Mobile Backscatter Communication for the Battery-less Internet of Things

Weining Song
Uppsala University
Sweden
weining.song@angstrom.uu.se

Thiemo Voigt
Uppsala University, RISE
Sweden
thiemo.voigt@angstrom.uu.se

Stefanos Kaxiras
Uppsala University
Sweden
stefanos.kaxiras@it.uu.se

Yuan Yao
Uppsala University
Sweden
yuan.yao@it.uu.se

Luca Mottola
Politecnico di Milano, RISE, Uppsala University
Italy and Sweden
luca.mottola@angstrom.uu.se

***Abstract*—We enable backscatter communication in the *battery-less mobile* Internet of Things (IoT). Backscatter communication is extensively studied in static settings. Existing designs are, however, fundamentally mismatched with mobility and time-varying energy patterns. Channel conditions rapidly fluctuate, impacting the achievable data rates and thus transmission costs. Energy availability varies unpredictably, possibly forcing devices to remain quiescent to recharge energy buffers. The two issues compound each other: while recharging, a battery-less mobile IoT device may miss more favorable channel conditions. We design a lightweight decision system that dynamically determines when to transmit by checking short-term trends in signal strength, while using Non-volatile Memory (NVM) to retain packets in unfavorable channel conditions and across energy failures. Using a prototype we built and real-world mobility and power traces, we compare our design against a rate-adaptive baseline that only considers the instantaneous channel conditions. Experimental results show that our system improves throughput by up to 5.16 × while reducing transmission energy consumption by up to 47.3%, with only 0.23% – 7.3% additional energy overhead.**

***Index Terms*—Internet of Things (IoT); backscatter communication; energy harvesting; non-volatile memory (NVM)**

I. Introduction

Backscatter communication allows battery-less Internet of Things (IoT) devices to transmit data with minimal energy consumption by modulating information onto an ambient carrier signal [1], [2]. Most existing backscatter systems are designed under the assumption of static environments, where channel conditions remain stable [3]–[5]. However, many IoT applications involve mobile energy-harvesting devices operating in dynamic environments [6]–[11], where both channel conditions and harvested energy vary rapidly over time. The mismatch between design assumptions and dynamic conditions fundamentally challenges existing backscatter systems.

Mobility in backscatter communications. Device mobility leads to continuously varying channel conditions between the backscatter tag, the carrier emitter, and the receiver [12]. Channel conditions degrade with increasing distance from the carrier emitter, requiring the system to transmit at a lower data rate to retain packet reliability [13]. Fig. 1 provides

