Co-Scheduling of Charging and Transmission in Supercapacitor Powered IoT

Aakriti Jain[1], Venkata Prashant Modekurthy[2], Abusayeed Saifullah[1], Marco Brocanelli[3]
[1]University of Texas at Dallas [2]University of Nevada, Las Vegas [3]Ohio State University
USA

***Abstract*—Supercapacitor-based energy harvesting offers a sustainable alternative to battery-powered IoT but introduces unique challenges when instantaneous harvested power cannot support simultaneous charging and data transmission. This limitation creates a critical trade-off: immediate transmission risks energy depletion, while prioritizing charging induces latency and suffers from unavoidable supercapacitor self-discharge. Existing battery-centric scheduling approaches overlook these factors, failing to capture energy leakage during idle periods. To address these limitations, we introduce the Charging-Transmission Co-Scheduling Problem, aiming to jointly optimize energy accumulation and packet delivery under stochastic harvesting conditions. First, we formulate this problem as a multi-objective Integer Linear Program (ILP) that maximizes network throughput and packet utility (reflecting data timeliness), explicitly accounting for supercapacitor-specific behaviors such as self-discharge dynamics and rapid charge-discharge cycles. To enable deployment on resource-constrained devices, we derive a relaxed Linear Program (LP) for efficient online approximation and a lightweight lookahead heuristic for devices unable to run LPs. As a case study, we integrate the framework into a Low-Power Wide-Area Networks (LPWANs) Media Access Control (MAC) protocol and show via practical experiments and NS-3 simulations that our approaches improve throughput and utility by 10–25% over baselines, while remaining close to the optimal ILP.**

I. Introduction

The rapid growth of low-power Internet of Things (IoT) applications is revealing the cost, lifetime, and sustainability limitations of battery-powered devices, motivating a shift toward battery-free systems powered by renewable energy sources such as solar, wind, and thermal [1], [2]. Supercapacitors are emerging as attractive alternatives due to their high power density, fast charge–discharge capability, and longevity, enabling deployment across short- to long-range networks for applications including environmental monitoring, agriculture, and smart cities [3]–[9]. However, in many deployments, the harvested power is insufficient to support simultaneous transmission and charging due to the limited harvester capacity and small form factors, invalidating the assumption of continuous operation as devices shrink. Under these constraints, devices must dynamically allocate energy between charging and transmission. However, simplistic strategies can significantly degrade performance. Blindly transmitting may rapidly deplete stored energy, leading to outages when harvesting opportunities arise, while over-prioritizing charging can cause packet backlogs and increased latency. These effects are further amplified in large-scale networks with high channel contention.

Existing approaches [4], [6] adapt sampling rates or power but do not explicitly coordinate *when to charge versus when to transmit*. Furthermore, supercapacitors suffer from *self-discharge*, causing energy loss even without activity. Existing lifetime maximization approaches [10], [11], designed for batteries with negligible leakage, are not suited for supercapacitor-powered IoT. Prior work on battery-free and energy-harvesting systems [12], [13] have primarily focused on synchronization and communication feasibility, which is fundamentally different from our problem and do not model supercapacitor self-discharge. Work on joint charging and scheduling [14] consider externally powered wireless chargers that control node charging decisions, a setting significantly different from passive ambient energy harvesting (e.g., solar). New challenges arise from the dynamic energy availability and packet arrivals; for instance, sudden environmental changes may trigger urgent transmissions when solar input is low. Such conditions require adaptive, online decision-making rather than static policies.

In this paper, we overcome the above challenges by introducing the **Charging-Transmission Co-Scheduling Problem** in supercapacitor-powered IoT. The goal is to *jointly schedule energy charging and packet transmissions* to maximize throughput and minimize waste, while accounting for self-discharge. We formulate this problem as a multi-objective integer linear program (ILP) that maximizes packet utility (which models the time-sensitive value of data) and throughput while addressing energy dynamics and self-discharge, providing an empirical performance bound. We then derive a linear program (LP) relaxation for efficient online approximation. To support resource-constrained IoT devices that may not afford running LP, we also propose a lightweight *lookahead heuristic* that uses energy predictions to prioritize transmissions during scarcity and charge during surplus. Finally, we customize the Media Access Control (MAC) protocol for Low-Power Wide-Area Networks (LPWANs) as a case study to incorporate this framework. To our knowledge, this is the first study of charging-transmission co-scheduling in supercapacitor-operated IoT. Through practical experiments and extensive simulations based on NS-3, we demonstrate that our approach significantly outperforms baseline approaches in terms of cumulative throughput and utility by approximately 10–25%, while closely approximating the performance of the optimal ILP solution.

In the rest of the paper, Section II provides the necessary

background and ILP formulation. Section III describes the proposed LP relaxation and heuristic solution. Section IV describes a MAC protocol for large-scale LPWAN. Section V presents the results and VI concludes the paper.

