Computer Networks 102 (2016) 96–108

Contents lists available at ScienceDirect

Computer Networks

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

Decentralized RFID coverage algorithms using writeable tags

Ahmed Jedda*, Mazen G. Khair, Hussein T. Mouftah

University of Ottawa, Ottawa, Canada

ARTICLE INFO

Article history:
Received 15 March 2015
Revised 3 March 2016
Accepted 17 March 2016
Available online 19 March 2016

Keywords:
RFID
RFID coverage
Topology construction
Distributed algorithms

ABSTRACT

A Radio Frequency IDentification (RFID) reader network is as a collaboration of RFID readers that aim to cover (i.e., identify, monitor, and track) every RFID tag in a given area. The RFID coverage (RFC) problem is defined as follows. Given a reader network, assign to each tag t a specific reader v in its proximity such that v is responsible for covering t (called its owner), while minimizing the number of owner readers. The problem has applications in energy conservation and in eliminating readers and data redundancy from the reader networks. We introduce a number of decentralized algorithms for the RFID coverage problem: 1) algorithms `RANDOM`, `RANDOM`$^+$, and `MAX-MIN` which are randomized algorithms that run in $O(1)$ write/read rounds, 2) algorithm `GDE` which is an efficient decentralized implementation of the greedy set cover algorithm, and 3) an improvement of `GDE` which is called . Our algorithms assume that the RFID tags are writeable, where a writeable tag is a passive RFID tag with writeable memory. We show using simulation experiments that our algorithms outperform major RFID coverage algorithms in various scenarios with respect to a number of performance metrics.

1. Introduction

A Radio Frequency IDentification (RFID) system generally consists of an RFID *reader* and an RFID *tag*. A reader sends a radio signal to the tag. Upon the reception of the reader's signal, the tag replies to the reader with a signal that contains the tag identifier and other parameters about the tag if possible. This working principle allows RFID systems to be used for identifying, tracking, and monitoring physical objects by simply attaching tags to them. Advances in hardware manufacturing led to significant improvements in the cost, size and performance of RFID systems. As a result, the use of RFID systems became an economically feasible option for many applications. For instance, RFID is notably used in the logistics, defence, aerospace, health and pharmaceutical sectors.

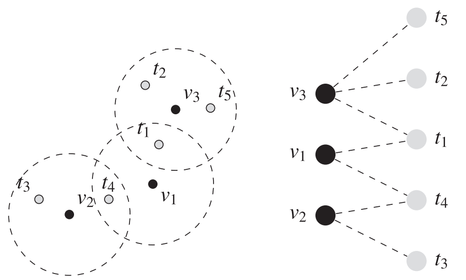
A main factor contributing to the recent widespread use of RFID is the low cost and high performance of RFID *passive tags*. A passive tag consists of an embedded circuit, a memory, and a transceiver, but no battery as it is empowered by the energy of the signals received by readers in its proximity. Its size can be in the orders of millimetres [1] allowing it to be attached to various objects. Some types of passive tags, called *writeable tags*, contain writeable memory [2]. Readers in proximity may write in the memory of writeable tags by sending radio signals. We focus in this paper on RFID systems that consist of passive writeable tags. There are other types of RFID tags that contain batteries. Some of these tags are allowed to initiate communication with the readers, and hence called *active tags*, while some others do not have this feature, and hence called *semi-passive tags*. These types of tags are of higher cost compared to passive tags, and thus have limited applicability.

The large scale of RFID systems is foreseen due to the low cost and small size of RFID tags and due to the large number of RFID applications. The main drivers of such networks are: 1) the Internet of Things (IoT), where every identifiable physical object (or, a *thing*) is expected to be connected to the Internet by attaching RFID tags or other uniquely identifiable tags, and 2) large supply chains such as those of the US Department of Defence, WalMart, Toyota, and others. The main problem in large scale RFID systems is the coverage of tags (i.e. identifying, monitoring, and tracking). The basic approach to overcome this problem is the use of collaborations of readers, called *reader network*, in order to cover all tags in a given area. Each reader in a reader network is responsible for covering a subset of the tags and report its readings to a special server that collects and processes the data gathered by all readers. An example of a reader network that consists of three readers and five tags is illustrated in Fig. 1(a). The coverage relationships between readers and tags are usually modeled as a bipartite graph as shown in Fig. 1(b).

We study the problem of optimizing the energy consumption of a reader network by eliminating unnecessary redundancy at the

* Corresponding author. Tel.: +15147062637.
E-mail addresses: ahmed.jedda@uottawa.ca (A. Jedda), mkhair@site.uottawa.ca (M.G. Khair), mouftah@uottawa.ca (H.T. Mouftah).

http://dx.doi.org/10.1016/j.comnet.2016.03.003
1389-1286/



(a) A sample reader network consisting of three readers (black dots) and five tags (gray dots). The dashed circles represent the covering range of readers.

(b) A bipartite graph model for the reader network on the left. A reader is linked to a tag if it covers it.

Fig. 1. Sample reader network illustrating readers redundancy.

readers level. The problem we consider is called the *RFID coverage* (RFC) problem. There are two objectives of the RFID coverage problem. The first is to assign each tag to a reader in its proximity called its *owner*. The owner of a tag is the only reader in the network that is responsible for reporting readings about the tag. The second objective of the RFID coverage problem is to minimize the set of owners in a given reader network (also called *non-redundant readers*).

