Exploiting Rush Hours for Energy-Efficient Contact Probing in Opportunistic Data Collection

Xiuchao Wu, Kenneth N. Brown, and Cormac J. Sreenan
Mobile and Internet Systems Laboratory and Cork Constraint Computation Centre
Department of Computer Science, University College Cork, Ireland
Email:{x.wu, k.brown, cjs}@cs.ucc.ie

***Abstract*—In many potential wireless sensor network applications, the cost of the base station infrastructure can be prohibitive. Instead, we consider the use of mobile devices carried by people in their daily life to collect sensor data opportunistically. As the movement of these mobile nodes is, by definition, uncontrolled, contact probing becomes a challenging task, particularly for sensor nodes which need to be aggressively duty-cycled to achieve long life.**

It has been reported that when the duty-cycle of a sensor node is fixed, SNIP, a sensor node-initiated probing mechanism, performs much better than mobile node-initiated probing mechanisms. Considering that the intended applications are delay-tolerant, mobile nodes tend to follow some repeated mobility patterns, and contacts are distributed unevenly in temporal, SNIP-RH is proposed in this paper to further improve the performance of contact probing through exploiting *Rush Hours* during which contacts arrive more frequently.

In SNIP-RH, SNIP is activated only when the time is within *Rush Hours* and there are enough data to be uploaded in the next probed contact. As for the duty-cycle, it is selected based on the mean of contact length that is learned online. Both analysis and simulation results indicate that under a typical simulated roadside wireless sensor network scenario, SNIP-RH can significantly reduce the energy consumed for probing the contacts, that are necessary for uploading the sensed data, or significantly increase the probed contact capacity under a sensor node's energy budget for contact probing.

I. Introduction

As wireless sensor networks mature, we expect to see long-term deployments for applications such as environmental monitoring, house water/gas/electricity meter reading, and structural health monitoring. These applications typically involve large numbers of sparsely deployed (static) sensor nodes that report data that is inherently delay tolerant, since the response (if any) requires human intervention over long time scales. For example, analysis of environmental monitoring data is rarely urgent, and meter readings for billing purposes can be delayed by weeks. Neighboring nodes in these sparse wireless sensor networks are far away from each other, and typically cannot communicate directly or even indirectly through multi-hop paths. On the other hand, deploying large numbers of fixed sink nodes would incur prohibitive costs in terms of deployment, maintenance, and data back-haul.

In [1][2][3][4][5][6][7], the use of resource-rich mobile nodes (mobile sinks or mobile relays) was proposed to move around in the deployed area and collect data from sensor nodes. Depending on the application, the mobile nodes can be either part of the external environment or part of the network, and their mobility can be either controllable or not. In this paper, we assume that mobility is not controlled and thus the sensed data is collected opportunistically. Mobile nodes could be specific devices carried by objects (animals, employees, etc.) who move around the deployed area for purposes other than data collection. More interestingly, as illustrated in figure 1, they could also be smart phones and/or PDAs (installed with the corresponding radio and software) carried by unrelated people who pass through the deployed area in their daily life. Except the benefits of adopting mobile sinks discussed in [1][3] (the energy efficient one-hop data collection, the extended network lifetime through removing hotspots near the fixed basestation, etc.), the cost of data collection can also be reduced significantly through exploiting the uncontrollable, but free human mobility. Although opportunistic data collection may significantly increase the data delivery latency [1], there are many promising wireless sensor network applications which are delay-tolerant and it is worthwhile to improve the performance of opportunistic data collection.

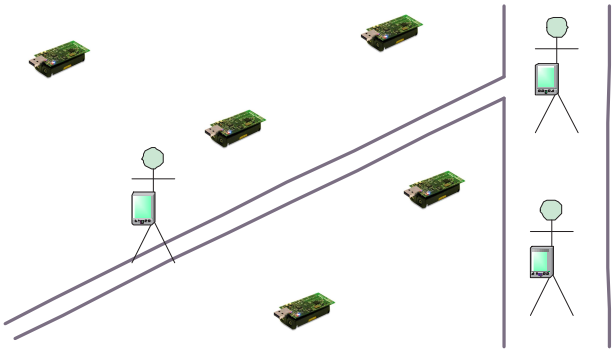


Fig. 1: Opportunistic Data Collection with Smart Phones

In opportunistic data collection, the sensed data can be collected from a sensor node only after a mobile node approaches and they become aware of each other. Hence, for contact probing, there are four processes in the system: the movement of a mobile node, the radio schedule of a mobile node, the radio schedule of a sensor node, and the beacons periodically transmitted by either mobile node or sensor node. To establish successful contact, a beacon must be sent out by one of them when they are close to each other and their radios are both

turned on. In other words, all four processes must occur at the same time. This can be difficult to achieve when mobile node's movement is uncontrollable and sensor node is required to maintain aggressive duty-cycles for reasons of life longevity.

Since the mobility in opportunistic data collection is uncontrollable, a contact probing mechanism is limited to control the broadcasting of beacons and the radio schedules of mobile node and sensor node. Considering that a mobile node could have relatively abundant energy via a re-chargable battery, the radio of mobile node can be always turned on [8][9]. Hence, it only needs to answer the following two questions.

1) For improving the probed contact capacity when the duty-cycle of a sensor node is fixed, who should be responsible to broadcast the beacons?
2) For energy-efficiently probing the necessary contacts for uploading its sensor reports, how should the sensor node schedule its radio used for contact probing?

The first question has been studied by us and SNIP, a sensor node-initiated probing mechanism, is proposed in [10]. In this paper, we will study the second problem, i.e., how to select the duty-cycle used by SNIP for energy efficiently probing the necessary contacts.

