MobiRate: Mobility-Aware Rate Adaptation Using PHY Information for Backscatter Networks

Wei Gong*†, Si Chen†, Jiangchuan Liu†, Zhi Wang‡
*School of Computer Science and Technology, University of Science and Technology of China, China
†School of Computing Science, Simon Fraser University, Canada
‡Graduate School at Shenzhen, Tsinghua University, China
{gongweig, sca228, jcliu}@sfu.ca, wangzhi@sz.tsinghua.edu.cn

Abstract—**In the past few years, various backscatter nodes have been invented for many emerging mobile applications, such as sports analytics, interactive gaming, and mobile healthcare. Backscatter networks are expected to provide a high-throughput and stable communication platform for those interconnected mobile nodes. Yet, through experiments, we find state-of-the-art rate adaptation methods for backscatter networks share a fundamental limitation of accommodating the hardware diversity of nodes because the common mapping paradigm that chooses the optimal rate based on the radio signal strength indicator (RSSI) or the like is hardly adaptable to hardware-dependent RSSIs. To address this issue, we propose MobiRate (Mobility-aware Rate adaptation) that fully exploits the mobility hints from PHY information and the characteristics of backscatter systems. The key insight is that mobility-hints, like velocity and position, can greatly benefit rate selection and channel probing. Specifically, we introduce a novel velocity-based loss rate estimation method that dynamically re-weighs packets based on time and mobility. In addition, we design a mobility-assisted probing trigger and a new selective-probing mechanism, significantly saving probing time. As MobiRate is fully compatible with the current standard, it is prototyped using a COTS RFID reader and a variety of commercial tags. Our extensive experiments demonstrate that MobiRate achieves up to 3.8x throughput gain over the state-of-the-art methods across a wide range of mobility, channel conditions, and tag types.**

I. INTRODUCTION

Recently backscatter technologies have been dramatically revolutionizing traditional active-radio sensors as they can provide a battery-free, small form-factor, and cheap alternative while achieving comparable sensing capabilities. Numerous new backscatter nodes have been introduced and innovatively used in many mobile applications across sports, gaming, and healthcare [1], [2], [3], [4]. Meanwhile, backscatter communication has expanded into a range of wireless platforms, including Bluetooth, WiFi, LoRa [1], [5]. As such, in the near future, the number of backscatter nodes will see explosive growth in both quantity and variety. Backscatter networks that promise to deliver a high-throughput inter-connect platform for all kinds of backscatter nodes need to be prepared for such challenges.

Several advances, like rateless coding [6] and parallel decoding [7], have been made along this line and significantly improve backscatter throughput. Unfortunately, these designs are incompatible with existing standards and thus leave billions of deployed backscatter nodes, like RFID tags, behind. In this paper, we target at delivering high throughput for mobile backscatter networks (MBN) through a standard-compatible way. More specifically, we focus on optimizing rate adaptation at the link layer.

There are, however, three major challenges in designing an efficient link layer for MBN:

1) *Hardware-dependent Rate Selection*. The first seminal work for rate adaptation in MBN is Blink [8]. Later on, to combat channel diversity, CARA [9] is proposed. Yet both methods are mapping based. They adapt rates through a well-trained 2D map where the (key,value) pair is as ([loss rate, RSSI], optimal rate). Through experimental observations, we find, however, this trained map is *hardware-dependent*, which means the maps trained from different kinds of tags are quite diverse. Thus, those methods experience severe performance degradation when applying one well-trained map to other types of tags. The root cause comes from that the tags are from various vendors or designed for diverse usages, e.g., sub-sea pipe tag and heat resistant tag, and they tend to exhibit different SNR (signal noise ratio) responses due to different antenna lengths, circuit designs, and manufacture processes, even under the same channel condition.
2) *Inefficient Probing Trigger*. Probing cost is another critical factor in throughput optimization. Efficient probing triggers that can eliminate unnecessary probes are always favored. Nevertheless, this aspect is under-investigated in existing methods. Blink [8] starts probing when it detects mobility-pattern changes because it assumes the optimal rate changes with the tag's move; however, it is not always true. Let us suppose the tag keeps moving in a far-away zone, say 12 m from the reader. In such a zone where the path fading is dominating and multipath fading is negligible [8], the optimal rate would not change but keeps at the lowest. Thus, probes in such cases have no benefit but time is wasted. Blink's hands are tied for such cases as it does have a clue of the tag's location. CARA [9] feels the same say but has to probe regularly. Therefore, the room for considerable improvement over existing methods is apparent if fine-grained mobility-hints are

made available.

3) *Inaccurate Channel Estimation*. The loss-rate estimation is vital to all rate adaptation methods as it directly affects the accuracy of rate selection. In Blink [8], the loss-rate estimation is mainly designed for a single node. When multiple nodes are involved, differentiating the causes of a packet failure, like collisions from other tags or CRC errors due to poor channels, becomes extremely difficult. To address it, CARA [9] proposes a collision estimation method based on the slotted-ALOHA model to calibrate the observed loss rate. Yet, the estimate is still inaccurate due to the unpredictable capture effects in practical systems [10].

To tackle the above problems, we propose MobiRate, a Mobility-aware Rate adaptation method using PHY information for backscatter networks. It eliminates hardware dependency by introducing a throughput-based rate adaptation framework and extensively uses fine-grained mobility hints to optimize rate selection and probing efficiency. It achieves high-throughput while being compatible with the standard and commercial RFID readers. To do so, it mainly consists of three major components. First, it introduces a velocity-based loss-rate estimation that incorporates the impact of dynamic velocity on packet losses. Second, it presents a mobility-assisted probing trigger that uses mobility hints, including position and direction, to considerably reduce unnecessary probes. Third, it designs a selective probing method that novelly leverages the built-in system command to enable per-tag probing, eliminating potential MAC collisions.

We prototype MobiRate using a Thingmagic M6e reader and evaluate it with 12 types of tags across different vendors. We compare MobiRate against two state-of-the-art solutions, Blink and CARA, in a wide range of settings. Based on 120 traces across different mobility, channel conditions, and tag types, our evaluation results show that MobiRate achieves overall throughput gains of 3.8x over Blink and 2.9x over CARA on average. The gains come from transmission and probing. On average, MobiRate reduces probing cost significantly by 7.5x compared to Blink, and by 6.1x compared to CARA; MobiRate's rate selection module achieves throughput gains of 2.4x over Blink and 1.9x over CARA.