II. Problem Formulation

In this section, we formulate the Charging–Transmission Co-Scheduling problem as an **Integer Linear Program (ILP)** for a single supercapacitor-powered IoT node. In Section IV, we extend this formulation to multi-node settings. The objective is to jointly maximize network throughput and packet utility, while explicitly modeling supercapacitor-specific characteristics such as self-discharge.

Supercapacitor Background. Supercapacitors (also known as ultracapacitors) store energy electrostatically rather than chemically. Their behavior can be accurately modeled using standard capacitor dynamics. When charged with a constant current I, the time required to increase the voltage from V_i to V_f is $t = C(V_f - V_i)/I$, where C is the capacitance. Under constant-voltage charging, the capacitor voltage follows $V(t) = V_{\text{source}}\left(1 - e^{-t/(RC)}\right)$, requiring $t = -RC\ln\left(1 - \frac{V_f}{V_{\text{source}}}\right)$ to reach voltage V_f, where R captures the equivalent circuit resistance. During discharge, the voltage decays exponentially as $V(t) = V_0 e^{-t/(RC)}$. The energy stored at voltage V is $E = \frac{1}{2}CV^2$, and the energy released when the voltage drops from V_{initial} to V_{final} is $E_{\text{discharge}} = \frac{1}{2}C\left(V_{\text{initial}}^2 - V_{\text{final}}^2\right)$. Supercapacitors also exhibit *self-discharge*, whereby stored energy decays automatically over time even when the device is idle.

We consider an IoT node powered by a supercapacitor and a green energy harvesting source. Rather than directly powering the node, the solar panel charges the supercapacitor, which serves as a green-energy buffer [15]. The node periodically generates data packets and must decide, at each decision point, whether to transmit the packet or to prioritize charging the supercapacitor. Since packet transmission and charging cannot occur concurrently, scheduling decisions directly affect both energy availability and packet delay. Transmitting immediately may deplete the supercapacitor, whereas delaying transmission allows additional energy accumulation but incurs utility loss due to aging data and unavoidable self-discharge during idle periods. Therefore, we formulate the Charging–Transmission Co-Scheduling Problem as a **multi-objective optimization** framework that jointly maximizes the network throughput and packet utility under stochastic energy conditions.

A. Model and Decision Variables

We adopt a discrete-time model where time is divided into equal-length sampling periods indexed by $i = 1, \ldots, n$, each of duration T seconds. The optimization is solved over a *sliding window* of n sampling periods. For each sampling period i, the scheduler determines whether the packet generated in that period should be transmitted and, if so, when transmission should begin within the period. At the end of the period, the window advances by one step (period), and the optimization is re-solved using updated decision variables and energy information.

For each sampling period $i \in \{1, \ldots, n\}$, $X_i \in \{0, 1\}$ is a binary variable that indicates whether the packet in period i is transmitted. If a packet is transmitted, Y_i denotes the delay (in seconds), measured from the beginning of period i to the start of packet transmission. $Y_i \in [0,\, T - \rho_{\max}]$, where $\rho_{\max}$ denotes the maximum allowable transmission duration since sampling. E_i represents energy stored in the supercapacitor at the end of period i. E_0 is the initial energy at the start of the first period in the sliding window. $E_{\min}$ is the worst-case energy required for a successful packet transmission. $E_{\max}$ is the maximum energy storage capacity of the supercapacitor. R_i denotes the green energy harvested during period i with $R_{\max}$ being the maximum harvestable energy per period.

Let D denote the maximum energy lost due to self-discharge within one sampling period. We model self-discharge only when neither packet transmission nor energy harvesting occurs during the period. Although self-discharge may occur at finer time scales (e.g., when energy is harvested during part of a period but absent later), accurately capturing such behavior would require millisecond-scale energy prediction and additional decision variables, substantially increasing model complexity. Moreover, the self-discharge rate of a supercapacitor is typically much smaller than its charging rate and negligible compared to transmission energy consumption. Accordingly, we introduce a binary variable $\beta_i \in \{0, 1\}$ to indicate whether self-discharge occurs during sampling period i, where $\beta_i = 1$ if the supercapacitor self-discharges and $\beta_i = 0$ otherwise.

B. ILP Formulation

We now present the complete ILP formulation, which jointly optimizes packet utility and throughput under energy dynamics, self-discharge behavior, and feasibility constraints.