Considering that the expected applications of opportunistic data collection are delay tolerant, a sensor node could have more freedom when scheduling SNIP operations. Based on the observations that mobile nodes (smart phones carried by people) normally follow some repeated mobility patterns and the temporal distribution of contacts within an epoch of mobile nodes' mobility pattern tends to be uneven, SNIP-RH is proposed in this paper for energy efficiently probing the necessary contacts through exploiting *Rush Hours*, during which contacts arrive more frequently.

In SNIP-RH, SNIP is activated only when the time is within *Rush Hours* and there are enough data to be uploaded in the next probed contact. As for the duty-cycle, it is selected based on the mean of contact length that is learned when SNIP is activated. SNIP-RH has been implemented in Contiki-OS and simulated with COOJA. The preliminary analysis and simulation results indicate that under a typical simulated road-side wireless sensor network scenario, SNIP-RH can significantly reduce the energy consumed for probing the necessary contacts or significantly increase the probed contact capacity under a sensor node's energy budget for contact probing.

This paper is organized as follows. Section II first introduces the reference model for contact probing in opportunistic data collection and SNIP is briefly introduced in section III. The motivations of exploiting *Rush Hours* are then discussed in section IV. After that, section V models the scheduling of SNIP for exploiting *Rush Hours* as an optimization problem. SNIP-RH, a much more practical scheduling mechanism, is then presented in section VI. Evaluation results are also presented and discussed in VII. Finally, section VIII compares with related work and section IX concludes.

II. Reference Model for Contact Probing

Figure 2 illustrates the reference model for contact probing in opportunistic data collection. The mobile node's mobility is uncontrollable and cannot be predicted accurately by sensor nodes. For simplicity, we assume that the network is spare enough so that at any time at most a single (static) sensor node and a single mobile node can reach each other. In the case that multiple mobile nodes move together, this assumption can be easily removed by adopting some collision avoidance or contention resolution techniques [11] and allowing a sensor node to choose one of these mobile nodes randomly or based on their radio signal strength, movement speed, etc. We also assume that the same commodity radio (Zigbee-compilant radio [12], etc.) is installed on both mobile nodes and sensor nodes, i.e., they have the same communication range (R). When carrying out contact probing, the radio of a sensor node is duty-cycled for achieving a long life. More specifically, the radio is turned on for a fixed period (T_{on}) and turned off for another fixed period (T_{off}) alternatively. Hence, the duration of a cycle (T_{cycle}) is the sum of T_{on} and T_{off} and the duty-cycle (d) equals to T_{on}/T_{cycle}.

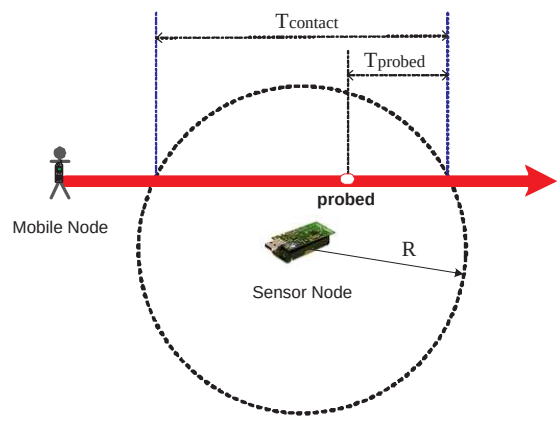


Fig. 2: Contact Probing in Opportunistic Data Collection

Under this scenario, the sensed data can be collected from a sensor node only after a mobile node approaches and they become aware of each other. As shown in figure 2, the event of the mobile node encountering a sensor node is referred to as a contact and the contact length ($T_{contact}$) is the duration for which the mobile node stays within the communication range of the sensor node. As for T_{probed}, it starts immediately after both of them are aware of the presence of each other and it can be used to derive the amount of data that could be collected in this contact. For a contact probing mechanism, it should be designed so that a contact can be successfully probed with high probability and the contact is probed as early as possible. More specifically, when a sensor node's duty-cycle is fixed, a contact probing mechanism should try to maximize $\Upsilon = \frac{T_{probed}}{T_{contact}}$, the percent of contact capacity that is probed successfully for data collection. Table I lists the notations used here and the following sections.

Notation	Comments
R	the communication range of the radio used by all nodes
T_{on}	the period that sensor node's radio is turned on
T_{off}	the period that sensor node's radio is turned off
d	the duty-cycle of a sensor node
T_{cycle}	the cycle length of a duty-cycled sensor node $T_{cycle} = T_{on} + T_{off}$, $d = T_{on}/T_{cycle}$
$T_{contact}$	the length of a contact
T_{probed}	the time probed for data collection during a contact
Υ	the percent of probed contact capacity ($\Upsilon = \frac{T_{probed}}{T_{contact}}$)
$T_{interval}$	the time between two consecutive contacts
T_{epoch}	the epoch length of mobile nodes' mobility pattern
N	the number of time-slots of an epoch
ζ	the contact capacity probed during an epoch
ζ_i	the contact capacity probed during the ith time-slot
ζ_{target}	the amount of contact capacity that is just enough to transmit the sensor reports generated in an epoch
Φ	the contact probing overhead. It is the time that the radio is turned on during an epoch
Φ_{max}	energy budget of a sensor node. It is the maximal time that the radio can be turned on during an epoch.
ρ	cost for per unit of probed contact capacity ($\rho = \Phi/\zeta$)

TABLE I: Notations

III. A Brief Introduction to SNIP

To answer the first question faced by contact probing in the context of opportunistic data collection, SNIP, a Sensor Node-Initiated Probing mechanism, was proposed in [10]. SNIP is designed based on the following two reasonable assumptions, i.e., the radio of mobile nodes, which have relatively abundant energy via a re-chargable battery, can be always turned on and the radio of sensor nodes consumes almost the same amount of energy in transmitting and receiving/listening modes [8][9]. The basic principle of SNIP is that the sensor node initiates probing rather than a mobile node. Thus a sensor node will broadcast a beacon immediately after its radio is turned on according to its duty-cycle. Since the radio of a mobile node is always turned on, if sensor node broadcasts a beacon when they are close to each other, this contact will be definitely probed successfully, assuming of course that the beacon is not lost or corrupted due to contention, which is unlikely in sparse deployments and short range transmissions.