II. Motivation

A. Backscatter Primer

As shown in Figure 1, a typical backscatter system is composed of a reader and one or more tags. The reader first starts the communication by sending a continuous wave to the tag. The tag captures energy from this incoming wave to power itself and then transmits its signal by backscattering the same wave using ON-OFF keying. Specifically, the tag sends a bit '1' by changing its antenna impedance to reflect the wave, and a bit '0' by just keeping in the silent state. Most backscatter nodes and systems share similar design principles. In this paper, we focus on devices that are compatible with the C1G2 standard [11], one of most widely accepted RFID standards across various countries. Extensions to other backscatter systems, e.g, WiFi backscatter, are left for future research.

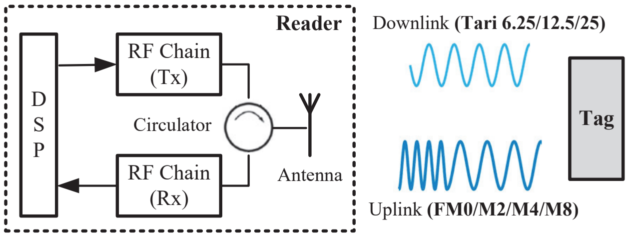


Fig. 1. Example of a backscatter system consisting of a reader and a tag. The downlink (Reader-to-Tag) rate can be tuned through Tari=6.25, 12.5, 25 μs, and the uplink (Tag-to-Reader) rate is decided by the encoding methods, FM0/M2/M4/M8 (assume the BLK is fixed). Both rates are set by the reader.

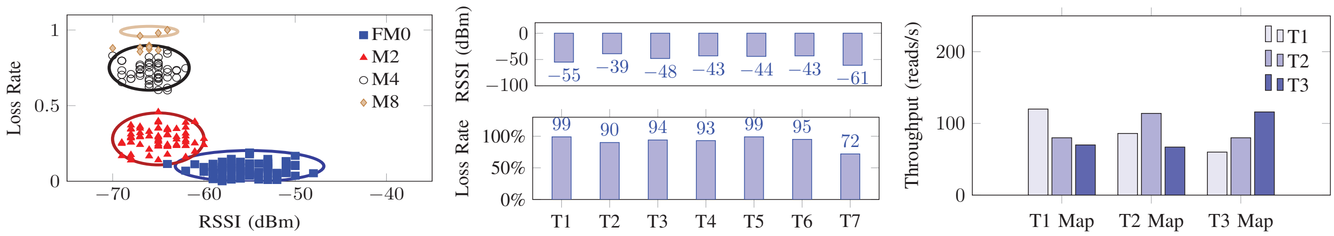
Different from active-radio links, a backscatter link includes a downlink (Reader-to-Tag) and a uplink (Tag-to-Reader). As the computation capability on tags is quite limited, the downlink adopts a simple amplitude modulation, Pulse Interval Encoding (PIE). In the standard, the length of a bit '0' is defined as Tari (Type A Reference Interval), and the length of a bit '1' is between 1.5Tari and 2Tari. As PIE is the only encoding method, the downlink rate is solely determined by Tari values. There are three options for Tari values, 6.25, 12.5, and 25 μs, corresponding to the maximum downlink rates, 160, 80 and 40 kbps if all the downlink bits are '0's. If more bit '1's are included, the practical rate would become lower. In contrast, the decoding ability of the reader is strong. The standard specifies four different encoding methods for the uplink, FM0, Miller2, Miller4, Miller8 [1], and the uplink rates are configured by these encoding methods and BLKs (Backscatter Link Frequency). For example, if BLK=640 kHz, the uplink rates are 640, 320, 160, and 80 kbps where FM0, M2, M4, and M8 are used, respectively. *Note that all the above configurable parameters, including Tari, uplink encoding, BLK, are completely controlled by the reader*. All the tag needs to do is to follow the reader's commands.

B. Observations

Rate adaptation is necessary and important for all wireless networks. An ideal rate adaptation method should timely choose the optimal rate that keeps pace with time-varying and location-dependent channels to maximize network throughput. The core of rate adaptation is how to choose the optimal rate for the current channel state.

In MBN, Blink [8] is the first work that uses both the loss rate and RSSI together as reliable channel measurements, because only RSSIs alone are not accurate due to multipath self-interference, a unique characteristic of MBN. Thus, a rate selection map is pre-trained to choose the optimal rate according to the probed loss rate and RSSI, as shown in Figure 2a. CARA [9] further optimizes throughput by exploiting spatial diversity and frequency diversity. While these two methods work well in their settings, we find that such mapping

[1] We use M2/M4/M8 to denote Miller2,Miller4, and Miller8.



(a) Rate selection map trained using a Higgs 3 tag. The BLK is fixed at 250 kHz for simplicity

(b) RSSIs and loss rates for 7 types of tags at the same location where FM0 is the optimal rate

(c) Throughput comparison using different maps

Fig. 2. Experiments showing the hardware diversity across 7 types of tags: **T1-High point piano tag; T2-SMARTRAC DogBone; T3-ImpinJ Monza 4D; T4-Alien Higgs 3; T5-Vulcan folded tag; T6-Vulcan Windshield tag; T7-confidex steelwave tag**.

based methods experience significant performance loss when different types of tags are involved.

First, we study the relationship between channel measurements and optimal rates across different types of tags. In this microbenmark, we have tested 7 kinds of tags from different vendors [2]. One group of results averaged across 20 traces is shown in Figure 2b. In this group, all the tags are tested at the same location one by one where FM0 is the optimal rate. Every setting is the same except the tag. The results show that *the channel measurements, including loss rates and RSSIs, differ significantly across diverse tags, even for almost the same channel quality,* indicated by the optimal rate (FM0) and the same location. Specifically, for T1 and T5 whose the loss rates are the same, the RSSI gap is -44-(-55)=11, which is huge. Moreover, the ranges of RSSIs and loss rates are considerably wide, namely, RSSIs∈[-61,-39], loss rates∈[0.72,0.99]. The root cause is different types of tags have diverse signal responses due to different antenna lengths, circuit designs, and manufacture processes. For example, the worst performed tag, T7, is a confidex steelwave tag designed for mounting on steel. It has an IP67 rating, which means it can be protected from dust and capable of withstanding water immersion between 15 cm and 1 meter for 30 minutes. We consistently observe the same phenomenon through similar experiments in different environments.