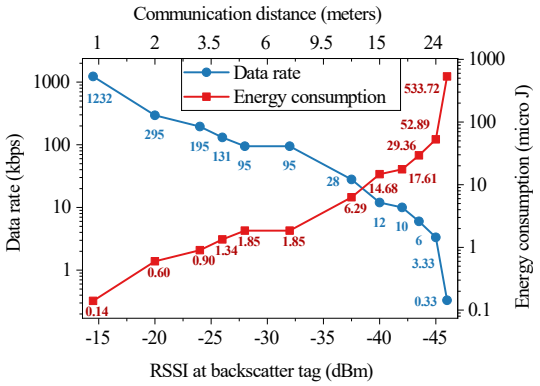


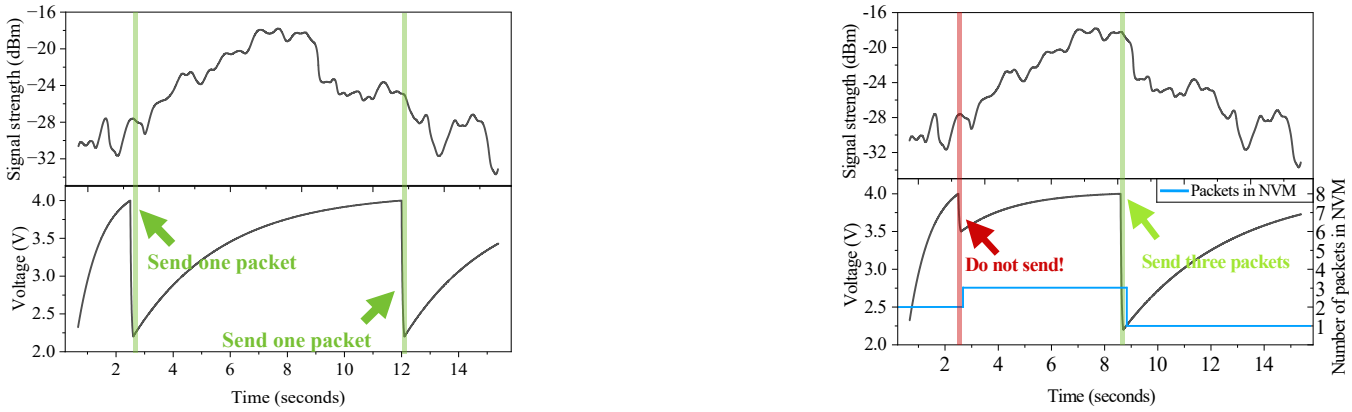
Fig. 1: Impact of channel quality and communication distance on transmission rate and energy cost [13].

quantitative evidence from experimental measurements, obtained by replicating the setup of prior work [13]. As the distance increases, the achievable data rate decreases, leading to longer transmission times and extended microcontroller active periods, and thus higher per-packet energy consumption.

At the same time, harvested energy is erratic and scarce, possibly causing frequent energy failures that force the device to stay off while recharging energy buffers. When the device is off, recharging and yet still mobile, possibly favorable channel conditions are missed [7], [14].

When mobility and energy harvesting coexist, both channel conditions and available energy vary over time and influence each other, leading to a distinct design problem where transmission decisions depend on future channel and energy dynamics. This limits the effectiveness of pre-tuned transmission strategies and motivates the need for runtime adaptation.

Mobile transmission control. Under time-varying channel conditions and energy availability, the system should intelligently decide *when it is the right time to transmit*. Given sufficient energy, postponing transmission under poor channel conditions may reduce energy consumption if better conditions and higher data rates become available later. However, future



(a) Rate-adaptive backscatter system.

(b) Mobile backscatter system.

Fig. 2: Example comparing rate-adaptive and mobile backscatter systems under identical signal and power traces.

channel and energy availability are uncertain, and such decisions must be made on the scarce compute resources available on mobile backscatter IoT devices with minimal overhead.

Unlike existing designs that base transmission decisions on instantaneous signal strength [13], our system monitors how the received signal strength indicator (RSSI) at the backscatter tag changes over time. When channel conditions improve and sufficient energy is available, the system provides more opportunities for packet transmissions; when conditions worsen, transmissions are throttled down to conserve energy for future opportunities. We use modern Non-volatile Memory (NVM) to persist unsent packets across energy failures and periods of poor channel conditions, allowing transmission to resume once energy and channel conditions improve. This design enables the device to adapt to time-varying channel and energy conditions, ultimately improving overall throughput.

We prototype the system on an MSP430FR5969 MCU and implement three applications: agricultural environment measurement (AEM), activity recognition (AR), and cold-chain equipment monitoring (CEM). We use a trace-driven simulator to evaluate the system against a rate-adaptive baseline [13]. Using real-world signal traces and power traces [15], experimental results show that our approach achieves up to 5.16 × the long-term throughput of the baseline. This improvement is enabled by up to 47.3% reduction in transmission energy consumption compared to the baseline, while introducing only 0.23% – 7.3% additional energy overhead.

II. Background and Related Work

Intermittent computing. Battery-less IoT devices rely solely on ambient energy [14], leading to frequent energy failures and intermittent execution due to erratic energy availability [7]. To maintain forward progress, these devices use NVM to persist program state across energy failures and restore it after recharging. Some solutions use checkpointing to save the intermediate program state to NVM [16]–[18]. This approach saves program state at specific points and restores it once sufficient energy is available. Existing techniques differ in checkpoint placement and whether their placement is determined at compile-time or run-time. These techniques focus on ensuring correct execution across power failures rather than proactively regulating communication decisions. In contrast, our work uses NVM to persist unsent packets and control state, allowing the system to proactively store packets even when sufficient energy is available and transmit them later under more favorable channel conditions.

Backscatter communication. By reflecting or absorbing radio frequency (RF) signals rather than generating its own carrier, backscatter communications enable data transmission with μW power consumption [1], [3]–[5], [19], [20].

