Ad Hoc Networks 17 (2014) 71–81

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Collecting data in ad-hoc networks with reduced uncertainty

Liron Levin [a], Alon Efrat [b], Michael Segal [a,*]

[a] Dept. of Communication Systems Engineering, Ben-Gurion University, Israel
[b] Dept. of Computer Science, The University of Arizona, USA

ARTICLE INFO

Article history:
Received 28 August 2013
Received in revised form 13 November 2013
Accepted 14 January 2014
Available online 24 January 2014

Keywords:
Data gathering
Mule traversal
Optimization
Approximation algorithm

ABSTRACT

We consider the data gathering problem in wireless ad-hoc networks where a data mule traverses a set of sensors, each with vital information on its surrounding, and collects their data. The mule goal is to collect as much information as possible thereby reducing the information uncertainty but in the same time avoid visiting some of the nodes to minimize its travel distance. We study the problem when the mule travels over a tree or a tour and propose a 3-approximation algorithm that minimizes both the information uncertainty and travel distance. We also show the applicability of our approach for solving data collection problems in varying domains such as temperature monitoring, surveillance systems and sensor placement. Simulation results show that the proposed solution converges to the optimal for varying set of topologies, such as grids, stars, linear and random networks.

1. Introduction

Ad-hoc networks have become a popular research area both theoretically [32] and practically [15,24,28]. One of the most interesting and practical class of applications for sensors networks is agriculture and environmental monitoring [8,22], where wireless nodes are scattered over a geographical area and form a dynamic, infrastructure-less ad-hoc network. Typical monitoring scenarios consist of two stages. In the first step, the sensors periodically sense their surroundings, collect data and process it if needed. In the second step, the collected data is delivered to a base station using either *local* or *global* data collection process. In local data collection [21], the nodes use multi-hop communication over a static topology [5] or dynamic topology [17] to relay their message to the base station. This communication schema is most suitable when the distribution of nodes is dense, since the close proximity between nodes reduces the cost of transmitting and receiving messages [23]. In global data collection [1,6,21,25,26], the nodes are visited by a traveling *mule*, which uses short-range wireless communication to collect data from nearby nodes. The mule aggregates the information received from the nodes and delivers it to the base station. The data mule approach is most suitable when the network topology is sparse, the distance to the nearest node or base station is too large, or when the communication infrastructure between nodes is unstable. In addition, this approach reduces the responsibility for message routing from the nodes thereby minimizing their processing power. Most papers that use the data collector mule focus on energy efficiency [1,16] or travel time optimization [30] without considering the quality of the collected data. Since data in each node is closely linked to the monitored area, deciding which data is significant and which is redundant can greatly contribute to the performance of the data collecting algorithm. For example, consider an environmental monitoring system with large distribution of sensors in a specific geo-location. The added knowledge from visiting one node in the area might contribute significantly to the overall understanding of the environment. However, the knowledge obtained by visiting additional nodes in nearby locations may not be worth the time and transmission cost of the visit. Thus, the value from visiting a node is not a constant and depends on the nodes that were visited before it. In this paper we study a dual-objective optimization problem, where the goal is to

* Corresponding author. Tel.: +972 8 6477234; fax: +972 8 6472883.
E-mail addresses: levinlir@gmail.com (L. Levin), alon@cs.arizona.edu (A. Efrat), segal@cse.bgu.ac.il (M. Segal).

1570-8705/$ - see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.adhoc.2014.01.005

minimize the mule's traveling distance while minimizing the amount of information uncertainty caused by not visited a subset of nodes by the mule.

1.1. Our contribution

We present a framework for solving the mule problem when data each node sense is correlated with its neighbors. Our novel technique is used to find a tour or a tree under varying objective functions and has a 3-approximation ratio guarantee.

1.2. Outline of the paper

The rest of the paper is organized as follows: In Section 2, we present the system model, provide motivation and present the Mule Tree Problem (*MTP*) and Mule Cycle Problem (*MCP*). In Section 3 we discuss about problem background and previous work. We show a dual-primal formulation of the problem in Section 4, develop our 3-approximation algorithm for MTP and MCP in Section 5 and also show how to apply the algorithms for multiple optimization functions. We numerically evaluate the performance of our solutions against several common algorithms in Section 6 and conclude in Section 7.