SNIP has been analyzed and the relationship among Υ, d, and $T_{contact}$ is modeled. More specifically, based on the assumption that mobile node is uncontrollable, T_{probed} is first modeled and it is straightforward to deduce $\Upsilon(d, T_{contact}) = \frac{T_{probed}}{T_{contact}}$. The model shows that Υ increases with d, and $T_{contact}$ significantly affects the curve. Another key observation is that when $T_{cycle} = \frac{T_{on}}{d} \geq T_{contact}$, i.e., $d \leq \frac{T_{on}}{T_{contact}}$, Υ is linearly related with d. In fact, closed-form equation can be deduced through modeling the following two cases separately.

$$\Upsilon = \begin{cases} \frac{T_{contact}}{2*T_{on}} * d & T_{cycle} \geq T_{contact} \\ 1 - \frac{T_{on}}{2*d*T_{contact}} & T_{cycle} < T_{contact} \end{cases} \quad (1)$$

SNIP has been implemented in Contiki-OS [13] and extensive simulations are carried out using COOJA [14]. Both the analysis and simulation results indicate that SNIP outperforms mobile node-initiated probing mechanisms [15], and we quantify the impact of key parameters. A key conclusion is that with a sensor node duty-cycle that is lower than 1%, the probed contact capacity can be increased by a factor of 2-10; alternatively, for probing the same amount of contact capacity, the energy consumed by SNIP is much less than the energy consumed by mobile node-initiated probing mechanisms. Hence, it is the sensor node who should be responsible to broadcast beacons and SNIP should be adopted. In this paper, we will study how to select the duty-cycle used by SNIP so that a sensor node can energy efficiently probe the necessary contacts for uploading its sensor reports.

IV. Motivation

The straightforward scheduling mechanism for a sensor node is to activate SNIP in *all time* with a duty-cycle, which is well selected so that the probed contact capacity is just enough to upload its sensed data. This scheduling mechanism will be referred as SNIP-AT. Considering that the intended applications are delay-tolerant, there should be further opportunities for improving the performance of contact probing. In this paper, SNIP-RH is motivated by the following observations.

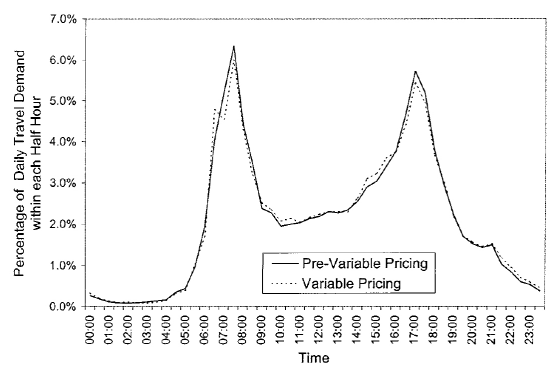


Fig. 3: Temporal distribution of eligible user travel demand at the Midpoint Bridge, Florida, USA (Fig. 1 in [16])

First, mobile nodes normally follow some repeated mobility patterns, especially when smart phones carried by people are used as mobile nodes. It has been shown that human trajectories have a high degree of temporal and spatial regularity and their mobility follows simple reproducible patterns [17]. The temporal distribution of human mobility also shows high unevenness [16]. Hence, the contacts between a sensor node and mobile nodes tend to arrive unevenly in temporal and *Rush Hours*, during which contacts arrive more frequently, exist widely in the environment. For example, figure 3 shows the temporal distribution of eligible user travel demand at the Midpoint Bridge, Florida, USA. It shows that *Rush Hours* do exist and don't disappear even after a variable pricing is adopted by the toll bridge for spreading the travel demand. The possible reason is that users must live according to the same timetable agreed by the whole society. Hence, *Rush Hours* do and will exist widely in the deployment environments. Due to the repeated pattern followed by human mobility, it also becomes possible for a sensor node to learn, predict, and exploit these *Rush Hours*.

Second, according to SNIP model, in the case that contact probing is only carried out in *Rush Hours* and the necessary contacts can be successfully probed, it consumes much less energy than SNIP-AT. Considering that the target applications of opportunistic data collection are low-data-rate wireless sensor network applications deployed in urban area with abundant human mobility, we should exploit *Rush Hours* when scheduling SNIP operations. The benefit of exploiting *Rush Hours* is further explained with the following simple analysis.

For simplicity, we assume that the length of contacts is a fixed value (l) and the epoch length is T_{epoch}. Contacts arrive with a frequency, f_{rh}, during *Rush Hours* whose length is T_{rh}. During the remaining time whose length is T_{other} ($T_{epoch} = T_{other} + T_{rh}$), contacts arrive with another frequency f_{other}. As for the necessary contacts for uploading all sensed data, they could be successfully probed by SNIP-AT with a duty-cycle (d_0) and they could also be probed when SNIP with another duty-cycle (d_1) is carried out only during *Rush Hours*. Hence, $T_{rh} * f_{rh} * l * \Upsilon(d_1, l) = (T_{rh} * f_{rh} + T_{other} * f_{other}) * l * \Upsilon(d_0, l)$. Consequently, $\frac{\Upsilon(d_0,l)}{\Upsilon(d_1,l)} = \frac{T_{rh}*f_{rh}}{(T_{rh}*f_{rh}+T_{other}*f_{other})}$.

