respectively. It shows that in both cases, MCF outperforms Greedy due to its optimality in energy cost. Fig. 6 increases l from 10, 20, 30, to 40 while setting b_i as 50, which shows again that MCF outperforms Greedy. However, in all the above cases, their performance differences do not change when it switches from time stamp of 100 to 1000, due to the randomness of packets' ages in both cases. Nonetheless, both MCF and Greedy achieve maximum information freshness in the ICSN as the stale packets with maximum ages (i.e., time stamps) are able to be replaced by newly generated packets.

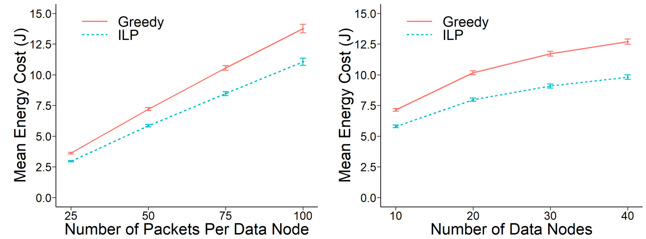
Algorithms for MIF. Next, we compare the algorithms for MIF, wherein each node has an initial energy capacity of 0.8 Joule and the ages of data packets are in the range of [1, 1000]. All the tested ICSN instances are feasible, which means all the stale packets can be replaced. Fig. 7(a) varies b_i from 25, 50, 75, to 100 while setting l as 10 while Fig. 7(b) varies l from 5 to 20 while fixing b_i as 50. Compared to MIF-W case (i.e., Fig. 5(b) and Fig. 6(b)), it is observed that the energy costs in MIF are slightly higher. As some nodes deplete their battery power, replacing some stale packets takes longer path in MIF than in MIF-W, incurring more energy cost.

We also study the fault tolerant capability of both ILP and Greedy. Fig. 8 shows the number of energy-depleted nodes resulted by apply them on an arbitrary ICSN instance. Greedy results in more energy depleted nodes than ILP does, as it is less energy-efficient than ILP. Nonetheless, both algorithms are fault-tolerant while still achieving maximum information freshness.

Comparing MIF-W and MIF. Finally, we quantitatively compare the results of MIF-W and MIF. In particular, we show the performance differences between MCF and Greedy in MIF-W, as well as the performance differences between ILP with Greedy in MIF. This is done by computing the percentage differences between the corresponding data values from Fig. 5, 6, and Fig. 7. Table 2 shows the results of varying b_i. The general observation is that the performance difference between MCF and Greedy in MIF-W is slightly larger than the performance difference between ILP and Greedy in MIF. This demonstrates that Greedy performs better in challenging scenario of limited energy. Table 3 gives the performance differences of algorithms in MIF and MIF-W by varying l. This time. we observe that performance differences of algorithms in MIF is evidently smaller than that in MIF-W, showing the Greedy is promising freshness maximization algorithm.

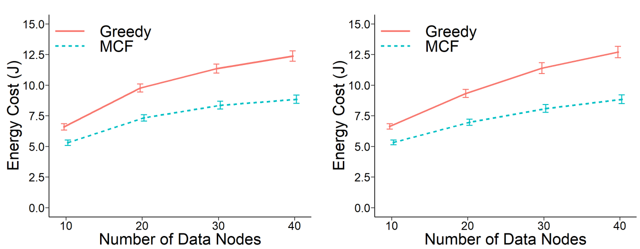
6 CONCLUSION AND FUTURE WORK

In this paper we presented a new algorithmic framework to study the information freshness in intermittently connected sensor networks (ICSNs) that are deployed in challenging environments. Applications of ICSNs include underwater or ocean sensor networks,



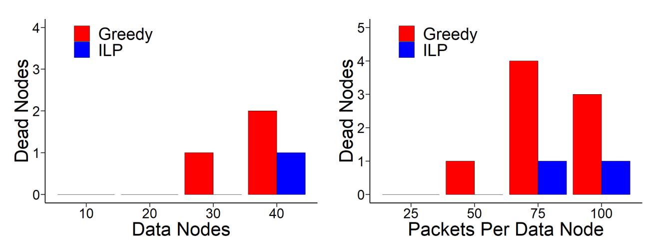
(a) Varying b_i. l = 10. (b) Varying l. b_i = 50.

Figure 7: Comparing algorithms for MIF. Time stamp = 1000.



(a) Time stamp = 100. (b) Time stamp = 1000.

Figure 6: Comparing algorithms for MIF-W by varying number of data nodes l. Each data node has 50 data packets.