Xu et al. [21] present a battery-free backscatter system that predicts intermittent WiFi excitation duration and schedules charging and transmission accordingly. In contrast, we consider device mobility that causes continuously varying channel conditions and harvested energy rather than a stable carrier with unpredictable duration. Muratkar et al. [12] study mobility in ambient backscatter at the physical layer through channel modeling and detection analysis. In contrast, our work focuses on system-level transmission control under mobility and energy harvesting. FreeBack [13] adapts its transmission rate to channel conditions estimated by the tag from the incoming carrier signal. Unlike our work, FreeBack uses instantaneous signal strength only for rate adaptation, without considering energy availability or proactively regulating transmissions.

III. System Design

A. Intuition

Rate-adaptive backscatter systems, such as FreeBack [13], adjust the transmission rate based on instantaneous channel conditions. In this work, the carrier emitter and receiver are co-located, which simplifies channel estimation based on the received signal strength of the carrier at the tag. We replicate the experimental setup [13] to characterize the relationship between RSSI, communication distance, data rate, and energy consumption. The results in Fig. 1 show that to transmit a 256-bit packet on the MSP430FR5969 MCU, the data rate drops from 1232 kbps at -14 dBm to 0.33 kbps at -46 dBm, while energy consumption increases from 0.14 μJ to 533.72 μJ, over 3000 × higher. Thus, transmitting under poor channel

conditions can waste scarce harvested energy. Furthermore, the rate-adaptive system only considers instantaneous channel conditions, ignoring cases where the tag moves out of communication range and where energy failures make transmission impossible in mobility and energy-harvesting scenarios.

In such scenarios, both future energy input and channel conditions are unknown, leading to a fundamental issue: *when it is the right time to transmit*. Fig. 2a illustrates the issue. At 2.5 s, the capacitor is fully charged, and the system transmits at an RSSI of -28 dBm. A single packet transmission consumes significant energy, causing the capacitor voltage to drop from 4.0 V to 2.2 V. This results in a long recharging period from 2.5 s to 12 s, missing the strong signal window (7 s – 9 s). In Fig. 2b, the application still generates a packet at 2.5 s, but stores it in NVM together with packets generated under unfavorable channel conditions. This preserves energy and allows the tag to resume transmission sooner as the signal improves. When the capacitor is fully charged again, the corresponding RSSI is -19 dBm. Under the same energy budget, with the capacitor voltage dropping from 4.0 V to 2.2 V, our system transmits three packets in a burst: two buffered in NVM during earlier weak channel conditions, and one newly generated packet. By aligning transmission with favorable channel conditions, we achieve significantly higher throughput with the same energy budget.

Ultimately, the key challenge is to decide when and how aggressively the system should transmit to maximize throughput under time-varying channel and energy conditions.

B. Transmission Control

Our design is loosely inspired by TCP congestion control, where network capacity is estimated to regulate transmissions [22]. Rather than estimating network capacity, our system infers transmission opportunities from channel trends and energy conditions and adjusts transmissions accordingly.

RSSI trend recognition. To detect channel trends with minimal state and computational overhead, the system maintains an exponential moving average (EMA) of the RSSI signal at the tag as a smoothed reference of recent channel conditions. The EMA is updated as follows:

$$S_n = \alpha_n R_n + (1 - \alpha_n) S_{n-1}, \tag{1}$$

where R_n is the current RSSI sample, S_n is the smoothed estimate, and S_{n-1} is the previous estimate. With energy harvesting, device wake-ups and RSSI sampling occur at irregular intervals. To handle this irregularity, the smoothing factor α_n is adapted based on the interval Δt between consecutive samples, measured by the real-time clock (RTC):

$$\alpha_n = 1 - e^{-\Delta t/\tau}, \tag{2}$$

where τ controls the effective memory of the filter and is set to 5 s in all experiments. This allows the EMA to naturally adjust to irregular sampling while still providing a stable reference.