Since a sensor node needs to be aggressively duty-cycled for life longevity, we also assume that both d_0 and d_1 are small enough so that Υ is linearly related with d, i.e., $\frac{\Upsilon(d_0,l)}{\Upsilon(d_1,l)} = \frac{d_0}{d_1}$. When SNIP-AT is used, the energy consumed for contact probing is $\Phi_{AT} = T_{epoch} * d_0 = (T_{rh} + T_{other}) * d_0$. When SNIP is carried out only during *Rush Hours*, the energy consumption is $\Phi_{rh} = T_{rh} * d_1 = \frac{(T_{rh}*f_{rh}+T_{other}*f_{other})*d_0}{f_{rh}} = T_{rh} * d_0 + \frac{T_{other}*f_{other}*d_0}{f_{rh}}$.

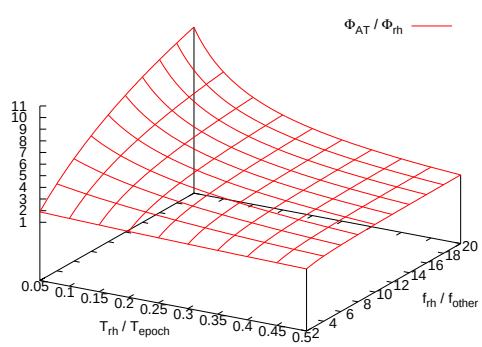


Fig. 4: Benefits of activating SNIP only during *Rush Hours*

Figure 4 plots the potential improvement in the metric of energy efficiency when SNIP is carried out only during *Rush Hours*. Considering that T_{rh} is normally a small percent of T_{epoch} and $f_{other} \ll f_{rh}$, it should be clear that $\Phi_{rh} \ll \Phi_{AT}$, i.e., exploiting *Rush Hours* can significantly reduce the energy consumed for contact probing.

In summary, *Rush Hours* exist widely, and they could and should be exploited when scheduling SNIP operations.

V. Modeling and Optimization-based Scheduling

Since the purpose of exploiting *Rush Hours* is to improve the performance of contact probing, SNIP scheduling will be modeled as an optimization problem in this section. More specifically, we assume that a sensor node has a target of the probed contact capacity, ζ_{target}, which is just enough to upload the sensor reports generated for satisfying Quality of Service requirements of the application. We also assume that a sensor node has a budget for the energy consumed by contact probing (Φ_{max}) so that it can assure a minimal lifetime. When it is possible to probe the necessary contacts (ζ_{target}) under the energy budget, a sensor node will try to minimize the energy consumption for extending its life. Otherwise, a sensor node will maximize the probed contact capacity under the energy budget and adjust the amount of sensor reports accordingly.

We assume that the epoch length of mobile nodes' repeated mobility pattern is T_{epoch} and an epoch is divided into N time-slots with the following length, t_1, t_2, ..., t_n. We also assume that the contact arrival process of each time-slot (both contact arrival frequency and contact length distribution) can be learned accurately. Based on the learned contact arrival process and the closed-form equation of SNIP (equation 1), we can deduce $\zeta_i(d_i)$, which is the amount of contact capacity probed during time-slot i when SNIP is carried out with a duty-cycle d_i. With a scheduling plan (d_1, d_2, ..., d_n), the total amount of probed contact capacity is $\zeta = \sum_{i=1}^{i=n} \zeta_i(d_i)$ and the energy consumed for contact probing is $\Phi = \sum_{i=1}^{i=n} t_i * d_i$. Hence, the task of scheduling the radio of a sensor node becomes a decision of the value of d_i used by SNIP during each time-slot for improving the performance of contact probing.

The SNIP scheduling problem can then be solved through the following two steps. In the first step, a sensor node tries to maximize ζ under the constraints that $\Phi \leq \Phi_{max}$ and $0 \leq d_i \leq 1$ ($i = 1, 2, ..., n$). If the maximal ζ is less than ζ_{target} which is just enough to upload all of the sensor reports generated with the target rate, the sensor node has the optimal scheduling plan now and it should reduce its data generation rate accordingly. Otherwise, the second step will be executed. In the second step, the sensor node will try to minimize Φ under the constraints that $\zeta \geq \zeta_{target}$ and $0 \leq d_i \leq 1$ ($i = 1, 2, ..., n$). Hence, the life of the sensor node can be maximized. The formal model of this two-step optimization-based scheduling mechanism is presented below and this scheduling mechanism will be referred as SNIP-OPT.

STEP 1

Objective:	Maximize ζ
Constraints:	$\Phi \leq \Phi_{max}$
	$0 \leq d_i \leq 1$, for each i

STEP 2

Objective:	Minimize Φ
Constraints:	$\zeta \geq \zeta_{target}$
	$0 \leq d_i \leq 1$, for each i

Although SNIP-OPT can produce the optimal scheduling plan, it may not be applicable in the real world. SNIP-OPT assumes that a sensor node knows the exact contact arrival process (both contact arrival frequency and contact length

distribution) for each time-slot. It is very hard for engineers to get all of these information for each sensor node. It is also not suitable to let sensor nodes learn this information and execute SNIP-OPT autonomously. First, the CPU of sensor node may not be powerful enough to solve the optimization problem in SNIP-OPT. Second, considering the large number of parameters to be estimated (contact arrival frequency and contact length distribution for each time-slot) and the low duty-cycle that must be used for life longevity, it is very challenging for a sensor node to learn the contact arrival process as required by SNIP-OPT.

VI. SNIP-RH

Although it is hard to get to know the exact contact capacity of each time-slot, it should be easy to determine the time-slots with more contact capacity. Based on this observation, SNIP-RH is designed in this paper for exploiting *Rush Hours* of the environment. The main principle of SNIP-RH is that SNIP is activated only during *Rush Hours*. Its details will be presented and discussed in this section.