(a) Varying b_i. l = 10. (b) Varying l. b_i = 50.

Figure 8: Number of Energy-Depleted Nodes.

Table 2: Performance Differences by Varying b_i (%).

b_i	25	50	75	100	Average
MIF-W (Fig. 5(a))	20.63	19.89	20.80	21.41	20.69
MIF-W (Fig. 5(b))	20.56	22.06	21.19	21.06	21.22
MIF (Fig. 7(a))	18.35	18.78	19.82	19.60	19.14

Table 3: Performance Differences by Varying l (%).

l	25	50	75	100	Average
MIF-W (Fig. 6(a))	19.64	24.95	26.32	28.48	24.85
MIF-W (Fig. 6(b))	19.72	25.23	28.97	30.41	26.08
MIF (Fig. 7(b))	18.90	21.63	22.45	22.75	21.43

volcano and seismic sensor networks, and underground sensor networks. Our goal is to maximize the information freshness of data packets stored in ICSNs while incurring minimum energy consumption in this process.

We proposed optimal minimum cost flow-based and ILP-based algorithms as well as efficient heuristic algorithms. Our approach addresses the dynamic scenario as well since the algorithms can be periodically executed to respond to the dynamic generations of new data packets at any time slot. As our work is the first one to study information freshness in ICSNs, we plan a few future research directions as follows. Currently, our work assumed all the new packets can be offloaded. This however may not be the case when sensor nodes have very limited battery power. Therefore we will study under which condition that not all the new packets can be offloaded; and if so, how to maximize the information freshness in this case. Second, we will consider some specific correlation model wherein nodes with close proximity store data packets with similar time stamps, and study how this could induce more efficient algorithms for information freshness maximization. Finally, we will study the hardness of the general MIF problem and propose efficient approximation algorithms with performance guarantees.

ACKNOWLEDGMENT

This work was supported by NSF under Grant No. 1911191, 1551221, and 1834620.

REFERENCES

[1] Ibm cplex optimizer. https://www.ibm.com/analytics/cplex-optimizer.

[2] M. A. Abd-Elmagid and H. S. Dhillon. Average peak age-of-information minimization in uav-assisted iot networks. *IEEE Transactions on Vehicular Technology*, 68(2):2003–2008, 2019.

[3] M. Baentsch, L. Baum, G. Molter, S. Rothkugel, and P. Sturm. Enhancing the web's infrastructure: from caching to replication. *IEEE Internet Computing*, 1(2):18–27, 1997.

[4] S. Basagni, L. Boloni, P. Gjanci, C. Petrioli, C. A. Phillips, and D. Turgut. Maximizing the value of sensed information in underwater wireless sensor networks via an autonomous underwater vehicle. In *Proc. of INFOCOM*, 2014.

[5] R. W. L. Coutinho, A. Boukerche, and S. Guercin. Performance evaluation of candidate set selection procedures in underwater sensor networks. In *Proc. of IEEE ICC 2019.*

[6] S. Edwards, T. Murray, T. O'Farrell, I. C. Rutt, P. Loskot, I. Martin, N. Selmes, R. Aspey, T. James, S. L. Bevan, and T. Baugé. A High-Resolution Sensor Network for Monitoring Glacier Dynamics. *IEEE SENSORS JOURNAL*, 14(11):3926–3931, 2013.

[7] R. Ghaffarivardavagh, S. S. Afzal, O. Rodriguez, and F. Adib. Ultra-wide band underwater backscatter via piezoelectric meta materials. In *Proc. of the ACM SIGCOMM 2020.*

[8] A. Goldberg. An efficient implementation of a scaling minimum-cost flow algorithm. *Journal of Algorithms*, 22:1–29, 1997.

[9] T. Q. S. Quek H. H. Yang, A. Arafa and H. V. Poor. Optimizing information freshness in wireless networks: A stochastic geometry approach. 20(6):2269–2280, 2021.

[10] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proc. of HICSS 2000.*

[11] Geoffrey A. Hollinger, Sunav Choudhary, Parastoo Qarabaqi, Christopher Murphy, Urbashi Mitra, Gaurav S. Sukhatme, Milica Stojanovic, Hanumant Singh, and Franz Hover. Underwater data collection using robotic sensor networks. *IEEE Journal on Selected Areas in Communications*, 30(5):899–911, 2012.

[12] X. Hou, Z. Sumpter, L. Burson, X., and B. Tang. Maximizing data preservation in intermittently connected sensor networks. In *Proc. of IEEE MASS 2012*, pages 448–452.