To further examine its impact on network throughput, we apply the maps learned from T1, T2, and T3 against each other. The results averaged across 50 different locations are shown in Figure 2c. We have two observations. First, the tag performs the best when tested using its own trained map. Specifically, T1, T2, and T3 achieve similar performances, which are 120, 114, and 116 reads/s when each is tested using its own map. Second, the throughput for each tag drops significantly when using the map trained from other tags. For example, the performance of T3 degrades to 60 reads/s using T1 trained map from 116 reads/s using its own map.

From the above experiments, we can conclude that the mapping based methods are hardly able to adapt to different hardware-dependent channel measurements and thus can hurt throughput seriously. Another drawback for mapping is the

[2]The specific types of tags are included in the caption of Figure 2

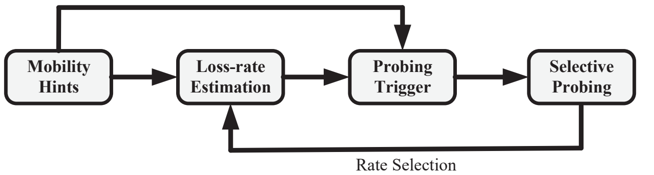


Fig. 3. Framework of MobiRate. It is throughput based and includes loss-rate estimation, probing trigger, and selective probing that helps choose the optimal rate.

accuracy of pre-trained map relies heavily on the classification performance. For better results, human-intervention is necessary for tuning parameters, e.g., the number of clusters. Moreover, those mapping based methods are susceptible to transient interference as later we show in Section V. In a word, we need a more flexible and robust method that can overcome the above mapping-related shortcomings.

III. MobiRate Design

A. Overview

To combat the drawbacks of mapping based methods, the successful experience from WiFi networks is using throughput based methods [12], [13] and some have been successfully deployed in commercial products, e.g., Minstrel [14] widely deployed in Linux OS, ATH9k [15] for Atheros WiFi cards. The basic idea of throughput based methods is that, instead of using signal strength related indicators that are susceptible to a variety of barely measurable factors, e.g., hardware diversity, multipath effects, interference, the *loss rate* deduced from the packet delivery history is a more reliable metric, and is more directly related to throughput across various environments. One prominent drawback of those throughput based methods [12], [13], [14], [15], however, is that while working very well with time-varying static channels, they are not aware of motions and thus cannot promptly respond to location-dependent channel changes. To achieve the best of two worlds, we design MobiRate, and its framework is shown in Figure 3. First, MobiRate uses a loss-rate estimation as its core. The key distinction of this estimation is that it is not fixed but velocity based, which comes from the common wisdom that as the node moves faster, the channel changes faster, and the optimal

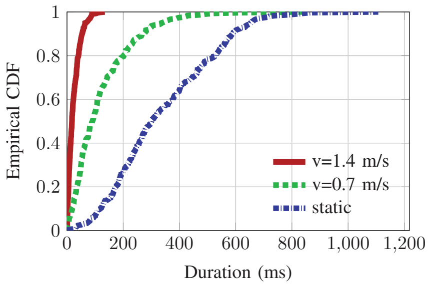


Fig. 4. ECDF of durations that a current rate remains optimal for three cases, v=0, 0.7, 1.4 m/s

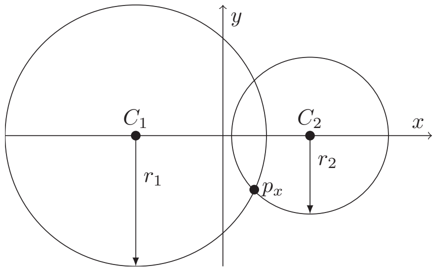


Fig. 5. Single-antenna localization using two anchor points

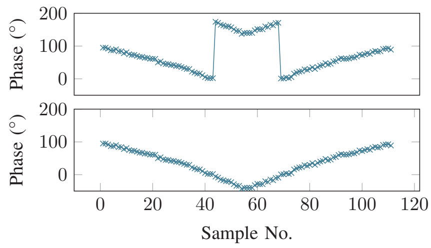


Fig. 6. Phase unwrapping. The tag moves outward 8 cm (corresponding to a phase-rotation, π) and then backward 8 cm. The up sub-figure is for wrapped phases and the down is for unwrapped phases

rate becomes stale more quickly. Second, it has a mobility-assisted probing trigger that uses velocity and position to cut unnecessary probing. Third, it introduces selective probing that novelly uses the built-in C1G2 commands to avoid MAC collisions and enables per-tag rate adaptation. Finally, the probing result will be compared with the performance of the current rate. The rate that has the smallest predicted average packet transmission time will be chosen. By exploiting the characteristics of backscatter links and mobility hints deduced from the unique PHY information of backscatter systems, we make MobiRate an efficient throughput-based rate adaptation.

B. Velocity-based Loss-Rate Estimation

Exponential Moving Average. In this section, we show how mobility hints can help rate adaptation. To begin with, we investigate how the optimal rate changes with mobility [3]. As shown in Figure 4, we observe that the optimal rate changes faster when the intensity of mobility, velocity, is higher. When the tag is static, it is possible to catch up with the change of the optimal rate using a large time window; when the tag is mobile, however, we have to increase the weight on the recent history instead of the stale one. To do so, MobiRate employs a packet (throughput) based rate adaptation method with a velocity-based smoothing factor. It does not require any training as in [8], [9]. Specifically, we maintain the current loss rate estimates as follows,

$$p_c' = \eta p_r + (1-\eta)p_c, \tag{1}$$

where p_c is the current loss rate at the current rate, p_r is the most recent packet delivery state, p_c' is the new current loss-rate estimate, η is the smoothing factor that is adjusted according to the velocity. p_c' is updated for every packet sent.

The above equation has been adopted by much prior work in WiFi networks [14], [15], but they cannot adapt to different mobility. Because η is the key here. Intuitively, a larger η would put less weight on p_c, the current loss rate, which is the previous history. Thus, a proper high η can ideally smooth out the effects of the past when the mobility is high, and a proper low η should work well when the mobility is minimal or even 0. MobiRate always chooses the best η according to the current velocity estimate [4].