A. Rush Hours

An epoch of mobile nodes' repeated mobility pattern is first divided into N time-slots with the same length and each time-slot is marked as "1" or "0". "1" indicates that a time-slot is in *Rush Hours*. N and T_{epoch} (the length of an epoch) need to be determined by engineers based on the deployment environment. T_{epoch} should equal to the epoch length of mobile nodes' repeated mobility pattern. As for N, it should be well selected based on the mobility pattern and the available resources of a sensor node. With a larger N, *Rush Hours* can be specified more accurately, but it takes more effort to identify *Rush Hours* among these time-slots.

When smart phones carried by people act as mobile nodes, T_{epoch} can be set to 24 hours since human mobility follows the diurnal pattern. As for N, it can be set to 24 so that the length of each time-slot is exactly one hour.

B. SNIP Scheduling

In SNIP-RH, we assume that the CPU of a sensor node wakes up periodically to decide whether to carry out SNIP. The sensor node will activate SNIP only when all of the following three conditions are satisfied.

1) To exploit *Rush Hours*, the current time-slot must be marked as "1".
2) The sensor node should also have enough data to be uploaded in the next probed contact. Hence, the probed contact capacity will not be wasted and the energy consumption for contact probing can be reduced. The threshold for available data can be set according to the amount of data uploaded in previous probed contacts.
3) The energy that had been consumed for contact probing in the current epoch should be less than the sensor node's energy budget for contact probing.

Hence, a sensor node needs to maintain the energy that it consumed for contact probing in the current epoch. It also needs to keep updating the average amount of data uploaded during a probed contact. To filter out the noise in the amount of data uploaded during a probed contact, an exponentially weighted moving average (EWMA) is used and a small weight is assigned to the new sample.

C. SNIP's Duty-cycle

The mean of contact length ($\bar{T}_{contact}$) is also learned for selecting d_{rh}, the duty-cycle used by SNIP when it is activated. To filter out the noise in the probed contact length, an EWMA is also used here and a small weight is assigned to the new sample. In SNIP-RH, d_{rh} is set to $T_{on}/\bar{T}_{contact}$ and this choice is made based on the following observation.

According to SNIP model, when $d \leq T_{on}/\bar{T}_{contact}$, Υ is increased linearly with the increase of d. When d is increased further, Υ is increased much slower. Consequently, when $d \leq T_{on}/\bar{T}_{contact}$, the energy cost for per unit of the probed contact capacity (ρ) will not change with d. And ρ will be increased with d if $d > T_{on}/\bar{T}_{contact}$. Here, $\rho = \frac{\Phi}{\zeta}$. Φ is the energy consumed for contact probing and it is linearly related with d. As for ζ, it is the amount of probed contact capacity and it is linearly related with Υ.

Hence, it is desirable if d_{rh} is not larger than $T_{on}/\bar{T}_{contact}$. Through letting $d_{rh} = T_{on}/\bar{T}_{contact}$, a sensor node can maximize the contact capacity probed during *Rush Hours* with the smallest ρ [1]. SNIP model in [10] also indicates that ρ does not increase abruptly when d_{rh} is slightly larger than $T_{on}/\bar{T}_{contact}$. Hence, in the metric of energy efficiency, SNIP-RH is not very sensitive to the accuracy of $\bar{T}_{contact}$.

VII. Evaluation and Discussion

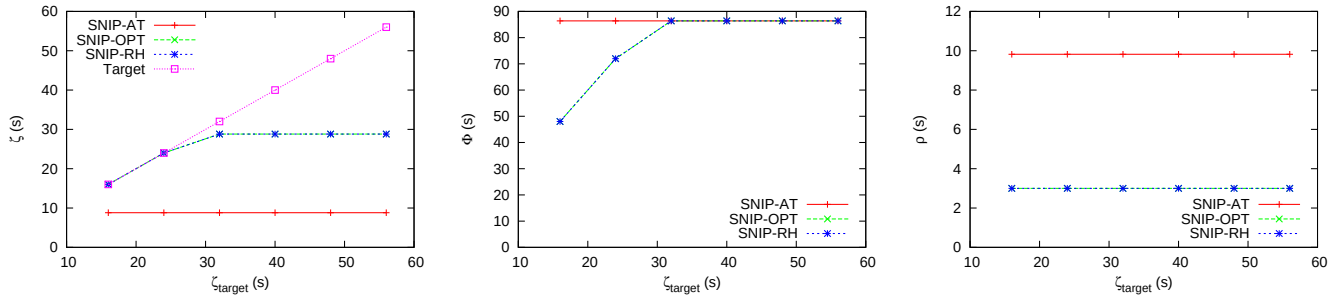
In this section, we will first present the evaluation results of SNIP-RH in a simulated roadside wireless sensor network. Its performance in dynamic environment is then discussed.

A. Evaluation

To evaluate SNIP-RH and compare it with SNIP-AT and SNIP-OPT, these scheduling mechanisms are studied under the following scenario of a simplified roadside wireless sensor network. The epoch length (T_{epoch}) is set to 24 hours, N is set to 24, and the *Rush Hours* are 7.00 to 9.00 and 17.00 to 19.00. In *Rush Hours*, the interval between two consecutive contacts ($T_{interval}$) is 300 seconds. In the other time, $T_{interval}$ equals to 1800 seconds. All of these contacts have the same length, i.e., $T_{contact}$ equals to 2s.