2. Network model

Let $G = (V, E, r)$ be a complete graph embedded in the Euclidean plane, where V is the set of wireless nodes $(|V| = n)$ and E is the set of undirected edges between nodes. Here r is the base station and the other nodes represent sensors. For each $v \in V$, let $i(v) \in \mathbb{R}_+$ be the amount of data sensed by v and for each $e \in E$, let c_e be the mule's cost of traversing an edge e. Consider a mobile entity with wireless capabilities called mule that visits a subset of nodes $V_m \subseteq V$ and collect the information they sense. The mule tour or cycle starts at r and traverse along a subset E_m of the edge of E. The mule can decide to skip over some nodes $\overline{V_m} = V \setminus V_m$ and to absorb a penalty $\varphi(\overline{V_m} \in \mathbb{R}_+$. The value of the penalty reflects the data the mule would not collect. It is worth noting that the mule does not always have to visit a node to know its stored value; in many cases, this value could be inferred from knowing values in different part of the graph. For example, in aggregation binary tree, where nodes store the sum of values of their descendants, knowing the sum at a node v and one of its children would reveal the value of the other child (by merely performing subtraction).

A natural question arises whether the mule should visit all nodes, or skip over some and absorb the penalty. We ask how to find a tour that is not too long, but contains enough monitored data. Formally, the problem is defined as follows:

2.1. The Mule Tree/Cycle Problem (MTP/MCP)

Input: Graph $G = (V, E)$, cost c_e per edge, information sensed by each node $i(v)$ and a root r
Output: A tree $T_{mule} = (V_m, E_m)$ or a cycle $C_{mule} = (V_m, E_m)$
Objective: $\min(\varphi(\overline{V_m}) + \sum_{e \in E_m} c_e)$

In our problem φ represents the penalty function of not visiting the nodes of $\overline{V_m}$. We present a framework for approximating the mule problem for a variety of penalty functions. The approximation ratio holds as long as the following condition is fulfilled:

Condition 1. *For any two disjoint node sets, S_i and S_j, $\varphi(S_i) \leqslant \varphi(S_i \cup S_j)$.*

Specifically, we show that different penalty functions cover a variety of practical scenarios and present two concrete examples for the mule problem. First, we show an example related to environmental monitoring.

2.2. Temperature monitoring

For simplicity, we discuss the case where each sensor measures temperature (though, of course, this is not a restriction). Let $i(v)$ be the average temperature in the area covered by v. The objective is for the base station r to "know" the temperature measured by *each* sensor, but this goal might require too many relays (forwarding) in a multi-hoop fashion, which might be beyond accessible network resources such as battery life and channels capacities. Hence we opt to use an aggregation tree, and once needed (as formally articulated below), send the mule to retrieve to r more measurements from a (carefully-selected) subset of the sensors, so the uncertainty will decrease below a desired threshold.

Let $T = (V, E', r)$ be a communication tree rooted at r. Along its edges, sensors sends (summarizations of) measured temperature samples between adjacent nodes toward r. We assume channel capacity and sensors batteries' life are limited, hence transmitting all measurement using large messages in T is not prohibited. Instead, we assume that only summarizations are sent. Let $\mathcal{D}(v)$ be the set of the descendants of v in T. Due to the capacity limitation, node v can only send $i(v)$ and its descendant data $\sum_{u \in \mathcal{D}(v)} i(u)$ to its parent in T. We define this process as data aggregation. After data aggregation is completed by all nodes, each node v maintains the aggregated information, $\mathcal{I}(v) = \sum_{u \in \mathcal{D}(v)} i(u) + i(u)$, which represents what v knows about the temperature in its surrounding. When the mule visits a node v, it can learn the local information sensed $i(v)$ and the aggregated information $\mathcal{I}(v)$. From this information, the mule can infer on the data discrepancy between v's data to its descendants that was visited, $\mathcal{U}(v) = \mathcal{I}(v) - \sum_{u \in \mathcal{D}(v) \cap V_m} \mathcal{I}(u)$. Since $\mathcal{U}(v)$ decreases as the amount of data we collect increases, we define this measure as information uncertainty. $\mathcal{U}(v)$ is illustrated in Fig. 1. In Fig. 1(a), we have the input tree T, visited nodes $V_m = \{r, v_1, v_2, v_3\}$, and $\mathcal{D}(r) = \{v_1, v_2, v_3\}$. The amount of information per node v, $\mathcal{I}(v)$, is depicted in Fig. 1(b). Intuitively, by visiting only 3 nodes, we learn the temperature of almost the entire area. As the number of visited nodes becomes higher, the amount of uncertainty decreases. Specifically, the following penalty functions can be used to minimize the uncertainty:

$$\varphi_1(\overline{V_m}) = \frac{\sum_{v \in V_m} \mathcal{U} v^2}{|V|},$$

$$\varphi_2(\overline{V_m}) = \max_{v \in V_m} \mathcal{U} v.$$