The instantaneous RSSI sample R_n reflects the channel quality from the emitter to the tag. In our setup, the carrier emitter and receiver are co-located, allowing this measurement to be used to estimate the overall backscatter channel quality. We obtain the current channel trend by subtracting the EMA reference, S_n, from the instantaneous RSSI sample:

$$T_n = R_n - S_n, \tag{3}$$

where a positive T_n indicates increasing channel quality, while a negative value indicates decreasing channel quality. This compares the current RSSI against the EMA reference to capture short-term channel dynamics.

This approach is computationally inexpensive and requires a small memory footprint. It provides sufficient trend information to guide transmission decisions without relying on complex prediction models for future signal estimation.

Transmission budget. The system maintains a non-negative aggressiveness counter that determines the number of packets scheduled for transmission during each active period. The counter is initialized to zero and updated at the beginning of each active period based on the detected RSSI trend and the observed charging duration from the RTC.

If the RSSI trend is increasing, the aggressiveness counter is increased by the most recent charging duration. A longer charging duration indicates lower harvested power and fewer future active periods, and therefore produces a larger increment; a shorter duration indicates more frequent future transmission opportunities and produces a smaller increment. If the RSSI trend is decreasing, the counter is reduced using the same charging-time-dependent step size. Once the counter reaches zero, the system proactively stores packets in the NVM, which is managed as a FIFO buffer, allowing them to be sent later when channel conditions improve.

By combining channel trends with recent charging history, the system adjusts transmission aggressiveness according to both channel evolution and energy availability, exploiting favorable transmission opportunities while conserving energy under unfavorable conditions.

Energy management. The system executes on harvested energy buffered in a capacitor, and the MCU shuts down when the capacitor voltage falls below a prespecified threshold. To ensure forward progress, the system persists unsent packets and minimal transmission-control state in NVM, including the aggressiveness counter and RSSI trend state. When energy is restored, the MCU recovers the persisted packets and control state and resumes execution.

IV. Evaluation

A. Evaluation Setup

Prototype. We implement the controller on an MSP430FR5969 MCU [23] with 64 KB of on-chip FRAM. A TI CC1352P7 [24] measures RSSI and forwards it to the MCU via serial peripheral interface (SPI), while the backscatter tag follows the LoRea architecture [19]. An EM3028-C7 RTC [25] provides timing for trend recognition, aggressiveness control, and application timestamping.

Benchmarks. We evaluate three applications with different computation and communication demands. Agricultural environment measurement (AEM) performs computation-intensive

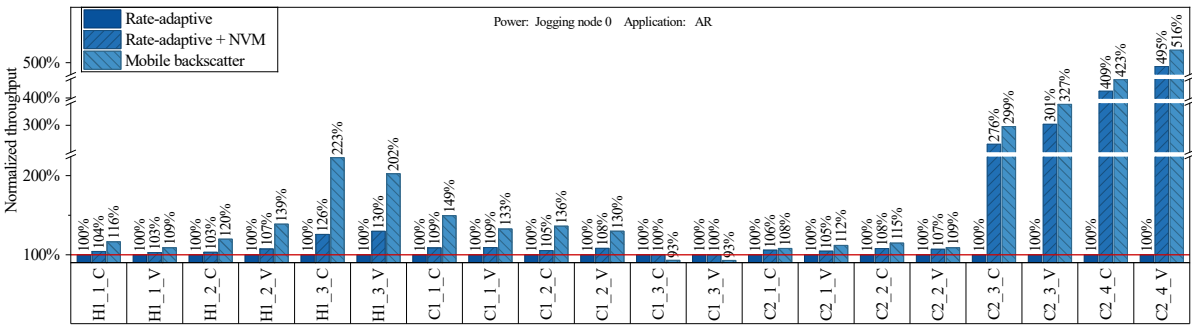


Fig. 3: Normalized throughput per signal trace for the AR application under the jogging node 0 power trace.

sensing and periodically transmits 256-bit packets [26]. Activity recognition (AR) performs lightweight activity recognition and transmits 128-bit packets [27]. Cold-chain equipment monitoring (CEM) applies LZW compression and transmits 640-bit packets in bursts [27].