Deducing Mobility Hints. To accurately set η, we need to estimate the velocity of the tag. Here, we introduce a novel *single-antenna* based tracking method that makes real-time position and velocity estimates available for rate adaptation. Prior work either inevitably introduces tracking delay on the order of at least seconds [17] or requires multiple antennas [18], which are not suitable for rate adaptation. Our solution is using phases, a PHY-hint, to measure distances. Phase measurements are supported in most commercial readers as specified in the LLRP standard [19]. For every successful read, the reader outputs a phase reading. This reported phase is an effective way to measure the distance between the reader and tag, denoted as R. From the electromagnetic theory, the relationship between R and the measured phase, θ, is as follows [17],

$$\theta = 2\pi\frac{2R}{\lambda} + \theta_D + \theta_R + N\pi + n, \tag{2}$$

where λ is the wavelength, θ_D, θ_R, are phase errors brought by tag and antenna diversity, and reflection characteristics, respectively, N is the integer ambiguity as the measured phase is with period π, n is the noise. If we know two phases at two locations, then the distance between the two can be approximated as

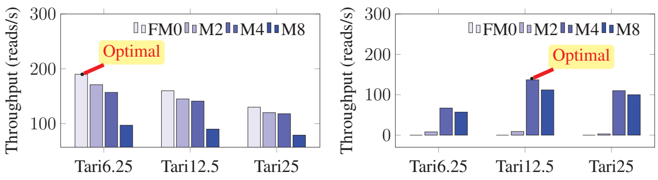
$$\Delta R \approx \frac{\lambda}{4\pi}\Delta\theta, \tag{3}$$

where θ_D, θ_R, and N are perfectly canceled out, and only noise n is left.

The insight of our tracking method is trading the number of anchor points with the number of antennas. Since we only have one antenna, we require the tag to pass two anchor points, whose coordinates are known in advance, so as to measure corresponding phases. When the tag moves to a new position, we can localize it by leveraging Equation 3 to obtain the distances between the new position and the anchor points. As

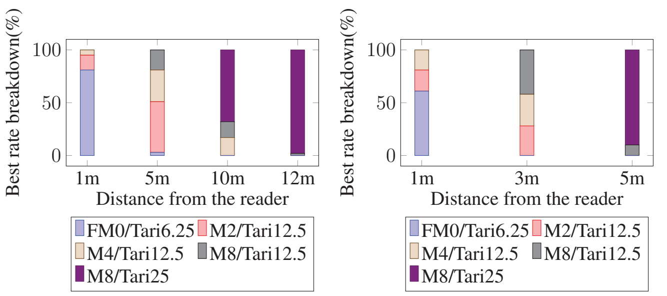
[3]The optimal rate is obtained from trace-based analysis as in [16].

[4]In MobiRate, the optimal η is set empirically: v = 0m/s, η=0.07; $0 < v < 0.8$ m/s, η=0.28; v$\geq$0.8 m/s, η=0.39.



(a) A good link (b) A bad link

Fig. 7. (a) is an example of a good channel, which favors the fastest uplink rate (FM0) and downlink rate (Tari=6.25); (b) is an example of a bad channel. Specifically, both FM0 and M2 encoding settings do not work, and the performance of Tari 6.25 is even worse than that of Tari 12.5, which suggests Tari 6.25 is an aggressive choice.



(a) optimal rate breakdown when the tag is within the antenna's beam (b) optimal rate breakdown when the tag is at the beam edge

Fig. 8. Optimal rate distribution at different distances. a) optimal rates are diverse within 10 m but become dominated by M8/Tari25 around 12 m; b) the similar phenomenon is observed around 5 m.

shown in Figure 5, C_1 and C_2 are the two anchor points, and p_x is the current position, which is unknown. Using Equation 3, we can estimate r_1 and r_2 reliably. Then by intersecting two circles, the position of p_x is found. Someone may note that there are two intersection points in the figure. To remove this ambiguity, we either simply add another anchor point and use the least square method to make an estimate or test if any intersection point is outside of the antenna's beam, e.g., the Laird panel antenna's beamwidth is about 65°.

There are several things worth noting here.

- Although N is cancelled in Equation 3, the actual phase still takes a fractional part (between 0 and π) and an integer part. Hence, phase unwrapping [20] is necessary for tracking this integer, as shown in Figure 6. Without such unwrapping, the distance calculated using Equation 3 would be always less than $\lambda/4$.
- To make our tracking robust, we employ an Extended Kalman Filter to obtain the final position and velocity estimates, because position and velocity errors can be bounded by the process modeling.
- The channel hopping that is mandatory in the C1G2 standard disrupts phase continuity. To solve this, we log the initial phases for all the channels at the anchor points. This way, we know the phase offsets between any two channels, which are constants. We can just pick any one channel as the base and calibrate all the phases from other channels to the base one.

C. Mobility-Assisted Probing Trigger

So far we have discussed how to adaptively compute the current loss rate based on the velocity, this part will present when we may need to adapt rates. MobiRate starts with the highest rate. When there are four successive failures, it will probe the next lower rate, which is similar to SampleRate [12]. When it stays at some rate for *a probing interval*, like a timer, it seeks the opportunity for the next higher rate.