Objective 1: Maximize Total Utility: Packet *utility* reflects the usefulness of sensed data and decreases with transmission delay. We model utility as a linearly decaying function within a sampling period. Let $U_i \in [0, 1]$ denote the utility of the packet generated in period i. The following constraints define the utility model:

$$U_i \le X_i,$$
$$U_i \le \left(1 - \frac{Y_i}{T}\right),$$
$$0 \le U_i \le 1.$$

The first constraint ensures that a packet contributes utility only if it is transmitted, while the second captures utility degradation as transmission is delayed within the period. The total utility objective is:

$$\max \sum_{i=1}^{n} U_i.$$

Objective 2: Maximize Total Throughput: Throughput is defined as the total number of successfully transmitted packets within the optimization window. This is expressed as:

$$\max \sum_{i=1}^{n} X_i.$$

Additional Constraints:

a) Binary and bound constraints: The transmission decision is binary and the transmission delay must be feasible within the sampling period:

$$X_i \in \{0,1\}, \quad Y_i \in [0, T-\rho_{\max}] \quad \forall i.$$

b) Energy harvesting and update equation with self-discharge: We model self-discharge using β_i. Specifically, the supercapacitor discharges D units of energy within one period if a packet is not transmitted during the i^{th} period and green energy is not available during the i^{th} period. Thus:

$$\beta_i > (1 - X_i)\left(1 - \frac{R_i}{R_{\max}}\right) - 1.$$

When $X_i = 0$ and $R_i = 0$, the above constraint reduces to $\beta_i > 0$, and the smallest value of β_i that satisfies the inequality is 1. When $X_i > 0$ or $R_i > 0$, the above constraint reduces to β_i being greater than a negative number, and the smallest value of β_i that satisfies the inequality is 0.

Incorporating β_i, the constraint on energy harvesting is:

$$E_i \leq E_{i-1} - (X_i \cdot E_{\min}) - (\beta_i \cdot D) + \left(\left(\frac{T-\rho_{\max}}{T}\right) X_i \cdot R_i\right).$$

This constraint captures the net energy balance of the supercapacitor over sampling period i by accounting for energy carryover, transmission consumption, supercapacitor self-discharge, and energy harvesting under the non-overlapping charging–transmission constraint. The term E_{i-1} represents the energy carried over from the previous sampling period and serves as the initial energy budget available for both charging and transmission decisions in period i. The transmission cost is captured by $X_i \cdot E_{\min}$. It models the worst-case energy consumption of a packet transmission. Since X_i is binary, this term subtracts $E_{\min}$ only when a packet is transmitted during period i, ensuring that the formulation remains conservative and feasible under varying channel and hardware conditions.

Energy loss due to supercapacitor self-discharge is accounted by the term $\beta_i \cdot D$. When $\beta_i = 1$, the node experiences an energy loss of at most D units during period i. As enforced by the self-discharge activation constraint, this loss occurs only during periods in which the node is neither transmitting nor harvesting energy, i.e., when the supercapacitor remains idle and leakage dominates the energy dynamics. Since both throughput and utility objectives benefit from higher available energy, the optimizer implicitly minimizes β_i at each time step, resulting in the smallest admissible self-discharge consistent with the node's operational state. This coupling prevents overestimation of the stored energy and avoids artificially preserving energy across idle periods.

The harvesting term $\left(\left(\frac{T-\rho_{\max}}{T}\right) X_i \cdot R_i\right)$ captures the fact that charging and transmission cannot occur simultaneously. When a packet is transmitted ($X_i = 1$), the transmission occupies at most $\rho_{\max}$ seconds within the sampling period, leaving only $T - \rho_{\max}$ seconds for energy accumulation. Assuming that harvested energy is uniformly distributed over time, the factor $\frac{T-\rho_{\max}}{T}$ therefore captures the fraction of the sampling period available for charging, assuming energy harvesting is uniformly distributed over time. When no packet is transmitted ($X_i = 0$), the harvesting contribution is naturally suppressed, as energy accumulation in that case is already captured by the self-discharge modeling and feasibility constraints.

c) Minimum energy requirement for transmission: A packet can be transmitted in sampling period i only if there exists a delay Y_i such that the energy stored in the supercapacitor at the transmission time is greater than $E_{\min}$. This feasibility requirement is modeled as:

$$X_i \leq \frac{E_{i-1} + \left(\frac{Y_i}{T}\right) R_i}{E_{min}} \quad \forall i.$$

This constraint enforces the feasibility of packet transmission by ensuring that sufficient energy is available in the supercapacitor at the transmission start time. The term E_{i-1} captures the residual energy carried over from the previous sampling period, while $\left(\frac{Y_i}{T}\right) R_i$ accounts for the portion of harvested energy accumulated prior to the transmission delay Y_i, assuming uniform energy harvesting within the period. By normalizing the available energy with respect to the minimum transmission energy $E_{\min}$, the constraint guarantees that a packet can be scheduled for transmission only if the supercapacitor energy meets the worst-case requirement.

d) Energy Capacity Constraint: Energy harvested cannot exceed the maximum capacity of the supercapacitor:

$$E_i \leq E_{\max}.$$

The ILP is solved using a sliding-window approach over n sampling periods, subject to the constraints. At the beginning of each period, the node updates its initial energy state and the predicted harvested energy over the optimization horizon. The ILP is then solved for the n-period window. However, only the decision variables corresponding to the first period of the window are retained for execution. After executing these decisions, the window shifts forward by one period, and the optimization is solved again over the updated horizon using the retained state variables.