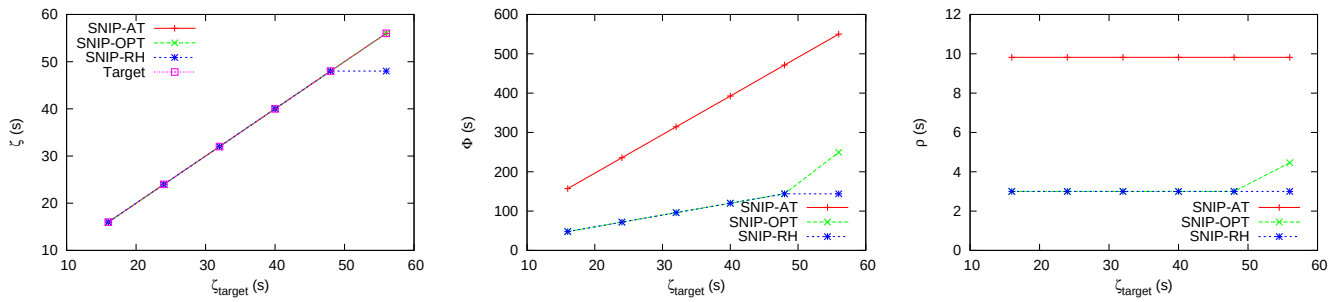
To study how well these SNIP scheduling mechanisms perform under different situations, Φ_{max} is set to $\frac{T_{epoch}}{1000}$ and $\frac{T_{epoch}}{100}$, and the target of the probed contact capacity, ζ_{target}, is set to 16s, 24s, 32s, 40s, 48s, and 56s.

[1] As shown in [10], when $T_{contact}$ follows the exponential distribution, Υ is not linearly related with d even if $T_{cycle} \geq \bar{T}_{contact}$, i.e., $d \leq T_{on}/\bar{T}_{contact}$. However, we still can oberve the obvious slope change at the point that $T_{cycle} = \bar{T}_{contact}$. Hence, $d = T_{on}/\bar{T}_{contact}$ should be a good selection for SNIP-RH even when $T_{contact}$ varies a lot.



(a) The Probed Contact Capacity (b) The Overhead of Contact Probing (c) Cost for Per-unit Probed Contact Capacity

Fig. 5: Analysis Results of SNIP Scheduling Mechanisms when $\Phi_{max} = \frac{T_{epoch}}{1000}$



(a) The Probed Contact Capacity (b) The Overhead of Contact Probing (c) Cost for Per-unit Probed Contact Capacity

Fig. 6: Analysis Results of SNIP Scheduling Mechanisms when $\Phi_{max} = \frac{T_{epoch}}{100}$

1) Numerical Results: Based on the models of these scheduling mechanisms, we first present and analyze their numerical results under the above scenario of a simplified roadside wireless sensor network.

Figure 5 shows the probed contact capacity (ζ_{probed}), the energy consumed by contact probing (Φ), and the energy cost for per unit of the probed contact capacity (ρ) during an epoch when $\Phi_{max} = \frac{T_{epoch}}{1000}$. These plots indicate that in all metrics, SNIP-RH performs much better than SNIP-AT and its performance is same with SNIP-OPT. When $\zeta_{target} \leq 24s$, SNIP-AT cannot probe the necessary contacts under the energy budget, but SNIP-RH still can energy efficiently probe the necessary contacts. When $\zeta_{target} > 24s$, although all of these scheduling mechanisms cannot probe the necessary contacts under the energy budget, compared with SNIP-AT, SNIP-RH can probe much more contact capacity with a much lower energy cost for per-unit of probed contact capacity.

Figure 6 shows the results when $\Phi_{max} = \frac{T_{epoch}}{100}$. These plots indicate that when Φ_{max} is large and $\zeta_{target} <= 48s$, SNIP-RH can probe the necessary contacts much more energy efficiently than SNIP-AT. Hence, SNIP-RH is a very good scheduling mechanism when low data rate wireless sensor network applications are deployed in urban areas with many mobile objects, i.e., abundant contacts. Figure 6 also indicates that when $\zeta_{target} = 56s$, the contact capacity in *Rush Hours* is not high enough and SNIP-RH can not probe the targeted contact capacity. SNIP-AT and SNIP-OPT can achieve this target in this case, but they also have a higher energy cost for per-unit of probed contact capacity. Depending on the application, it may be worthwhile to use a larger d_{rh} and/or mark more time-slots as *Rush Hours* for increasing the probed contact capacity. This issue will be studied in the future.

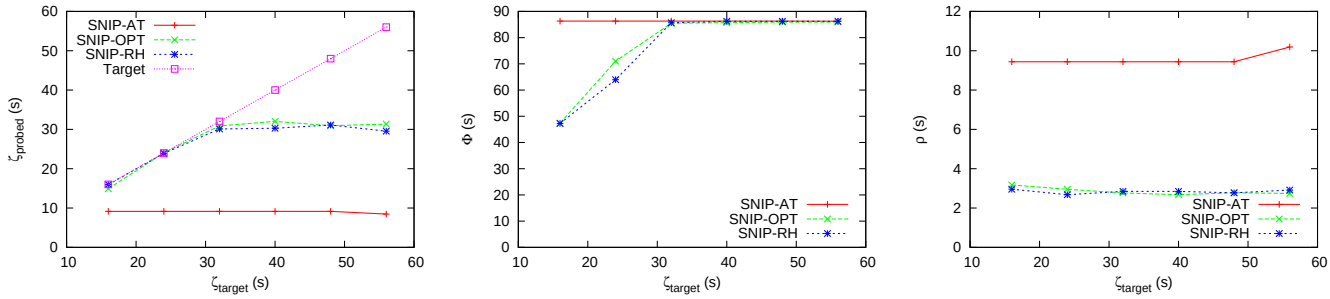
2) Simulation Results: To validate the above analysis, SNIP-AT, SNIP-OPT, and SNIP-RH are all implemented in Contiki-OS and simulated with COOJA, which incorporates a machine code instruction level emulator of the TELOSB sensor node, under all cases studied in the above analysis.