Probing Direction. Unlike state-of-the-art methods that only considers the uplink rate, such as Blink and CARA, MobiRate takes both the uplink and downlink rates into account. To see why the uplink rate is very important in backscatter networks, we present two examples in Figure 7. As aforementioned, the downlink rate is decided by Tari values, and the uplink rate is controlled by BLKs and FM0/M2/M4/M8. For simplicity, in this part, we fix the BLK at 250 kHz. To examine the impact of different Tari values on the throughput, we keep the tag at a fixed place first. As shown in Figure 7a, currently the link quality is good, and faster rates have better throughput. The optimal rates in this case are Tari=6.25 for the downlink and FM0 for the uplink. From this case we know that in the case of good channels, we would miss the chance to increase the throughput if a conservative Tari is chosen. For example, when the uplink is with M2, the throughput of Tari=6.25 is 171 reads/s, but it drops to 120 reads/s when Tari=25. This observation motivates us to use the fastest rate for throughput maximization; however, this is not always the case. After we move the tag 1-meter away, we observe dramatically different behaviors. As shown in Figure 7b, this time the link is experiencing some difficulties because the throughputs of both FM0 and M2 encoding methods are almost 0. In this case, the optimal rates become Tari=12.5 for the downlink and M4 for the uplink. This tells us that too aggressive rates would not benefit but hurt the overall throughput when the channel is not good enough. The root cause for the above phenomenon is that *if the downlink rate is not properly set, the uplink would be discontinued*. Thus, MobiRate always probes in both directions whenever needing to test the next higher or lower rate. For example, let the current rate be M2/Tari12.5, if we need to probe the next higher rate, we probe both FM0/Tari12.5 and M2/Tari6.25. Similarly, for the next lower rate, we probe both M4/Tari12.5 and M2/Tari25. Finally, we compare the probed results in both directions with the current one and choose the one with the smallest predicted packet transmission time among the three as the new rate.

Lower-Rate Probing Control. Although a fixed interval is used for probing higher rates, we show that MobiRate can control the lower-rate probing times to reduce probing cost through mobility hints. There are two cases we take special care of.

First, when the tag is in the following dead zones, a) within the antenna's beam, about 12 m from the reader; b) at the beam's edge, about 5 m from the reader, we keep the rate at

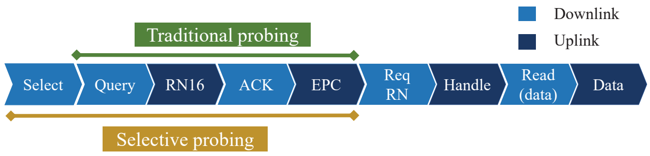


Fig. 9. Comparison of traditional and selective probing by the message breakdown. The additional *SELECT* command is only 46-bit long excluding the *MASK* field.

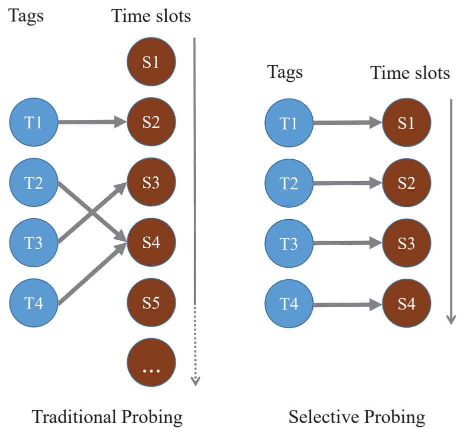


Fig. 10. Comparison of traditional and selective probing for multiple tags. In traditional probing, some slots that are collided (S4) or empty (S1&S5) are wasted; while selective probing's slots are fully used.

the lowest one, M8/Tari25, and cut all the probing triggers. This rule comes from the observations in Figure 8. We find that while the optimal rate distribution is quite diverse for most of the distances, M8/Tari25 is dominating in the above two zones. This is mainly because in both dead zones, path loss becomes the dominating factor impacting channel quality, and only the lowest possible rate can survive. Fortunately, our location-hint can help us determine whether the tag is in the dead zones.

Second, when the tag is moving towards the reader, we increase the number of successive failures to 8, which is twice of the original setting, in order to suppress a lower-rate probing. Because while the tag moves closer to the reader, the fast fading or multipath effects may cause transient failures. As observed from Figure 8, when the closer the tag is to the reader, the higher percentage (probability) the faster rates become optimal. For example, when the tag is 1 m away, FM0/Tari6.25 takes 81% share in Figure 8a. Thus, to keep from being trapped into lower rates and save probing cost, we choose to withstand more transient interferences by increasing the limit of tolerable successive failures.

D. Selective Probing

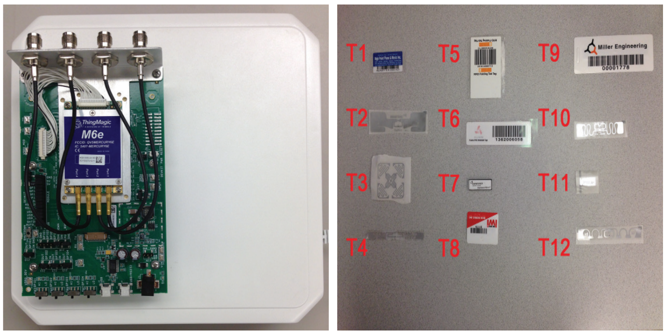
The previous part solves the problem when to probe. Now, we come to the last question how to probe and obtain accurate probing results. Following the C1G2 standard and commercial readers' implementation, the minimum unit that can be used as a probe is the identification process, where the downlink includes *Query* and *ACK* commands, and the uplink involves *RN16* and *EPC* replies, as shown in Figure 9. When the reader receives a correctly decoded *EPC*, the probe is marked as positive, otherwise negative. This mode is used by Blink [8] but has a main drawback where multiple tags may collide in some slots, resulting in 'fake' packet loss. To differentiate packet losses that are caused by poor channels and tag collisions, CARA [9] estimates the collision probability and use it to calibrate the observed loss rate. Yet, this method is still not very accurate due to the unpredictable capture effects [10].

Different from prior work, we observe the opportunity of using the C1G2 built-in *SELECT* command to overcome the above difficulty and achieve accurate channel estimation. The *SELECT* command is originally designed for choosing a tag population for inventory. One or more tags can be selected by the reader according to user-specified criteria, which is analogous to selecting records from a database. In a *SELECT* command, the reader can specify the *Memory Bank*, the associated starting address and length, and a *MASK*. In MobiRate, we intend to select a single tag for probing. Therefore, we can specify a *SELECT* command by setting the memory bank as *EPC*, starting address as 0, length as 96, and *MASK* as the wanted tags EPC. This way, *only the tag that matches the mask would reply*, eliminating tag collisions. Someone may note that this method requires the EPC before probing. Actually, this is not a problem. As our goal is to maximize the throughput for reading RFID-sensor data, we should already know which RFID-sensor we would like to collect data from in advance. Even when we may not know the sensor's ID, as shown in Figure 9, we can use an identification process to collect the ID.