While the ILP formulation yields optimal co-scheduling decisions, solving it incurs significant computational overhead due to binary variables and coupling across time. This complexity is prohibitive for resource-constrained IoT nodes with limited memory and processing capability. We derive resource-friendly alternatives in the next section.

III. Proposed Solution

To enable practical implementation on resource-constrained devices, we propose a relaxed LP for online approximation and, for devices unable to run LP, a lightweight heuristic.

A. LP relaxation

We derive an **LP relaxation** that enables significantly faster computation compared to the ILP. The relaxation is obtained by replacing the binary decision variables in the ILP with relaxed decision variables while preserving the original objective structure. The resulting multi-objective *LP formulation serves as our primary solution* and can be efficiently solved in practical IoT settings. Specifically, we introduce an integer decision variable Z_i constrained as $0 \leq Z_i \leq 1$, which captures the

scheduler's preference for transmitting the packet generated in sampling period i, when $Z_i \geq 0.5$, or for skipping, otherwise. This threshold-based interpretation allows the LP solution to guide discrete transmission decisions while avoiding the computational cost of enforcing integrality. The LP is solved using a sliding-window approach over n sampling periods.

Objective 1: Maximize Total Utility: Under the LP relaxation, the utility of each packet continues to reflect its timeliness, but is now coupled to the relaxed transmission decision. The utility is thus characterized by three constraints: 1) $U_i' \leq \left(1 - \frac{Y_i}{T}\right)$, 2) $U_i \leq min(Z_i + 0.5, 1)$, and 3) $0 \leq U_i \leq U_i'$. The auxiliary variable U_i' represents the maximum attainable utility for packet i given its transmission delay, capturing the same linear utility decay as in the ILP. The term $(Z_i + 0.5)$ serves as a conservative surrogate for the binary transmission decision. The second constraint links packet utility to the relaxed transmission decision, ensuring that a packet contributes non-zero utility only when the LP solution indicates a sufficient preference for transmission. The third constraint ensures that utility remains bounded by both delay-induced degradation and the relaxed feasibility of transmission. Thus, the objective $\max \sum_{i=1}^{n} U_i$ favors timely packet delivery while remaining consistent with the relaxed transmission decisions.

Objective 2: Maximize Total Throughput: Throughput under the LP relaxation is defined as the sum of the relaxed transmission decisions as $\max \sum_{i=1}^{n} Z_i$. Although Z_i is no longer binary, larger values indicate stronger preference for transmitting a packet in the corresponding sampling period. Maximizing the sum of Z_i thus encourages the scheduler to prioritize transmissions that are both energy-feasible and utility-efficient under the relaxed formulation.

Additional Constraints:

a) Bound constraints.: The relaxed transmission decision and delay satisfy that $Z_i \in [0, 1]$ and $Y_i \in [0, T - \rho_{\max}] \quad \forall i$. The relaxed transmission decision Z_i is constrained to the unit interval, and the transmission delay Y_i is restricted to lie within the feasible portion of the sampling period. These bounds preserve the physical interpretation of the decision variables while maintaining linearity.

b) Energy harvesting and self-discharge.: The supercapacitor discharges D units of energy within one period if a packet is not transmitted during the i^{th} period and green energy is not available during the i^{th} period. Under relaxation, we define α_i to capture the discharge behavior such that $\alpha_i \geq (Z_i - 0.5)\left(1 - \frac{R_i}{R_{\max}}\right)$. The relaxed energy update constraint incorporates the fractional transmission decision through the term $(Z_i - 0.5)$, conservatively accounting for both transmission energy consumption and the reduction in available harvesting time due to transmission. When $R_i = 0$ and $Z_i \leq 0.5$, the above constraint reduces to $\alpha_i \geq 1$, and the smallest value of α_i that satisfies the inequality is 1. When $X_i > 0$ or $R_i > 0.5$, the smallest value of α_i that satisfies the above inequality is $(Z_i - 0.5)\left(1 - \frac{R_i}{R_{\max}}\right)$. Incorporating α_i, the constraint on energy harvesting is:

$$E_i \leq E_{i-1} - (Z_i + 0.5) \cdot E_{\min} - \alpha_i \cdot D + (Z_i + 0.5)\left(\frac{T - \rho_{\max}}{T}\right) R_i$$

