Poster Abstract: Neighborhood Cardinality Estimation in Extreme Wireless Sensor Networks

Marco Cattani* Marco Zuniga Andreas Loukas Koen Langendoen

{m.cattani, m.a.zunigazamalloa, a.loukas, k.g.langendoen}@tudelft.nl

Delft University of Technology, The Netherlands

***Abstract*—We address the problem of estimating the neighborhood cardinality of nodes in extreme wireless sensor networks, where nodes are mobile, densities reach hundreds of neighbors, and *all* nodes want to estimate cardinality concurrently. Estimators need not only be accurate, but also fast, asynchronous (due to mobility), and lightweight (due to concurrency and high density). To cope with these requirements, we propose a mechanism with extremely low overhead that leverages the rendezvous time of duty-cycled medium access control (MAC) protocols; the shorter the rendezvous, the higher the cardinality, and vice versa. Implemented on Contiki OS and deployed on a 100-node testbed, our estimator determines neighborhood cardinalities with less than 10% error.**

I. Motivation

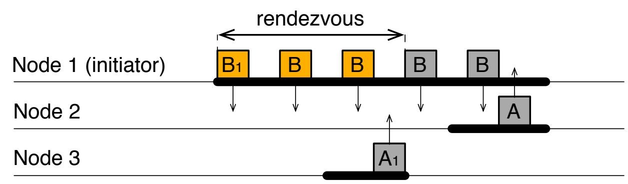
Our work is part of a larger project (COMMIT/) related to public safety. The goal is to provide coin-sized devices to attendees in large-scale open-air festivals, and issue alerts when the crowd *density* crosses a dangerous threshold. In this type of applications, all devices need to periodically estimate their surrounding density, which can reach to levels of hundreds of nodes. State-of-the-art estimators achieve an accuracy between 3% and 35% for up to 25 smartphones, relying on audio signals [1], bluetooth signals [2], or radio signals [4]. However, only a fraction of the nodes perform the estimation process, rendering them unsuitable for our application. In this work we therefor propose a low-overhead cardinality estimator that is robust to high densities and mobility, and show that it scales from a few tens of neighbors to one hundred neighbors, while allowing all nodes to perform the estimation concurrently.

II. Mechanism

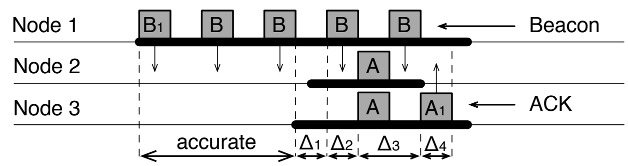
The key idea of our estimator is simple: in duty-cycled MAC protocols nodes periodically wake up within a given period (t_w) to listen for incoming messages. The time difference between two consecutive nodes' wake-ups (*rendezvous time*) captures the density of the neighborhood (n). The shorter the rendezvous time, the higher the density, and vice-versa. According to order statistics, the rendezvous time can be modeled using a Beta random variable U_k with expected value equal to $t_w/(n+1)$. Inverting this expectation, we can use the average of the observed rendezvous $\bar{u_k}$ to estimate the neighborhood cardinality $\hat{n} = (t_w/\bar{u_k}) - 1$.

III. Implementation

We implemented our estimator using a slightly modified version of Low Power Listening (LPL). As usual, when a node, called *initiator*, wants to communicate (see Figure 1(a)), it starts sending a strobe of beacons (B). During this strobe, the initiator announces that the first node has not yet been discovered by setting a flag in its beacons (orange packets). When the *first* neighbor wakes up (node 3), it receives a flagged beacon and sends an acknowledgement (A1). After receiving A1, the initiator clears the flag and continues its strobe normally until the intended destination (node 2) responds.



(a) Discovery the first awake neighbor with low power listening (LPL)



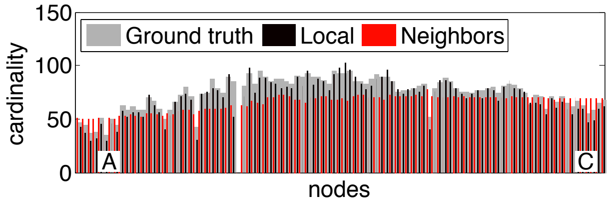
(b) Delays in the measurement of the rendezvous time.

Fig. 1. Accurately measure the rendezvous time is essential for our estimator.

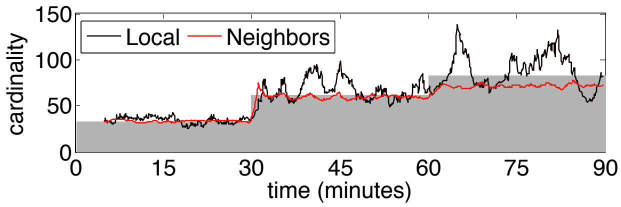
Measure the rendezvous. In principle, the rendezvous time can be obtained by measuring the time elapsed between the first beacon (B1) and the first acknowledgement (A1). Unfortunately, as Figure 1(b) shows, the rendezvous phase includes different kinds of delays; namely, collisions ($\Delta 3$), the transmission time of radio packets ($\Delta 2$, $\Delta 4$), and other delays that are characteristic of the MAC protocol ($\Delta 1$). To obtain an accurate measurement, the receiver (node 3) piggybacks the time elapsed since it woke up ($\Delta 1$+$\Delta 2$+$\Delta 3$) on its acknowledgment (A1).