Note that here we use the extra cost of *SELECT* commands to enable per-tag probing. A *SELECT* command is only about 45-bit long (excluding the *MASK*). Thus, such cost is negligible compared to the MAC inefficiency shown in Figure 10. More specifically, the selective-probing overhead grows linearly with the number of tags whereas traditional solutions increase quadratically. Another advantage of selective-probing is that we can achieve per-tag reading using the same mechanism and solve the problem of assigning different rates for different tags [9] as opposed to the traditional mode where all the tags have to use the same rate, further optimizing the network throughput.

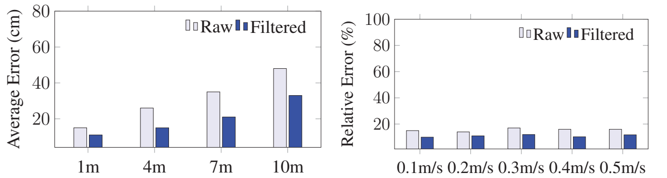
IV. Implementation

We mainly use a Thingmagic M6e reader in our implementation, as shown in Figure 11a. It is fully compatible with the C1G2 standard. The rate adaptation programs are written in C# using Mercury API SDK v1.29.3. Yet, we have the same problem as [8] does because most commercial readers share some common API constraints. For example, for Blink and CARA, their implementation can only set the data rate in the beginning of a round; the channel switch takes at least 30 ms and so is the minimum probing time. These issues would be largely mitigated when the future readers expose more flexible



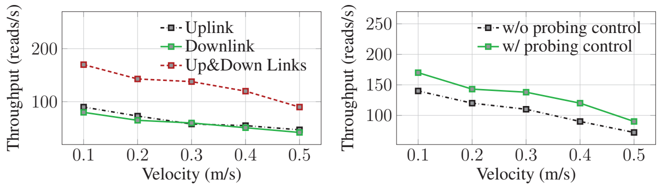
(a) An M6e reader and a panel antenna (b) 12 types of tags

Fig. 11. Main hardware in our testbed. T1-T7 are specified in Figure 2; T8: Iwi Iron works tag; T9: Miller Engineering tag; T10: SMARTRAC belt tag; T11-SMARTRAC trap tag; T12: SMARTRAC ShortDipole tag.



(a) Localization errors at different distances from the reader (b) Velocity relative errors at different mobility

Fig. 12. Comparison of raw estimates and Kalman filtered results. (a) The filtered results are always better than raw estimates and the localization errors grow with the distance due to attenuated RSSIs; b) The filtered velocity estimates are also better than raw results, while the velocity error is well under control across different mobility.



(a) Comparison of adapting uplink rates, downlink rates, and the both. (b) Comparison of cases that are with and without our lower-rate probing control mechanism.

Fig. 13. (a) shows that adapting both rates is necessary as adapting a single link can only achieve around half of the optimal throughput; (b) shows that the lower-rate probing control mechanism helps optimize throughput about 25% across different velocities.

APIs. Thus, for fair play, we use trace-driven experiments when the above limitations happen during comparisons.

We include 12 types of tags, as shown in Figure 11b, such as the Alien Higgs 3 tag (T4), the metal resistant tag (T7), the SMARTRAC ShortDipole tag (T12). As the sensor data does not have any impact on the transmission, we just simply write random bits into the tags.

Our parameters are set as follows. As the M6e reader only supports full options at BLK=250 kHz, where available Tari values are 6.25, 12.5, and 25, and the uplink encoding methods are FM0/M2/M4/M8. Thus, all our tests are using the same BLK. For probing, we assume all the tags' IDs are known and use them for selective probing. Each ID is 96-bit long. The power of reader is 30 dBm, fixed.

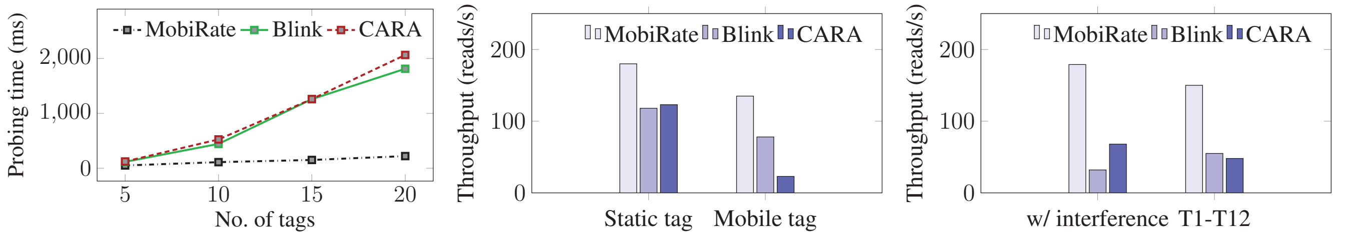
We mainly compare MobiRate with Blink [8] and CARA [9], two state-of-the-art MBN methods. Other WiFi based rate adaptation methods, e.g., are not included since no clear instructions about how to migrate them to MBN are available and simple modification is not recommended as many features of backscatter networks are unique, e.g., data rates for the uplink and downlink, the C1G2 standard, and PHY-hints.

V. Evaluation

Mobility Hints. To begin with, we investigate how our tracking method works. First, we put the tag at different distances from the reader within the beam, namely, 1 m, 4 m, 7 m, and 10 m. As shown in Figure 12a, the average errors across different distances of the raw and Kalman filtered estimates are 31 cm and 20 cm, which are adequate for the use of rate adaptation. Note that we achieve this only using a single antenna and the processing is real-time. To put it in the context, the 20 cm average error enables us to distinguish tags on the scale of an A4 paper. We also observe that this error is initially around 11 cm and increases to about 33 cm at 10 meters. Such increase is mainly due to the significantly attenuated signal strength at further distances. We also examine the quality of velocity estimates across different mobility. Here, we put the tag on a programmable robot, *iRobot Create 2*, whose velocity can be controlled between 0-0.5 m/s by programming. A raspberry Pi 3 is used to relay the commands from the laptop to the robot. Results are shown in Figure 12b. We observe that the velocity relative error is well under control across different velocities because all the filtered velocity estimates have errors below 12%. From the both figures, we confirm that our choice of the Kalman filter is rewarded because it makes both location and velocity estimates more robust and accurate.