[13] S. Hsu, Y. Yu, and B. Tang. dre^2: Achieving data resilience in wireless sensor networks: A quadratic programming approach. In *Proc. of MASS 2020.*

[14] H. hu, K. Xiong, G. Qu, Q. Ni, P. Fan, and K. Letaief. Aoi-minimal trajectory planning and data collection in uav-assisted wireless powered iot networks. *IEEE Internet of Things Journal*, 8(2):1211–1223, 2021.

[15] J. Jang and F. Adib. Underwater backscatter networking. In *Proc. of the ACM SIGCOMM 2019.*

[16] Muhammad Toaha Raza Khan, Yalew Zelalem Jembre, Syed Hassan Ahmed, Junho Seo, and Dongkyun Kim. Data freshness based auv path planning for uwsn in the internet of underwater things. In *IEEE GLOBECOM.*

[17] A. Labrinidis and N. Roussopoulos. Balancing performance and data freshness in web database servers. In *Proc. of the VLDB 2003*, 2003.

[18] L. Liu and R. Wang. Message dissemination for throughput optimization in storage-limited opportunistic underwater sensor networks. In *Proc. of SECON 2016.*

[19] Q. Liu, L. Deng, H. Zeng, and M. Chen. A tale of two metrics in network delay optimization. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications.*

[20] Q. Liu, L. Deng, H. Zeng, and M. Chen. A tale of two metrics in network delay optimization. *IEEE/ACM Transactions on Networking*, 28(3):1241–1254, 2020.

[21] Q. Liu, H. Zeng, and M. Chen. Minimizing age-of-information with throughput requirements in multi-path network communication. In *Proc. of ACM Mobihoc 2019*, page 41–50, 2019.

[22] Q. Liu, H. Zeng, M. Chen, and L. Liu. Cost minimization in multi-path communication under throughput and maximum delay constraints. In *IEEE INFOCOM 2020.*

[23] E. Pacitti and E. Simon. Update propagation strategies to improve freshness in lazy master replicated databases. *VLDB Journal: Very Large Data Bases*, 8(3):305–318, 2000.

[24] V. Peralta. Data freshness and data accuracy: A state of the art. 2006. Technical Report.

[25] D. E. Phillips, M. Moazzami, G. Xing, and J. M. Lees. A sensor network for real-time volcano tomography: System design and deployment. In *Proc. of IEEE ICCCN 2017.*

[26] M. Rahmati and D. Pompili. Uwsvc: Scalable video coding transmission for in-network underwater imagery analysis. In *Proc. of IEEE MASS 2019.*

[27] K. A. Ravindra, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, 1993.

[28] R. Sugihara and R. K. Gupta. Path planning of data mules in sensor networks. *ACM Trans. Sen. Netw.*, 8(1):1:1–1:27, 2011.

[29] Z. Sun and I. F. Akyildiz. On capacity of magnetic induction-based wireless underground sensor networks. In *Proc. of INFOCOM*, 2012.

[30] R. Talak, S. Karaman, and E Modiano. Optimizing information freshness in wireless networks under general interference constraints. In *Proc. of ACM Mobihoc 2018*, 2018.

[31] Rui Tan, Guoliang Xin, Jinzhu Chen, Wen-Zhan Song, and Renjie Huang. Fusion-based volcanic earthquake detection and timing in wireless sensor networks. *ACM Transaction on Sensor Networks (ACM TOSN)*, 9, 2013.

[32] B. Tang. dao^2: Overcoming overall storage overflow in intermittently connected sensor networks. In *Proc. of IEEE INFOCOM 2018.*

[33] B. Tang, N. Jaggi, H. Wu, and R. Kurkal. Energy efficient data redistribution in sensor networks. *ACM Transactions on Sensor Networks*, 9(2):1–28, May 2013.

[34] A. Vakali and G. Pallis. Content delivery networks: status and trends. *IEEE Internet Computing*, 7(6):68–74, 2003.

[35] Ning Wang and Jie Wu. Trajectory scheduling for timely data report in underwater wireless sensor networks. In *2015 IEEE Global Communications Conference (GLOBECOM).*

[36] Y. Wei, S.H. Son, and J.A. Stankovic. Maintaining data freshness in distributed real-time databases. In *Proc. of 16th Euromicro Conference on Real-Time Systems, 2004.*

[37] X. Xue, X. Hou, B. Tang, and R. Bagai. Data preservation in intermittently connected sensor networks with data priorities. In *Proc. of IEEE SECON 2013*, pages 65–73.