This constraint enforces a conservative upper bound on the energy stored in the supercapacitor at the end of sampling period i under the relaxed transmission decision. The transmission energy cost and the harvesting opportunity are both scaled by $Z_i + 0.5$, ensuring that when the LP solution favors transmission ($Z_i \geq 0.5$), the formulation fully accounts for the worst-case transmission energy consumption and the corresponding reduction in charging time. The self-discharge term $\alpha_i \cdot D$ captures leakage during idle periods. The objective function maximizes the energy stored in the supercapacitor since it facilitates higher throughput and utility, and hence, α_i is minimized at every time step, resulting in a bound on the estimate of energy stored in the supercapacitor as:

$$E_i \leq E_{i-1} - E_{\min} + \left(\frac{T - \rho_{\max}}{T}\right) R_i.$$

c) Minimum energy requirement for transmission: The packet can be transmitted if there exists a Y_i at which the energy stored in the supercapacitor is greater than E_{min}:

$$Z_i + 0.5 \leq \frac{E_{i-1} + \left(\frac{Y_i}{T}\right) R_i}{E_{min}} \quad \forall i.$$

The minimum energy requirement for transmission further couples the relaxed decision variable Z_i with the transmission timing variable Y_i. This constraint ensures that a strong transmission preference ($Z_i \geq 0.5$) is assigned only when sufficient energy can be accumulated in the supercapacitor prior to the transmission start time. As a result, the LP assigns high values of Z_i only when both energy availability and timing feasibility are satisfied, preventing the LP from favoring transmissions that are not physically feasible.

d) Energy capacity constraint: The energy harvested cannot be greater than the maximum energy that can be stored in the supercapacitor $E_i \leq E_{\max}$.

Note that the LP relaxation can become infeasible in two rare cases: (1) when the available energy is less than half the minimum required for transmission, forcing $Z_i < 0$ and violating feasibility bounds, and (2) when the required transmission energy exceeds both stored and harvested energy, making the relaxed energy constraint unsatisfiable. In both cases, the optimal decision is to skip transmission in the current period and recompute in the next.

Single-Objective Weighted-Sum LP: Although the LP relaxation significantly reduces the computational complexity of the original ILP, it remains a multi-objective optimization problem, which may still be impractical to solve on resource-constrained embedded devices. To further simplify execution, we approximate the multi-objective LP as a weighted sum of throughput and utility, while retaining all feasibility and energy constraints unchanged. Specifically, we combine the two objectives into a single scalar objective:

$$\max \ \lambda \sum_{i=1}^{n} U_i + (1 - \lambda) \sum_{i=1}^{n} Z_i, \quad 0 \leq \lambda \leq 1 \tag{1}$$

where U_i denotes the utility of the packet generated in sampling period i, and Z_i is the relaxed transmission decision variable. The weight λ controls the trade-off between packet timeliness and throughput. λ can be selected based on application requirements. Delay-sensitive applications favor larger values of λ to prioritize packet utility, while throughput-oriented or delay-tolerant applications prefer smaller values of λ to maximize the number of delivered packets.

All constraints of the LP relaxation, including energy evolution, self-discharge modeling, transmission feasibility, and supercapacitor capacity limits, are preserved without modification. Only the preference between sending more packets vs. sending fresher packets is adjusted via the weight. As a result, the weighted-sum formulation maintains the same physical feasibility guarantees as the LP while reducing the optimization to a single-objective problem that is easier to solve and tune in embedded environments.

The LP relaxation provides a computationally efficient approximation of the original ILP formulation, striking a balance between optimality and practical deployability. By relaxing integrality constraints, it preserves the core structure of the co-scheduling problem, while enabling polynomial-time solution with efficient bound. Although the LP may produce fractional decisions that require thresholding, these solutions retain tight performance bounds and provide effective guidance for transmission scheduling. As such, the LP serves as the core decision-making engine of our framework. However, repeated LP execution may incur non-trivial computational and energy overhead on highly resource-constrained IoT nodes. Thus, we additionally propose a lightweight heuristic that distills the insights of the LP formulation into simple decision rules suitable for ultra-low-power devices.

B. The Lookahead Heuristic

Naïve heuristics such as always prioritizing charging whenever green energy is available or transmitting packets immediately upon arrival without considering the supercapacitor's charge state, perform poorly in dense network deployments. Uncoordinated transmissions can rapidly deplete stored energy and miss opportunities to harvest abundant green energy, while indiscriminate charging can waste transmission opportunities and intensify contention when many nodes transmit simultaneously. To address these limitations, we propose the *lookahead heuristic*, which looks at the energy generated in time windows into the future and decides whether the packet should transmitted within the current period or not.

The lookahead heuristic leverages short-term green energy predictions to adaptively prioritize transmissions during energy scarcity and opportunistically charge during periods of surplus. The heuristic operates at each sampling period and supports partial charging and transmission within a period: once a packet transmission completes, any remaining time in the period is used for charging before next packet arrives.