Probing Trigger. Next, we check the impact of our probing trigger module. We primarily investigate two factors: 1) adapting both rates for the uplink and downlink; 2) the effect of the lower-rate probing control mechanism. We compare the performances of three cases: adapting only the uplink, only the downlink, and the both. Results are shown in Figure 13a. We observe that the throughput of adapting both links is way better than the other two cases. For example, when the velocity is 0.3 m/s, adapting only the uplink has a throughput of 58 reads/s while adapting only the downlink has 60 reads/s throughput. In contrast, if both are adapted, a throughput of 138 reads/s is achieved. Similar trends can be observed across different velocities. This validates our former claim that both rates should be carefully taken care of as the uplink and downlink are a unified backscatter link.

Moreover, we study the impact of the lower-rate probing control mechanism. As shown in Figure 13a, the solution with the control mechanism is consistently better the one without it. For instance, when the velocity is 0.4 m/s, the solution without the control mechanism has a throughput of 90 reads/s

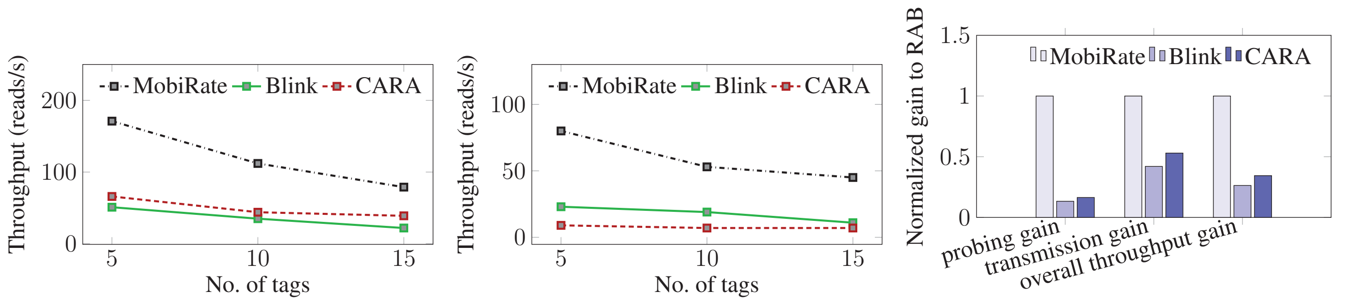


(a) Comparison of different methods in terms of probing time

(b) Impact of probing for a single tag

(c) Comparison of different methods under interferences and hardware diversity

Fig. 14. Investigating the impact of probing in detail. a) shows that the probing costs of Blink and CARA are way larger than that of MobiRate; b) show MobiRate performs the best with the help of novel probing strategy for both statics and mobile cases; c) shows our mobility-assisted probing method and loss rate estimation bring more throughput gains than Blink and CARA do under interferences and hardware diversity scenarios.



(a) Static cases

(b) Mobile cases

(c) Comparison of gain breakdown

Fig. 15. Overall throughput comparison with various settings, including mobility, time, places, tag types and populations.

while the one with it has a 120 reads/s throughput, which is a 1.34x gain. Overall, the lower-rate probing control mechanism proves its effectiveness by achieving a 1.25x throughput gain on average.

Probing Cost. Then, we examine the probing cost in detail. First, we check probing time by comparing MobiRate against Blink and CARA with different tag populations. Figure 14a shows that the probing cost of Blink and CARA grows way faster with the number of tags than that of MobiRate. Specifically, the probing costs of Blink and CARA are 1823 ms and 2051 ms, corresponding to 8.2x and 9.4x more than that of MobiRate when the number of tags is 20. This is primarily due to the selective probing that probes tags linearly without MAC collisions. Blink and CARA do not have such support.

To study the impact of our probing method that excludes selective probing on the overall throughput, we conduct a comparison using only a single tag under both static and mobile scenarios. In this mobile scenario, we put the tag on the human body and let him do random walking at the normal walking velocity, about 1 m/s. Figure 14b shows that the throughput of MobiRate is considerably better than those of Blink and CARA. Note that while there is no much difference between Blink and CARA in the static case, CARA performs worse than Blink does in the mobile case because it cannot detect mobility changes.

We further examine how these three methods perform with interferences and diverse hardware. We create interferences by employing a USRP reader that intermittently sends out queries around the M6e reader. As demonstrated in Figure 14c, both Blink and CARA perform badly whereas MobiRate stays robust with interferences. In addition, the performance of MobiRate is substantially better than those of Blink and CARA when diverse tags are tested (T1-T12). The robustness of MobiRate comes primarily from the throughput based rate adaption that is universal to different hardware, and our mobility-assisted probing trigger that cuts unnecessary probes.

Overall Performance. Finally, we investigate the overall performance of the whole system. First, we study the static case where all tags are placed randomly. Figure 15a shows that when there are 5 tags, the throughput of MobiRate is 3.4x and 2.0x better than Blink and CARA, respectively. The same trend can be observed across different tag numbers.

The results of mobile cases are shown in Figure 15b. We observe that all the three systems are negatively impacted by mobility, but MobiRate stays strong whereas the other two surrender. Particularly, when the number of tags is 15, MobiRate achieves 4.1x and 6.4x throughput gains over Blink and CARA. CARA is the worst because it lacks mobility hints.

Moreover, we perform over 120 tests across a wide range of settings, including mobility, channel conditions, tag types, and tag numbers. We vary the velocity of tags from 0 to 1 m/s and tag populations from 1 to 15. The traces are collected during a 10-day period at two different places, an indoor office and a lobby. The overall gains and their breakdown are reported in Figure 15c. We find that MobiRate achieves overall throughput gains of 3.8x over Blink and 2.9x over CARA on average. By breaking down these gains, we find that MobiRate reduces probing cost significantly by 7.5x compared to Blink, and by 6.1x compared to CARA. The main source of this probing gain is the design of mobility-assisted probing trigger and selective probing. Meanwhile, MobiRate is 2.4x and 1.9x better than Blink and CARA in data transmission. This gain is mainly brought by our throughput based rate selection that adapts rates for both the uplink and downlink.