Experimental setup. The carrier emitter and receiver are co-located and stationary, while the backscatter tag is mobile. The tag uses a capacitor sized according to the energy demand of the running application, ensuring sufficient energy to complete at least one active period. Since running the prototype for extended periods under reproducible signal and energy conditions is impractical, we build a simulator using task-level energy consumption and execution time measured on the hardware prototype. The simulator models capacitor charging and discharging, task execution, backscatter transmission, and their energy consumption. We use two one-hour real-world power traces collected from people jogging [15]. We also record 20 real-world signal traces in diverse indoor environments, including two corridors and a large hall. The traces are collected along 10 movement paths with both constant-speed and varying-speed mobility, with each trace lasting around two minutes. For long-term evaluation, each trace is repeated to form a one-hour trace. We name each trace based on its location, path index, and speed. For example, *C1_2_C* denotes corridor 1, path 2, with constant speed.

Baselines. We compare against two baselines. The *rate-adaptive* baseline follows Huang et al. [13] and adjusts the data rate according to instantaneous RSSI. The *NVM-based rate-adaptive* baseline extends the rate-adaptive baseline by buffering packets in NVM when they cannot be transmitted due to weak signals or energy failures. However, it does not use channel trends or energy status to regulate transmissions.

B. Throughput

To understand how our transmission control improves throughput under mobility and energy harvesting, we measure the number of packets completely transmitted in a one-hour simulation for each application.

Average throughput. Fig. 4 shows the average throughput normalized to the rate-adaptive baseline. On average, the mobile backscatter system can achieve 1.39 × the throughput of the rate-adaptive baseline for the AEM application under both power traces. For the AR application, it achieves 1.40×

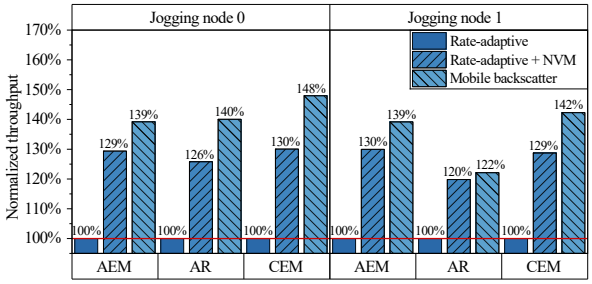


Fig. 4: Normalized average throughput. *The mobile backscatter system achieves up to 1.48 × average throughput.*

and 1.22× the throughput of the rate-adaptive baseline under jogging node 0 and jogging node 1 power traces, respectively. The CEM application achieves the highest throughput gains, reaching 1.48× and 1.42× the throughput of the rate-adaptive baseline under the two different power traces.

Throughput per trace. Normalized throughput varies across signal traces, depending on the channel dynamics experienced by the mobile tag. We use the AR application under the jogging node 0 and node 1 power traces as a representative example, as shown in Fig. 3 and Fig. 5. More detailed experimental results are available in our extended version [28].

Under the jogging node 0 power trace, the mobile backscatter system achieves its highest throughput under the *C2_4_V* signal trace. This trace remains below the transmission threshold for 63.4% of the time. During these periods, the mobile backscatter system stores generated packets in NVM, and later exploits the RSSI trend to transmit them under improved channel conditions, leading to significant throughput gains. In contrast, the NVM-based rate-adaptive baseline stores packets in NVM, but it does not exploit channel trends and energy status, resulting in lower throughput. The rate-adaptive baseline simply discards packets when transmission is not possible, leading to the lowest throughput.

The benefit, however, depends on whether future favorable channel conditions provide sufficient transmission opportunities to deliver packets accumulated under weak channel conditions. An exception is observed for the AR application under the jogging node 1 power trace and the *C2_4_V* signal trace, as shown in Fig. 5, where the mobile backscatter system performs worse than the NVM-based rate-adaptive baseline. Because AR has a lightweight computational workload and the jogging node 1 provides high input energy, the system generates a large