At the beginning of each sampling period, the node wakes up and checks whether a packet is available for transmission. If a packet is present, the node first evaluates whether sufficient energy can be made available to support its transmission within the current period. If so, the packet is marked as a candidate for transmission. Before committing to transmission, the heuristic examines a finite number of upcoming periods, referred to as the *lookahead window*[1], to predict future green energy availability. The length of the lookahead window is user-defined and can be selected based on application requirements and device capabilities.

If the predicted energy availability within the lookahead window is uniform or non-scarce, the packet is transmitted immediately, and any remaining time in the period is used to charge the supercapacitor. However, if the lookahead window indicates imminent energy scarcity but surplus energy in the current period, the heuristic may choose to drop the current packet and prioritize charging instead. This decision also aims to improve overall transmission utility. Dropping a packet in an earlier period can enable subsequent packets to be transmitted earlier within their respective periods, yielding higher overall utility. Conversely, transmitting immediately may delay future transmissions due to insufficient charging opportunities, thereby reducing their utility. Accordingly, the heuristic drops the packet only when doing so is predicted to increase cumulative utility over the lookahead window; otherwise, the packet is transmitted.

If sufficient energy is not available, the node prioritizes charging until just enough energy is accumulated to support a transmission within the current period. Once this threshold is reached, the packet is transmitted, and the remaining time is again used for charging. If the node is unable to accumulate sufficient energy within the period to support a transmission, the packet is dropped and the entire period is scheduled to charge the supercapacitor.

By jointly considering the current energy state, partial charging opportunities, and predicted future energy availability, the lookahead heuristic avoids both premature transmissions and missed charging opportunities, enabling efficient and utility-aware charging–transmission co-scheduling on resource-constrained IoT devices.

The LP formulation jointly optimizes charging and transmission decisions over the entire optimization window, capturing long-term trade-offs among energy accumulation, transmission timing, and packet utility. In contrast, the lookahead heuristic makes decisions only for the current sampling period using short-term energy predictions, significantly reducing computational overhead but without guaranteeing global optimality. Thus, the heuristic is not intended to replace the LP, but rather to provide a practical fallback for devices where solving the LP online is infeasible.

IV. The MAC protocol

We next apply the proposed charging–transmission co-scheduling framework to LPWANs as a representative deployment scenario. In particular, we propose a new MAC

[1]The lookahead window used in the heuristic corresponds to the same sliding-window horizon employed in the LP formulation. Both represent a finite future time window over which energy availability is considered to guide current scheduling decisions, and the terms are used interchangeably.

protocol to incorporate energy-aware scheduling decisions derived from our formulation. We use LoRa (Long Range) [16] as an example LPWAN technology due to its wide adoption in energy-constrained IoT deployments. While our design is instantiated for LoRa, the proposed MAC-level integration is general and can be adapted to other LPWAN technologies.

LoRa employs Chirp Spread Spectrum (CSS) modulation, enabling robust long-range communication at low SNR by spreading signals over wide bandwidths. It supports configurable spreading factors (SF7–SF12), bandwidth, coding rate, and carrier frequency, trading data rate for time-on-air and energy consumption. In the US 902–928 MHz ISM band, LoRa provides multiple uplink (125 kHz/500 kHz) and downlink (500 kHz) channels. The LoRaWAN MAC operates using pure ALOHA, with no built-in collision resolution.

We consider a dense LoRa network comprising thousands of low-power nodes communicating with one or more gateways. Nodes remain in sleep mode for most of the time and periodically wake up to sense the environment and transmit the collected data. Each node is powered by a supercapacitor, which in turn is powered by a green energy source. LoRa networks typically support delay-tolerant applications such as smart farming and wildlife monitoring [17], [18], where the usefulness (or freshness) of sensed data degrades gradually over time. For most applications, a moderate reduction in utility is acceptable, whereas intermittent operation (i.e., missed packet deliveries) is undesirable; therefore, packets are expected to be delivered within a reasonable delay. Nodes operate under a combination of time-triggered and event-triggered sampling regimes.

We consider a solar panel as the green energy source. The instantaneous power generated by the energy harvester varies over time and is influenced by environmental conditions such as reduced solar availability during dawn and dusk. Nodes predict the harvested energy using a lightweight neural network trained at the gateway, as described in [19]. We assume that the average harvested power remains approximately constant over a sampling period, enabling stable charging and transmission decisions over short time horizons.

LoRaWAN supports parallel packet reception on multiple m_{up} uplink channels. When combined with s SFs, this results in a total of $s * m_{\text{up}}$ orthogonal *Virtual Channels (VCs)*, that can support concurrent transmissions [20], [21]. The gateway exploits this capability to partition nodes into *virtual groups*, where all nodes within a group share the same VC. Nodes assigned to different VCs do not interfere with each other, allowing interference to be localized within each virtual group.