VI. Related Work

Rate adaptation for wireless networks can be broadly classified into two categories:

Mapping based: The mapping based methods assume that the best rate can be chosen based on SNR related metrics, e.g., BER, RSSI. Many excellent solutions have been proposed. FARA [21] and novelly allows each OFDM sub-carrier to pick a modulation and a code rate that match its SNR. Its rate selection is built on an SNR-rate mapping table. ESNR [16] presents a delivery model that leverages channel state information to combat frequency selective fading. It introduces a new metric, effective SNR, which is used to look up the optimal rate in the table. Blink [8] and CARA [9] share the similar idea and further exploit the RSSI and loss rate together to mitigating the exclusive backscatter phenomenon, multipath self-interference. These methods, however, are not universal, because the ground truth of SNRs is very hard to obtain due to hardware diversity and interference [22].

Throughput based: In contrast, throughput based methods are universal and robust, and thus are favored by many commercial-product implementations, e.g., Minstrel [14] and Ath9k [15]. The most famous solution in this class is SampleRate [12]. They work well in static environments but cannot make timely responses to location changes because of unawareness of mobility states. To solve this, a mobility-assisted solution has been proposed in [23]. Nevertheless, it can only obtain coarse-grained mobility hints and does not consider many backscatter unique characteristics, e.g., adapting rates for both the uplink and downlink, MAC collisions in probing, and location-based probing trigger, as MobiRate does.

In summary, inspired by the prior works, our work's novelty lies in obtaining real-time fine-grained mobility-hints and using these hints to design an efficient and C1G2-compatible solution exclusively for backscatter networks.

VII. Conclusion and Future Work

We have presented MobiRate, a mobility-aware rate adaptation method for backscatter networks. Through a novel single-antenna tracking method, accurate mobility hints have been obtained. By leveraging such mobility-hints, we have shown that it can achieve significant performance gains in both rate selection and probing cost over the two state-of-the-art works, Blink and CARA. A prototype has been demonstrated using a variety of different types of tags. We have verified effectiveness through extensive experiments.

Acknowledgment

We would like to thank the anonymous reviewers for valuable and insightful comments. This work was supported by a Canada Technology Demonstration Program (TDP) grant, a Canada NSERC Discovery Grant, and an NSERC E.W.R. Steacie Memorial Fellowship, and NSFC under Grant No. 61472268.

References

[1] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J.R. Smith, and S. Gollakota. LoRa Backscatter: Enabling The Vision of Ubiquitous Connectivity. In *Proc. of ACM UBICOMP*, 2017.

[2] T. Li, S. Wu, S. Chen, and M. Yang. Energy efficient algorithms for the RFID estimation problem. In *Proc. of IEEE INFOCOM*, 2010.

[3] X. Liu, B. Xiao, K. Li, J. Wu, A.X. Liu, H. Qi, and X. Xie. RFID cardinality estimation with blocker tags. In *Proc. of IEEE INFOCOM*, 2015.

[4] J. Han, H. Ding, C. Qian, D. Ma, W. Xi, Z. Wang, Z. Jiang, and L. Shangguan. CBID: A Customer Behavior Identification System Using Passive Tags. In *Proc. of IEEE ICNP*, 2014.

[5] B. Kellogg, V. Talla, S. Gollakota, and J. R. Smith. Passive wi-fi: Bringing low power to wi-fi transmissions. In *Proc. of USENIX NSDI*, 2016.

[6] J. Wang, H. Hassanieh, D. Katabi, and P. Indyk. Efficient and Reliable Low-Power Backscatter Networks. In *Proc. of ACM SIGCOMM*, 2012.

[7] P. Hu, P. Zhang, and D. Ganesan. Laissez-faire: Fully AsymmetricBackscatter Communication. In *Proc. of ACM SIGCOMM*, 2015.

[8] P. Zhang, J. Gummeson, and D. Ganesan. Blink: A high throughput link layer for backscatter communication. In *Proc. of ACM MobiSys*, 2012.

[9] W. Gong, H. Liu, K. Liu, Q. Ma, and Y. Liu. Exploiting Channel Diversity for Rate Adaptation in Backscatter Communication Networks. In *Proc. of IEEE INFOCOM*, 2016.

[10] M. Buettner and D. Wetherall. An empirical study of UHF RFID performance. In *Proc. of ACM MobiCom*, 2008.

[11] EPC C1G2 Standard. http://www.gs1.org/epcrfid/epc-rfid-uhf-air-interface-protocol/2-0-1, 2017.

[12] J. C. Bicket. *Bit-rate selection in wireless networks*. PhD thesis, Massachusetts Institute of Technology, 2005.

[13] I. Pefkianakis, Y. Hu, S. H. Wong, H. Yang, and S. Lu. MIMO rate adaptation in 802.11 n wireless networks. In *Proc. of ACM MOBICOM*, 2010.

[14] Minstrel. https://wireless.wiki.kernel.org/en/developers/documentation/mac80211/ratecontrol/minstrel, 2017.

[15] ATH9k. https://wireless.wiki.kernel.org/en/users/drivers/ath9k, 2017.

[16] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Predictable 802.11 packet delivery from wireless channel measurements. In *Proc. of ACM SIGCOMM*, 2010.

[17] L. Yang, Y. Chen, X. Li, C. Xiao, M. Li, and Y. Liu. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. In *Proc. of ACM MobiCom*, 2014.

[18] J. Wang, D. Vasisht, and D. Katabi. RF-IDraw: Virtual Touch Screen in the Air Using RF Signals. In *Proc. of ACM SIGCOMM*, 2014.

[19] Low Level Reader Protocol. http://www.gs1.org/epcrfid/epc-rfid-llrp/1-1-0, 2017.

[20] R.M. Goldstein, H.A. Zebker, and C.L. Werner. Satellite radar interferometry: Two-dimensional phase unwrapping. *Radio science*, 23(4):713–720, 1988.

[21] H. Rahul, F. Edalat, D. Katabi, and C. G. Sodini. Frequency-aware rate adaptation and MAC protocols. In *Proc. of ACM MobiCom*, 2009.

[22] J. Zhang, K. Tan, J. Zhao, H. Wu, and Y. Zhang. A practical SNR-guided rate adaptation. In *Proc. of IEEE INFOCOM*, 2008.

[23] L. Sun, S. Sen, and D. Koutsonikolas. Bringing mobility-awareness to WLANs using PHY layer information. In *Proc. of ACM CONEXT*, 2014.