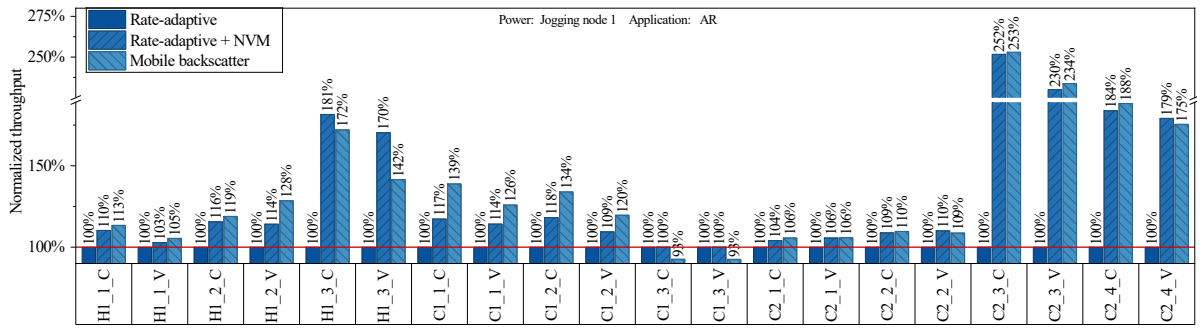


Fig. 5: Normalized throughput per signal trace for the AR application under the jogging node 1 power trace.

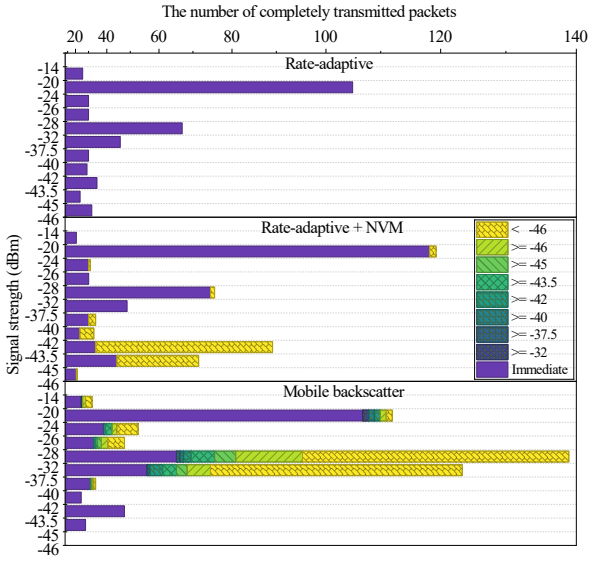


Fig. 6: Comparison of packet transmission behavior across RSSIs for the AEM application under the *C1_2_C* signal trace and jogging node 0 power trace.

number of packets during weak signal conditions. Since the same *C2_4_V* trace remains below the transmission threshold for 63.4% of the time, the subsequent favorable channel conditions provide insufficient opportunities to transmit all buffered packets. As a result, the mobile backscatter system retains more packets in NVM, while the NVM-based rate-adaptive baseline transmits whenever the minimum signal-strength and energy requirements are met. This yields higher throughput for the NVM-based rate-adaptive baseline in this case, although transmissions under weaker channel conditions are less energy-efficient. Similar limitations exist under the *H1_3_C* and *H1_3_V* signal traces.

The lowest normalized throughput is observed under the *C1_3_C* and *C1_3_V* signal traces. These traces are recorded when the tag moves at a fixed distance (3 meters) from the carrier emitter, with RSSI fluctuating between -20 dBm and -30 dBm and no clear trend. Under such scenarios, few transmissions need to be deferred, limiting the benefit of transmission control. Thus, the mobile backscatter system performs similarly to the baselines. The AR application exhibits slightly lower throughput than the baselines, as its lightweight workload makes the overhead noticeable.

Throughput under different RSSIs. Fig. 6 shows where packets are transmitted across RSSI levels for the AEM application under the *C1_2_C* signal trace and jogging node 0 power trace. The NVM-based rate-adaptive baseline stores packets generated below the -46 dBm transmission threshold, but sends them once the minimum signal and energy requirements are met, primarily between -45 dBm and -42 dBm. In contrast, the mobile backscatter system defers these packets and transmits them under stronger channel conditions, mainly between -37.5 dBm and -28 dBm. It also stores packets generated at usable RSSI when better transmission opportunities are expected. As a result, the mobile backscatter system transmits 619 packets, compared with 454 for the rate-adaptive baseline and 555 for the NVM-based rate-adaptive baseline. This gain is achieved by deferring packets generated under poor channel conditions and transmitting them under stronger channel conditions, thereby reducing energy consumption per packet and improving throughput.