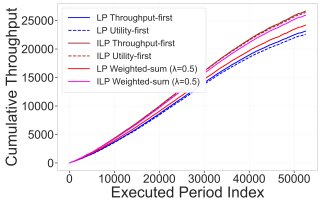
In addition to VC-based grouping, nodes are further separated based on their green energy generation patterns. Specifically, nodes that are geographically distant are assigned to the same virtual group. The intuition is that nearby nodes tend to exhibit similar solar energy harvesting profiles and therefore experience energy surplus and scarcity at similar times. Placing such nodes within the same group would increase the likelihood of overlapping charging or transmission decisions, leading to higher collision probability. In contrast, grouping nodes with dissimilar energy harvesting patterns reduces temporal overlap in their transmission attempts, thereby lowering contention within a VC. As a result, the MAC protocol effectively reduces the network-wide scheduling problem to co-scheduling charging and transmission decisions *within* each virtual group.

Within a virtual group, the charging–transmission co-scheduling problem is solved independently at each node. Time is divided into fixed-length sampling periods of duration T. At the beginning of each period, a node wakes up from sleep, performs sensing, and determines whether a packet is available for transmission. Based on its current supercapacitor energy level and predicted green energy availability, the node locally decides whether to transmit the packet or prioritize charging by executing either the LP-based solution or the lightweight lookahead heuristic, depending on application requirements and device capabilities. If a transmission is scheduled, it occupies at most $\rho_{\max}$ seconds, after which the radio returns to sleep and the supercapacitor is charged for the remainder of the period. All decisions are made locally, without exchanging state information or coordinating with other nodes. This eliminates the need for time synchronization and control message dissemination, making the proposed MAC highly scalable.

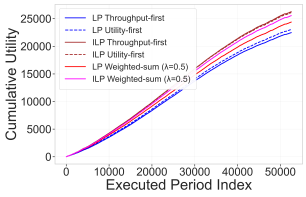
Since nodes in different groups use orthogonal VCs, simultaneous transmissions from two or more nodes from different virtual groups do not collide collide. Only if two nodes within the same virtual group independently transmit in the same sampling period, a collision may occur; however, such events are rere since virtual group are selected to have diverse energy harvesting patterns and minimal overlap probability. When confirmed uplink transmissions are enabled in LoRaWAN, successful packet reception is acknowledged by the gateway using a standard LoRaWAN downlink acknowledgment (ACK). After completing an uplink transmission, a node opens the receive windows RX1 (and RX2 if needed) at predefined offsets, during which the gateway may transmit an ACK. If no ACK is received, the node infers transmission failure. If a packet transmitted in period i is not acknowledged, it is considered lost and is not retained for retransmission. When a new packet is generated at the beginning of period $i + 1$, it replaces any previously undelivered packet in the buffer. This design choice reflects the delay-tolerant nature of typical LoRa applications, where delivering the most recent (freshest) data is preferred over retransmitting old data whose utility has already decayed. If a new node joins the network at any time, it is assigned a VC, and the grouping is redone. If an old node leaves the network, the network operates as is and no additional action is needed.

V. Evaluation

We evaluate the proposed charging–transmission co-scheduling framework at three levels. First, we compare the ILP and LP formulations for a single node to validate the effectiveness of the LP relaxation. Second, we conduct large-scale NS-3 simulations to evaluate the LP-based scheduler and

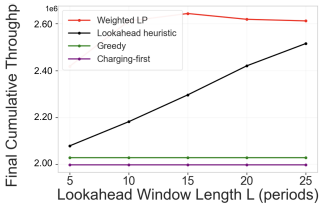


(a) Cumulative Throughput

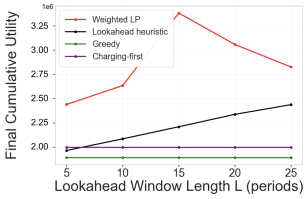


(b) Cumulative Utility

Fig. 1: Comparison of LP and ILP formulations with leakage current.

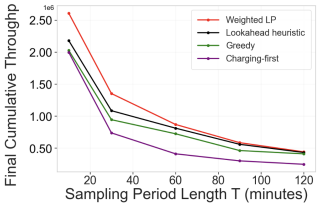


(a) Cumulative Throughput

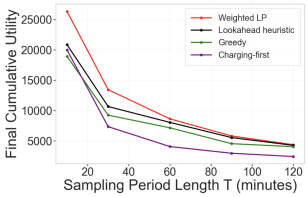


(b) Cumulative Utility

Fig. 2: Results under varying lookahead windows.

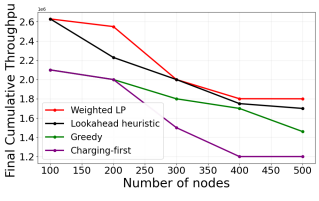


(a) Cumulative Throughput

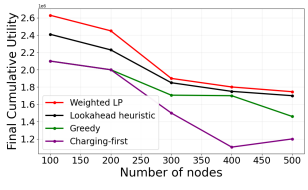


(b) Cumulative Utility

Fig. 3: Results under varying sampling periods.