A solution to the RFID coverage problem eliminates two types of redundancies; 1) data redundancy and 2) readers redundancy. *Data redundancy* occurs in situations where two readers or more report the same readings about the same tag. This type of redundancy a) causes problems in processing and mining the data generated by a reader network [3], and b) causes an increase in network traffic. Eliminating data redundancy can be done by assigning to each tag an owner reader, since only the owner of a tag is allowed to report readings about it. Note that a tag does not need to know which reader owns it, however, a reader must be aware of the tags it owns. This subproblem of the RFID coverage problem is called the *tag reporting* problem. *Readers redundancy* occurs if a tag or more in the reader network is covered by more than one reader. Minimizing the number of readers in a given reader network, while preserving the network coverage, improves the network energy consumption. This subproblem is called the *redundant readers elimination* problem. As an example, consider the reader networks of Fig. 1(a). We can assign v_2 as the owner of t_4 and t_3 and v_3 as the owner of t_1, t_2 and t_5. Reader v_1 therefore can be switched off. The negative impact of data and readers redundancies on reader networks become clearer as the reader network increases in scale.

The RFID coverage problem is similar to some variants of the sensor coverage problem [4,5]. It was introduced in [6] under the names of the *tag reporting* problem and the *redundant readers elimination* problem where it is assumed that the only means of communication to solve the problem is *reader-tag communications*. Herein, the readers cannot directly exchange messages, but they are allowed to write and read the memory contents of the tags in proximity using what is called *write/read rounds*. This model was later used in [7–11], and others. Another version of the RFID coverage problem, introduced in [12], does not allow the use of writeable tags, but allows direct message exchange between the readers using wireless communications. We focus on the first type of RFID coverage; the *reader-tag RFID coverage* problem.

Write/read rounds. A basic component in reader-tag RFID coverage algorithms[1] are *write/read rounds*. A randomized implementation of write/read rounds was introduced in [6]. Abstractly, a write/read round consists of two phases; write phase and read phase. In the write phase, every reader v writes a set of bits, called the *weight* of v and denoted by $\mathcal{W}(v)$, in the memory of all (or some) neighbor tags (i.e., tags that are covered by v). The readers wait for a specific period of time to allow every reader to do the same. In the read phase, the readers read the content of the memory of neighbor tags. At that time, the memory of a tag t contains the weights of all the neighbor readers of t that wrote in it during the write phase. More details on write/read rounds are given in Section 2.2.

Contributions. An RFID coverage algorithm is evaluated by the number of non-redundant readers it generates and the number of write/read rounds it executes. Many existing algorithms aim to achieve this objective by using a single write/read round but with different, sometimes sophisticated, definitions of the reader weights. A tag t is owned by the neighbour reader v that has the maximum weight $\mathcal{W}(v)$. In our *first set of contributions*, we set $\mathcal{W}(v)$ for every reader v to be a random number combined with the unique identifier of v. This introduces a simple single write/read round algorithm called `RANDOM`, which should be considered as a benchmark to similar algorithms due to its simplicity. Nevertheless, the simulation experiments in Section 7 show that `RANDOM` outperforms similar algorithms in practical scenarios. We also introduce algorithm `RANDOM`$^+$ and `MAX-MIN`, which both further improve the performance of `RANDOM` using additional write/read rounds, where each round is ran with a new randomly generated weight. These algorithms are shown to generate a low number of non-redundant readers with the cheap cost of one additional round (or few more).

The *second set of contributions* consists of two algorithms. The first is called the Greedy Decentralized Elimination (`GDE`) algorithm. It is the first decentralized algorithm that gives the same result of the centralized greedy set-cover algorithm. This algorithm generates the least number of non-redundant readers compared to existing RFID coverage algorithms. However, `GDE` runs in at most $|\mathcal{R}|$ iterations, where $\mathcal{R}$ is the set of readers. Each iteration consists of two write/read rounds. To improve `GDE` write/read complexity, we introduce `LIMITED-GDE` which limits the number of write/read rounds to $O(1)$ while keeping the number of non-redundant readers within an acceptable level that is still better than many other major algorithms.

The RFID coverage problem may appear with additional constraints, such as multihop communication connectivity between readers, k-coverage for improved fault-tolerance [13], handling faulty communication links, or achieving load balancing between readers. None of these constraints are considered in this paper because:

1. There is still room for improvements in the unconstrained version of the RFC problem as will be shown later, and
2. Studying the problem without constraints provides a better understanding of it, which helps later in a better understanding of its constrained versions.

Paper organization. Section 2 gives a survey of related work. Section 3 formalizes the problem and the mathematical model used. Algorithms `RANDOM`, `RANDOM`$^+$, `MAX-MIN` are described in Section 4. `GDE`, and `LIMITED-GDE` are described in sections 5 and 6 respectively. Each algorithm is given with a theoretical proof of correctness and complexity analysis. In Section 7 we use simulation experiments to study the empirical performance of our algorithms. Section 8 concludes the paper.

[1] We use the term RFID coverage in this paper to denote reader-tag RFID coverage.