Both $T_{contact}$ and $T_{interval}$ follow a normal distribution with small deviation (a tenth of the mean). The sensed data is generated with a constant rate derived from ζ_{target}. The duty-cycle used by SNIP-AT and the scheduling plan used by SNIP-OPT are calculated based on the simulated environment and are incorporated into the codes.

For each experiment setup, these scheduling mechanisms have been simulated for two weeks and the average results for one epoch (one day) are plotted. Figure 7 and 8 show the results when Φ_{max} equals to $\frac{T_{epoch}}{1000}$ and $\frac{T_{epoch}}{100}$, respectively. These plots indicate that although there is a lot of variance in simulation results, the conclusions drawn from above analysis results are still correct.

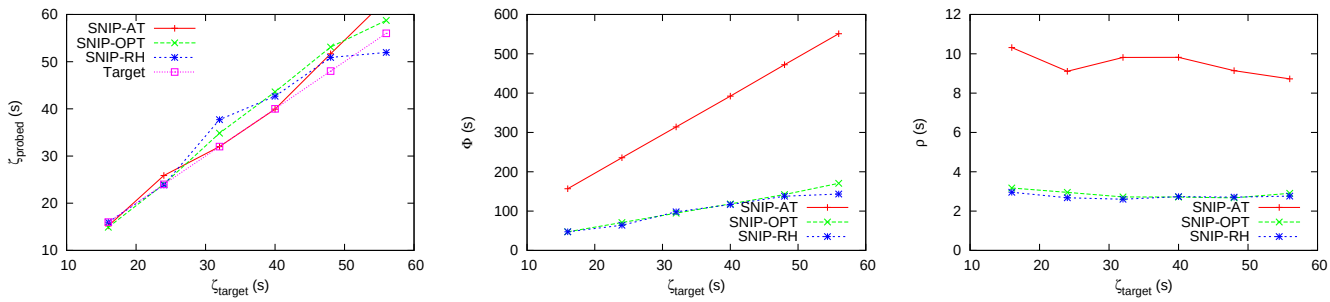
B. Discussion

In the above evaluations, we assume that a sensor node gets to know *Rush Hours* from engineers. When deploying in the



(a) The Probed Contact Capacity (b) The Overhead of Contact Probing (c) Cost for Per-unit Probed Contact Capacity

Fig. 7: Simulation Results of SNIP Scheduling Mechanisms when $\Phi_{max} = \frac{T_{epoch}}{1000}$



(a) The Probed Contact Capacity (b) The Overhead of Contact Probing (c) Cost for Per-unit Probed Contact Capacity

Fig. 8: Simulation Results of SNIP Scheduling Mechanisms when $\Phi_{max} = \frac{T_{epoch}}{100}$

real world, a sensor node may identify *Rush Hours* of the environment autonomously. To achieve this purpose, a sensor node can first run SNIP-AT for a while (a small number of epochs) to learn *Rush Hours*. It can then use SNIP-RH to improve the performance of contact probing. Considering that a sensor node only needs to learn the order of these time-slots' contact capacity, the learning phase could be short and the used duty-cycle could be very small. Hence, *Rush Hours* could be learned quickly and energy efficiently.

In some deployment environments, *Rush Hours* do have some seasonal differences [18]. In this case, a sensor node can simultaneously run SNIP-AT with a very very small duty-cycle so that it can continuously track the seasonal shift of *Rush Hours*. In the future, we will evaluate the performance of this kind of adaptive SNIP-RH in dynamic environments.

Contact arrival process also does not change with the time in the above evaluations. However, it tends to vary a lot in the real world. SNIP-AT seems to be promising in dynamic environment since SNIP is activated in all time and the environment can be learned in the course. However, to adjust its duty-cycle according to the environment and ζ_{target} (the target of probed contact capacity), SNIP-AT needs to accurately learn the contact arrival process of an epoch, both contact arrival frequency and contact length distribution. Considering that the expected applications of opportunistic data collection are low data rate, it is very hard to accurately learn the time-varying contact arrival process with just a few samples (the probed contacts). The smaller ζ_{target} is, the worse the situation is. Hence, a sensor node may react based on inaccurate information and this kind of adaptive SNIP-AT may not work well in dynamic environments. As discussed in subsection V, SNIP-OPT cannot work in dynamic environments. It is even harder for a sensor node to learn the contact arrival process for each time-slot. The sensor node may not be powerful enough to solve the optimization problem too.

As for SNIP-RH, we argue that it could work well. Although the amount of a time-slot's contact capacity varies a lot in different epoches, the mobility pattern is invariant in long term and *Rush Hours* will change very seldom. In the case that contact capacity during *Rush Hours* is high enough to support ζ_{target}, SNIP-RH is not sensitive to the variance of the amount of contact capacity during *Rush Hours*. When contact capacity becomes less, a sensor node just needs to run SNIP-RH for a longer time within *Rush Hours* and the necessary contacts still can be probed.

VIII. Related Work

Adaptive contact probing has been studied in Bluetooth-based opportunistic applications [19] and other delay-tolerant applications [20][21]. Through tuning the probing frequency, these proposals try to achieve better tradeoff between the probability of missing a contact and the energy consumed by contact probing. The characteristics of contact arrival process are first studied and the adaptive rules are designed

accordingly. For instance, the self-similarity of the contact arrival process among Bluetooth phones had been observed in [20] and the authors propose to increase the probing rate abruptly once a new contact is seen.

Instead of maintaining a low contact miss ratio, contact probing in opportunistic data collection tries to probe the necessary contacts energy efficiently, and the contact miss ratio can be large when ζ_{target} is small and contacts in the environment are abundant. Furthermore, the characteristics of contact arrival process cannot be utilized due to the following reasons. First, sensor nodes are deployed at different places and their contacts with mobile nodes may follow different patterns. Second, due to the low duty-cycle used by contact probing and the small memory of a sensor node, the node may not be able to learn the characteristics of its contact arrival process autonomously and timely.