C. Energy

We analyze task-level energy consumption to explain how the mobile backscatter system improves throughput. Fig. 7 presents the breakdown across the three applications under the *C1_2_C* signal trace and jogging node 0 power trace.

AEM is computation-intensive, with computation accounting for over 80% of total energy. The mobile backscatter system reduces the fraction of energy spent on transmission from 16% in the rate-adaptive baseline to 4%, while achieving 1.36 × the throughput of the rate-adaptive baseline.

AR has a lighter computational and communication demand than AEM. Computation accounts for around 25% of the total energy, while other tasks such as sensing and packet processing contribute another 30%. Transmission accounts for 43% and 49% of the energy in the two baselines, respectively, but only 27% in the mobile backscatter system. This improves transmission energy efficiency while achieving 1.36 × the throughput of the rate-adaptive baseline.

CEM transmits less frequently because data are compressed and sent in blocks. By deferring transmissions under poor channel conditions, the mobile backscatter system reduces the fraction of energy spent on transmission to 8%, compared with 24% and 26% for the baselines, and achieves 1.50 × the throughput of the rate-adaptive baseline.

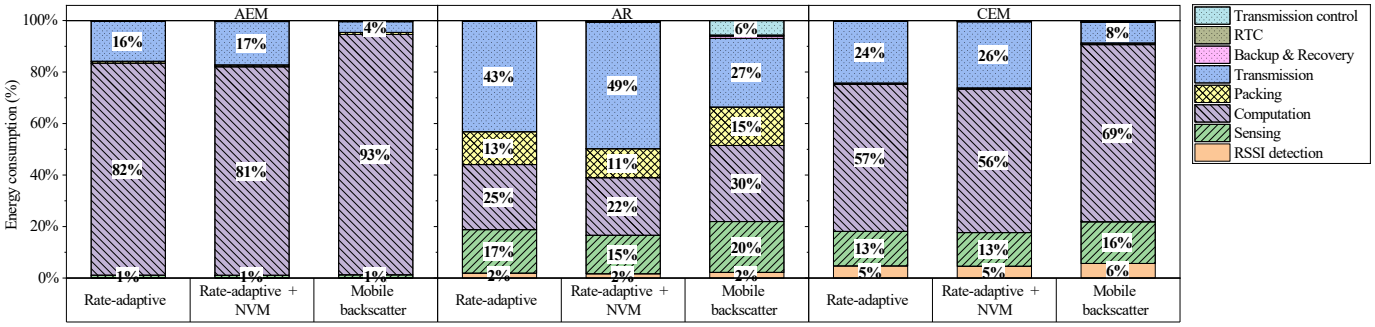


Fig. 7: Energy breakdown per application under the *C1_2_C* signal trace and jogging node 0 power trace.

Across all evaluation results, the mobile backscatter system achieves up to 47.3% reduction in transmission energy consumption compared with the rate-adaptive baseline. While improving transmission energy efficiency, the system also introduces energy overhead. In AEM and CEM, this overhead is negligible, accounting for only 0.32% and 0.23% of the total energy consumption, respectively, because other tasks dominate the energy budget. In AR, the lightweight workload and small packet size make this overhead more noticeable, where it accounts for around 6% of total energy. Nevertheless, the energy overhead remains small across all evaluated configurations, with a maximum of 7.3%.

The extended version of this paper includes two microbenchmarks that evaluate how different system factors affect throughput, together with an evaluation of packet age [28].

V. Conclusion

We presented a mobile backscatter communication system for battery-less mobile IoT devices. Our design introduces a lightweight transmission control that regulates packet transmissions based on real-time channel and energy variations. We persist critical state and packets in NVM, maintaining progress across energy failures and poor channel conditions where transmissions are inefficient or impossible. We implement this design on a hardware prototype based on the MSP430FR5969 MCU and evaluate the system using three representative IoT applications and real-world power and signal traces. Experimental results show that we achieve up to 5.16 $\times$ the throughput of the rate-adaptive baseline, enabled by up to 47.3% reduction in transmission energy consumption, while introducing only 0.23%–7.3% additional energy overhead.

Acknowledgements. This work is supported by the Swedish Foundation for Strategic Research (SSF) and by the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.3—Call for Tender No. 1561 of 11.10.2022 of the Italian Ministero dell'Università e della Ricerca (MUR); funded by the European Union—NextGenerationEU.

