Journal of Network and Computer Applications 65 (2016) 84–102

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

Heuristic solutions to the target identifiability problem in directional sensor networks

Wilson M. Tan [a,b,*], Stephen A. Jarvis [a]

[a] Performance Computing and Visualisation Group, Department of Computer Science, University of Warwick, UK
[b] Networks and Distributed Systems Group, Department of Computer Science, University of the Philippines, Quezon City, Metro Manila, Philippines

ARTICLE INFO

Article history:
Received 21 December 2014
Received in revised form
31 January 2016
Accepted 15 February 2016
Available online 24 February 2016

Keywords:
Sensor networks
Directional sensing antenna
Heuristic algorithm
Coverage quality
Distributed algorithms

ABSTRACT

Existing algorithms for orienting sensors in directional sensor networks have primarily concerned themselves with the problem of maximizing the number of covered targets, assuming that target identification is a non-issue. Such an assumption however, does not hold true in all situations. In this paper, heuristic algorithms for choosing active sensors and orienting them with the goal of balancing coverage and identifiability are presented. The performance of the algorithms are verified via extensive simulations, and shown to confer increased target identifiability compared to algorithms originally designed to simply maximize the number of targets covered.

1. Introduction and motivation

Directional sensors are sensors whose sensing capabilities are limited within an angle range (Ma and Liu, 2005, 2007). In comparison, an omnidirectional sensor's sensing range covers everything around it. In geometric terms, the covered area of an omnidirectional sensor is a circle centered on the sensor, while that of a directional sensor is a sector. In WSNs, nodes that are *directional* can imply that the node has directional capability in sensing and/or communication (Ma and Liu, 2005). In this paper, we solely focus on the sensing capability, and thus will interchangeably use the terms ‘nodes’ and ‘sensors’.

Examples of sensors that are inherently directional in nature include video sensors (Rahimi et al., 2005), ultrasonic sensors (Djugash et al., 2006), and infrared sensors (Szewczyk et al., 2004). Acoustic sensors (or microphones) can also potentially be directional (Panasonic, 2013), although most of the existing work in WSN literature so far have utilized omnidirectional microphones (Santini et al., 2008; Hakala et al., 2010).

Because the sensed region of a directional node is constrained within an angle range, of primary concern is the *total sensing coverage* provided by such a network of directional sensors. A problem that frequently needs to be solved is ‘*Given a number of directional sensors distributed in space, how should each sensor's sensing region be oriented such that the total sensing coverage of the network is maximized*?’.

Coverage-maximizing algorithms proposed in the literature fall into two categories: those that are *target-centric* and those that are *area-centric*.

In target-centric algorithms, it is assumed that there are a finite number of *static* targets distributed in the area, and it is desired that as many as possible of such targets be covered. Sometimes, for issues of energy efficiency, it is also desired that the covering be also done with the least number of sensors possible – such a problem is formalized in Ai and Abouzeid (2006) as the Maximum Coverage Minimum Sensors (*MCMS*) Problem.

In area-centric algorithms, there are no specific objects or targets of interest; the entire area is of sensing interest, and it is desired that as much of it is covered. In such algorithms, the problem is equivalent to minimizing the *overlap* between the sensed regions of sensors. An example of such an algorithm was presented in Tao et al. (2013).

In this work, we solely focus on target-centric algorithms.

Most coverage algorithms focus on maximizing the number of targets covered. This is reasonable when the targets are continually being monitored, and easy target identification is built-in to the system. This is true to a certain extent for visual sensor networks: for instance, a Closed Circuit Television (CCTV) camera (being remotely watched by a person) monitoring a street

* Corresponding author at: Networks and Distributed Systems Group, Department of Computer Science, University of the Philippines, Quezon City, Metro Manila, Philippines.
E-mail address: wmtan@dcs.upd.edu.ph (W.M. Tan).

http://dx.doi.org/10.1016/j.jnca.2016.02.011
1084-8045/

intersection. However, this assumption does not hold true for all situations. For instance, if we assume that a single acoustic sensor is monitoring two possible sound sources, in general, the sensor will be able to detect that a sound was generated, but not which source generated it – at least not unless the sound generated by each source is distinct, and even then, not without further digital signal processing.

As a slight deviation from all previous studies, we shall work with the following assumptions:

1. Targets generate *events*, and these events randomly occur. Events occur only one at a time, and they are brief enough that they do not overlap in time.
2. Events can be detected by the sensor if the source is within the sensor's sensed region, but the events themselves say nothing about which source generated it.

In this work, instead of just focusing on the number of targets covered, we will also concern ourselves with the identifiability of the targets or event sources.

As a motivating example, we present Fig. 1, where the dots s_1 and s_2 are sensors while the stars a_1–a_5 are the targets. The dotted regions around each sensor represent the possible covered or sensed region of each sensor. Each sensor can be oriented to be in 1 of 4 possible orientations. Orientation 1 represents the region covered by the sector from 0° to 90°, 2 covers 90° to 180°, 3 covers 180° to 270°, and 4 covers 270° to 360°. In this example, the boundaries of the sensed regions of each sensor are aligned with the cardinal directions (and with each other). The 4 possible orientations for s_1 are labeled in Fig. 1. It must be noted however that such an alignment is not required by the algorithms that will be discussed. In Fig. 1 s_1 is in orientation 1, while s_2 is in orientation 2. With these orientations, a_2 is covered by s_1, a_3 is covered by s_1 and s_2, and a_4 is covered by a_2. A total of 3 targets are covered. a_1 and a_5 are not covered.