Reinforcement Learning has also been exploited to decide the duty-cycle used by a sensor node for contact probing [18][22]. However, a sensor node can only explore a small number of states and strategies (duty-cycle values) due to its limited resources. With a small duty-cycle, it is also challenging to recognize the state and adopt the suitable strategy in a timely manner. Hence, a sensor node must also make decisions based on the inaccurate information learned with a small duty-cycle and the performance may be adversely affected.

When the duty-cycle must be low and the dynamic contact arrival process can not be accurately learned online, it should be necessary to exploit the long-term invariant of the environment, such as *Rush Hours* in the repeated mobility pattern of mobile nodes. This is why and how SNIP-RH is designed.

IX. Conclusion

Based on the observations that the intended applications of opportunistic data collection are delay-tolerant, mobile nodes tend to follow some repeated mobility patterns, and contacts are distributed unevenly in temporal, SNIP-RH is proposed in this paper to improve the performance of contact probing through exploiting *Rush Hours*, during which contacts arrive more frequently. The preliminary analysis and simulation results indicate that under typical roadside wireless sensor network scenarios, SNIP-RH can significantly reduce the energy consumed for probing the necessary contacts or significantly increase the probed contact capacity under a sensor node's energy budget for contact probing.

In future work, we will evaluate SNIP-RH plus SNIP-AT (with a very small duty-cycle) through trace-based simulations and/or experiments in real word. Furthermore, we will investigate the issues that arise when smart phones act as mobile nodes, such as incentives, user privacy, and data integrity that are encountered in participatory sensing [23] and/or online recommendation system [24].

Acknowledgements

This work is supported in part by the HEA PRTLI-IV NEMBES Grant. The authors thank the Cork Constraint Computation Centre (4C) at University College Cork for use of their computing cluster.

References

[1] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling a three-tier architecture for sparse sensor networks," in *IEEE SNPA Workshop*, 2003, pp. 30–41.

[2] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using predictable observer mobility for power efficient design of sensor networks," in *IPSN*, 2003, pp. 129–145.

[3] W. Wang, V. Srinivasan, and K.-C. Chua, "Using mobile relays to prolong the lifetime of wireless sensor networks," in *Mobicom*, 2005, pp. 270–283.

[4] B. Pasztor, M. Musolesi, and C. Mascolo, "Opportunistic mobile sensor data collection with scar," in *MASS*, 2007, pp. 1–12.

[5] A. A. Somasundara, A. Kansal, D. D. Jea, D. Estrin, and M. B. Srivastava, "Controllably mobile infrastructure for low energy embedded networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 8, pp. 958–973, August 2006.

[6] W. Liu, J. Wang, G. Xing, and L. Huang, "Throughput capacity of mobility-assisted data collection in wireless sensor networks," in *MASS*, 2009, pp. 70–79.

[7] M. Zhao and Y. Yang, "Bounded relay hop mobile data gathering in wireless sensor networks," in *MASS*, 2009, pp. 373–382.

[8] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone," in *USENIX Annual Technical Conference*, 2010.

[9] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling ultra-low power wireless research," in *IPSN*, 2005, pp. 364–369.

[10] X. Wu, K. N. Brown, and C. J. Sreenan, "SNIP: A sensor node-initiated probing mechanism for opportunistic data collection in sparse wireless sensor networks," in *IEEE INFOCOM Workshops*, 2011.

[11] F. Osterlind, N. Wirstrom, N. Tsiftes, N. Finne, T. Voigt, and A. Dunkels, "Strawman: Making sudden traffic surges graceful in low-power wireless networks," in *ACM HotEMNET Workshop*, 2010.

[12] "IEEE 802.15.4, Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)," IEEE, 2006.

[13] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki - a lightweight and flexible operating system for tiny networked sensors," in *LCN*, 2004, pp. 455–462.

[14] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level simulation in cooja," in *EWSN*, 2007.

[15] G. Anastasi, M. Conti, and M. D. Francesco, "An analytical study of reliable and energy-efficient data collection in sparse sensor networks with mobile relays," in *EWSN*, 2009, pp. 199–215.

[16] A. Cain, M. W. Burris, and R. M. Pendyala, "Impact of variable pricing on temporal distribution of travel demand," *Transportation Research Record*, no. 1747, pp. 36–43, 2001.

[17] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, pp. 779–782, 2008.

[18] V. Dyo and C. Mascolo, "A node discovery service for partially mobile sensor networks," in *MIDSENS*, 2007, pp. 13–18.

[19] C. Drula, "Fast and energy efficient neighbour discovery for opportunistic networking with bluetooth," Master Thesis, U. of Toronto, 2005.

[20] W. Wang, V. Srinivasan, and M. Motani, "Adaptive contact probing mechanisms for delay tolerant applications," in *Mobicom*, 2007, pp. 230–241.

[21] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, C. Mascolo, B. Pasztor, S. Scellato, N. Trigoni, R. Wohlers, and K. Yousef, "Evolution and sustainability of a wildlife monitoring sensor network," in *Sensys*, 2010, pp. 127–140.

[22] M. D. Francesco, K. U. Shah, M. J. Kumar, and G. Anastasi, "An adaptive strategy for energy-efficient data collection in sparse wireless sensor networks," in *EWSN*, 2010, pp. 322–337.

[23] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory sensing," in *World Sensor Web Workshop (in conjunction with Sensys)*, 2006.

[24] H. Yu, C. Shi, M. Kaminsky, P. B. Gibbons, and F. Xiao, "Dsybil: Optimal sybil-resistance for recommendation systems," in *IEEE Symposium on Security and Privacy*, 2009, pp. 283–298.