Efficient collection. As any other estimator based on order statistics, the larger the number of samples, the closer the mean gets to the expected value and the more accurate the estimation. Unfortunately, in mobile networks nodes only have a limited amount of time to capture as many samples as possible. Thus, the number of samples w used to compute the average trades off accuracy for adaptability: the bigger w, the more time it takes to adapt to changes in cardinality. For this reason, whenever two nodes rendezvous, they exchange the average of their local samples. This exploits the spatial correlation on nodes' density and drastically increases the

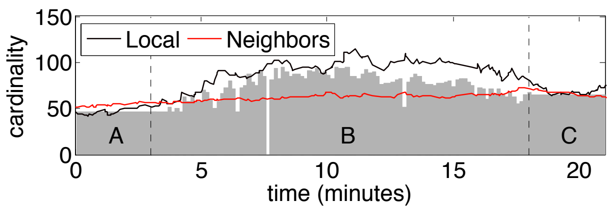
*Marco Cattani was supported by the Dutch national program COMMIT/



(a) Uneven density in space. Since nodes are installed along the corridor of our building, at the two far-ends (A and C) the density is the smallest.



(b) Uneven density over time. We start with the 30 nodes placed in the center of the corridor. Then, we expand towards the far ends.



(c) Three mobile nodes move from one end of the corridor to the other.

Fig. 2. Cardinality estimation in different dynamic scenarios.

number of samples collected in time. Note that this is only possible because our estimator is designed to allow *all* nodes to perform concurrent estimations.

IV. Evaluation

To evaluate our estimator, we ran a set of experiments on a testbed consisting of 100 nodes equipped with an MSP430 processor and a CC1101 radio [3]. For each experiment, we compute the ground-truth cardinality of a node as the number of neighbors that successfully exchanged a message at least once during the duration of an experimental run. With regards to the parameters, unless stated otherwise, our experiments use the following values: wakeup period (t_w) = sampling period = 1 s, window size (w) = 50, network size = 100 nodes. Note that in the experiments, we differentiate between estimations computed using only local samples (*local*) and using neighbors' samples (*neighbors*) to highlight the different characteristics of the two.

Uneven density in space. For most cardinalities, our estimator has a relative error of ≈10% for *local* and under 5% for *neighbors*. The main exception occurs at $n = 100$, when uneven densities among nodes badly affect the spatial averaging mechanism, see Figure 2(a). For *local*, each node has a precise view of its own neighborhood, as shown by the accurate mapping between the individual node estimations and the ground truth. For *neighbors* however, by averaging the views of its neighbors, a node with a low cardinality (with regards to its neighbors) will overestimate its density, and vice versa.

Uneven density over time. Figure 2(b) shows a series of 90-minute experiments in which the neighborhood cardinality grows by 30 nodes each 30 minutes. With the same w, *neighbors* has a lower variance (better accuracy) and takes less time to adapt to changes (≈ 1 minute). During this time, a walking person (1 m/s) can cover approximately 60 meters while sampling the current neighborhood. Assuming a device with a transmission range of 50 m, our estimator should be able to cope with the dynamics of our application scenarios (crowd monitoring in open-air festivals).

Mobile nodes. As a final experiment, we equipped 3 colleagues with a sensor node and asked them to move according to a predefined path, see Figure 2(c). The experiment lasted 20 minutes. In the first 3 minutes, we asked our colleagues to stand on one far-end of the testbed. Subsequently, they slowly moved to the other end of the testbed (C). The slow movement (section B) was required to get an accurate measurement of the ground truth: at each step we waited approximately 10 s to obtain a reasonably accurate snapshot of the cardinality of the testbed's node that was closest to the mobile node. Figure 2(c) shows the estimated neighborhood cardinality of one of the mobile nodes and highlights the tradeoff between *local* and *neighbors*. If a quick and rough estimation is required, *neighbors* is the best solution. On the other hand, if a more accurate, but longer, measurement is needed *local* should be used.

V. Conclusions

In this poster we address the issue of determining the neighborhood cardinality in extreme wireless sensor networks, where node mobility necessitate a robust and agile approach. Moreover, we support high densities and concurrent estimations, requirements that are necessary in public safety applications such as crowd monitoring. Since traditional approaches cannot meet these stringent requirements, we developed an estimator based on observing the inter-arrival times of nodes waking up. We implemented our estimator on the Contiki OS, and evaluated it on a testbed with node densities up to 100 nodes. Our estimator achieves solid performance results with typical estimation errors below 10%, which compares favorably to state-of-the-art solutions. The estimator was also demonstrated to handle abrupt changes in density, exemplified through a few nodes moving through our testbed.

References

[1] P. Kannan and S. Venkatagiri. Low cost crowd counting using audio tones. In *Embedded Network Sensor Systems*, pages 155–168, 2012.
[2] J. Weppner and P. Lukowicz. Collaborative Crowd Density Estimation with Mobile Phones. In *Embedded Networked Sensor Systems*, 2011.
[3] M. Woehrle, M. C. Bor, and K. G. Langendoen. 868 MHz: a noiseless environment, but no free lunch for protocol design. In *Networked Sensing Systems*, INSS, pages 1–8, 2012.
[4] W. Zeng, A. Arora, and K. Srinivasan. Low power counting via collaborative wireless communications. In *Information processing in sensor networks (IPSN)*, pages 43–54, 2013.