References

[1] V. Liu *et al.*, "Ambient backscatter: Wireless communication out of thin air," *ACM SIGCOMM computer communication review*, 2013.
[2] N. Van Huynh *et al.*, "Ambient backscatter communications: A contemporary survey," *IEEE Communications surveys & tutorials*, 2018.
[3] B. Kellogg *et al.*, "Wi-fi backscatter: Internet connectivity for rf-powered devices," in *ACM Conference on SIGCOMM*, 2014.
[4] Y. Peng *et al.*, "Plora: A passive long-range data network from ambient lora transmissions," in *Conference of the ACM special interest group on data communication*, 2018.
[5] A. Wang *et al.*, "FM backscatter: Enabling connected cities and smart fabrics," in *NSDI*, 2017.
[6] C. A. Boano *et al.*, "Exploiting the lqi variance for rapid channel quality assessment," in *IPSN*, 2009.
[7] S. Ahmed *et al.*, "The internet of batteryless things," *Communications of the ACM*, 2024.
[8] S. Khalifa *et al.*, "Harke: Human activity recognition from kinetic energy harvesting data in wearable devices," *IEEE Transactions on Mobile Computing*, 2017.
[9] L. Wang *et al.*, "Overview of human kinetic energy harvesting and application," *ACS Applied Energy Materials*, 2022.
[10] Z. Lai *et al.*, "Self-powered and self-sensing devices based on human motion," 2022.
[11] A. Haroun *et al.*, "Progress in micro/nano sensors and nanoenergy for future aiot-based smart home applications," *Nano Express*, 2021.
[12] T. S. Muratkar *et al.*, "Ambient backscatter communication with mobile rf source for iot-based applications," *AEU-International Journal of Electronics and Communications*, 2021.
[13] G. Huang *et al.*, "FreeBack: Blind and distributed rate adaptation in LoRa-based backscatter networks," in *WCNC*, 2021.
[14] N. A. Bhatti *et al.*, "Energy harvesting and wireless transfer in sensor network applications: Concepts and experiences," *ACM Transactions on Sensor Networks (TOSN)*, 2016.
[15] K. Geissdoerfer and M. Zimmerling, "Learning to communicate effectively between battery-free devices," in *NSDI*, 2022.
[16] S. Ahmed *et al.*, "Efficient intermittent computing with differential checkpointing," in *International Conference on Languages, Compilers, and Tools for Embedded Systems*, 2019.
[17] N. A. Bhatti and L. Mottola, "Harvos: Efficient code instrumentation for transiently-powered embedded sensing," in *IPSN*, 2017.
[18] K. Maeng and B. Lucia, "Adaptive dynamic checkpointing for safe efficient intermittent computing," in *OSDI*, 2018.
[19] A. Varshney *et al.*, "Lorea: A backscatter architecture that achieves a long communication range," in *SenSys*, 2017.
[20] V. Talla *et al.*, "Lora backscatter: Enabling the vision of ubiquitous connectivity," *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, 2017.
[21] Y. Xu *et al.*, "ecUWB: A Energy-Centric Communication Scheme for Unstable WiFi Based Backscatter Systems," in *International Conference on Computer Communications and Networks (ICCCN)*, 2020.
[22] M. Allman *et al.*, "TCP congestion control," Tech. Rep., 2009.
[23] "MSP430FR5969," https://www.ti.com/product/MSP430FR5969, Accessed: 2026.
[24] "CC1352P7," https://www.ti.com/product/CC1352P7, Accessed: 2026.
[25] "EM3028-C7," https://www.emmicroelectronic.com/product/real-time-clocks-modules/em3028-c7, Accessed: 2026.
[26] S. N. Daskalakis *et al.*, "Ambient backscatterers using fm broadcasting for low cost and low power wireless applications," *IEEE Transactions on Microwave Theory and Techniques*, 2017.
[27] K. Maeng *et al.*, "Alpaca: Intermittent execution without checkpoints," *Proceedings of the ACM on Programming Languages*, 2017.
[28] W. Song *et al.*, "Mobile backscatter communication for the battery-less internet of things," 2026. [Online]. Available: https://arxiv.org/abs/2609.01465