It is easy to verify that 3 is the maximum number of targets that can be covered by any network configuration. A *network configuration* is a set of active sensors, each with its corresponding orientation.

However, the configuration $s_1 - 1$ (shorthand for s_1 in orientation 1), $s_2 - 2$ is not the only configuration that yields 3 covered targets. $s_1 - 2$, $s_2 - 2$ also yields 3 covered targets, as will $s_1 - 1$, $s_2 - 1$. There are differences however in the identifiabilities conferred by these configurations to the targets that they cover.

Let us begin with $s_1 - 2$, $s_2 - 2$. When a_1 generates an event, s_1 will be able to detect it, and we know for certain that a_1 generated the event since it is the only target covered by s_1. When a_3 generates an event, s_2 will be able to detect it, but we are not sure whether it was a_3 or a_4 that generated the event. The best that can be done is hazard a guess with 50% probability of being correct. The same analysis holds for $s_1 - 1$, $s_2 - 1$.

Compare this with $s_1 - 1$, $s_2 - 2$. When a_2 generates and event, s_1 will be able to detect it. At first glance, it seems like we might not be able to distinguish whether it was a_2 or a_3 which generated the event since both are covered by s_1. However, we can know that it is *not* a_3, since s_2 did not detect anything. Hence, it must be a_2 which generated the event. In other words, whether a target generated an event or not can be deduced not just from which sensors detected something, but also from those that did *not* detect anything.

We call the set of sensor states (where *state* indicates whether a sensor detected something or not) which signifies a target generating an event as the target's *syndrome*. A syndrome is a tuple of values, one for each active sensor in the system, each denoting whether a sensor will detect anything upon the target generating an event. For a sensor x, let s_x be the element for the sensor in the tuple if the sensor will detect anything, and $s_x{}'$ if it will not. In our latest example, a_2 has the syndrome $s_1 s_2'$. Table 1 enumerates the syndrome for each covered target in each of the network configuration that yields 3 covered targets.

In Table 1, we can clearly see why the configuration $s_1 - 1$, $s_2 - 2$ affords better target identifiability: in that configuration, each covered target has its own syndrome. In comparison, in the other two configurations, two targets have to share a single syndrome, resulting in ambiguity when identifying their events.

It must be noted that some aspects of Fig. 1 do not hold true in other situations.

Firstly, sometimes, assigning a single syndrome for each target is simply impossible. Nevertheless, it is conceivable that even in such situations, it is desirable to minimize the ambiguity between events as much as possible.

Secondly, in other networks (especially those that are *underprovisioned*, meaning there are few sensors relative to targets), it is possible that improved target identifiability will come at the cost of less targets covered. The acceptable trade-off between the number of targets covered and their identifiability will vary from one application to the next.

Another use of the concept of identifiability is in *overprovisioned* networks – that is, networks where there is a surplus in the number of sensors, and even after the maximum possible number of targets has been covered, there are still sensors that are not covering anything. In previous studies, the extra sensors are used in extending the network lifetime: sensors form *cover sets* that take turns covering the targets (Cai et al., 2007). Clearly, another possible use for such extra sensors is in increasing the identifiability of targets.

This paper makes three contributions: firstly, the introduction of the concept of *syndromes*; secondly, the definition of the Maximum Target Identifiability-Aware Utility with Minimum Sensors (MTIAUMS) problem; and finally, six heuristic algorithms that determine network configurations that strike a balance between identifiability and the number of targets covered.

This paper is structured as follows. The notations utilized are discussed in Section 2. The problem is formally defined, and its NP-hardness proven in Section 3. Centralized heuristic algorithms are presented in Section 4, while distributed heuristic algorithms are presented in Section 5. The methodology used to test the algorithms, and the results of the simulations, are presented in Section 6. A discussion of related work follows in Section 7, while Section 8 concludes the paper.

2. Notations

The following notations are used in this paper

- M: number of targets
- a_i: a specific target, $1 \le i \le M$
- A: the set of all targets $A = \{a_1, a_2, \dots a_M\}$
- N: number of nodes
- s_i: a specific node, $1 \le i \le N$
- S: the set of all nodes $S = \{s_1, s_2, \dots s_N\}$
- r: the sensing radius; a target is said to be covered by one of the node's orientations if the distance between the target and the node is less than or equal to the sensing radius
- W: number of orientations with which each node can work with
- $\phi_{i,j}$: set of targets covered by node i when it is working with orientation j, $1 \le i \le N, 0 \le j \le W$. Note that we allow j to take on the value of 0: this indicates that the node's sensing mechanism is inactive, thus $\phi_{i,0} = \{\}, \forall i$
- Φ_i: set of all targets within range of s_i, *regardless* of orientation; $\Phi_i = \{\cup \phi_{i,j} \mid 1 \le j \le W\}$
- Q_i: set of nodes comprised of one-hop neighbours of s_i
- Φ_i': set of all targets within range of s_i's one-hop neighbours, sans those also seen by s_i; $\Phi_i' = \{\cup \Phi_j \setminus \Phi_i \mid s_j \in Q_i\}$