(a) Cumulative Throughput



(b) Cumulative Utility

Fig. 4: Results under varying number of nodes.

the lookahead heuristic in dense multi-node LoRa networks. Finally, we validate feasibility through a small-scale hardware testbed experiment.

A. ILP vs. LP Comparison Results

1) Setup: We compare the multi-objective ILP and LP formulation, and weighted-sum LP relaxation (Eq. 1) for a single node using a Python implementation with Gurobi [22].The node is powered by a supercapacitor with capacitance $C = 0.5$ F, representative of commercially available LoRa devices. We use a lookahead window of length $L = 5$ periods, a sampling period length of $T = 10$ minutes, and set $\lambda = 0.5$. The executed period index is in minutes and is indexed (divided by T) over a year-long solar energy traces from a real solar panel [23].

2) Results: Fig. 1 shows that the LP solution closely tracks the ILP across both cumulative throughput and cumulative utility, with only a small performance gap and consistent relative ordering between throughput-first and utility-first solutions. The weighted-sum LP consistently lies in between. This demonstrates that the LP relaxation preserves the key scheduling trade-offs of the ILP while significantly reducing computational complexity. This motivates the use of the LP formulation as the primary scheduling approach in large-scale and resource-constrained settings. Based on this observation, we use the weighted-sum LP formulation ($\lambda = 0.5$) for the remainder of the evaluation.

B. Large-Scale Network Simulation Results

1) Setup: We evaluate our approach using the LoRaWAN NS-3 module [24] in networks of up to 500 nodes communicating with a single gateway. Nodes are randomly distributed within a 5 km radius and powered by supercapacitors with capacitance $C = 0.5$ F. We set $\lambda = 0.5$. We use three-month-long solar energy traces with randomized variations to emulate spatial diversity in harvesting conditions.

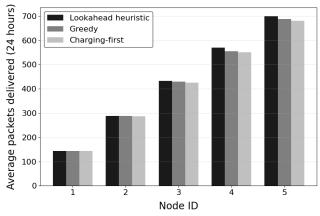
2) Baselines and metrics: To the best of our knowledge, no prior work co-schedules charging and transmission for supercapacitor-powered LPWANs. We therefore compare our approaches-LP and lookahead heuristic against standard LoRaWAN and two ablation baselines: (i) Greedy Heuristic, which transmits packets whenever sufficient energy is available without considering future energy availability or packet utility. It can be viewed similarly to lookahead heuristic with a window of size one period. (ii) Charging-First Heuristic, which prioritizes charging over transmission, does not support partial charging within a period, and resumes transmission only after the supercapacitor is fully charged. We evaluate performance using two primary metrics: cumulative throughput, defined as the total number of successfully delivered packets, and cumulative utility, which captures the aggregated timeliness of delivered data over the evaluation horizon.

3) Results:

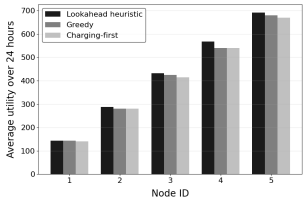
a) Varying lookahead window.: We fix $T = 10$ minutes and 200 nodes, and vary the lookahead window length L from 5 to 25 periods. Fig. 2 shows that both the LP and lookahead heuristic benefit from larger L by better anticipating energy scarcity. The LP achieves peak performance at moderate L ($\approx$ 15), while the heuristic improves monotonically. Both consistently outperform greedy and charging-first baselines, indicating that short to moderate lookahead windows capture most of the achievable gains.

b) Varying sampling period.: We vary the sampling period T from 10 to 120 minutes while fixing $L = 10$ and 200 nodes. As shown in Fig. 3, throughput and utility decrease with increasing T due to reduced scheduling opportunities and higher delay. Nevertheless, the LP consistently outperforms all baselines across all T, while the lookahead heuristic remains within 5–10% of LP performance.

c) Varying number of nodes.: We evaluate scalability by varying no. of nodes from 100 - 500 with $L = 5$ and $T = 10$ min. Fig. 4 shows that although performance degrades with



(a) Average throughput per node



(b) Average per-node utility

Fig. 6: Experiment results.

higher contention, the LP and lookahead heuristic maintain significant advantages over both baselines. The lookahead heuristic remains within 5–8% of LP performance, demonstrating scalability to dense LPWAN deployments.

C. Small Scale Testbed Results

1) Setup: We conduct an experiment to evaluate our approach in an indoor setup as depicted in Fig. 5. We employ five Dragino LoRa shields [25] with Arduino Uno R3 [26] as the LoRa nodes and a USRP B200 [27] as the LoRa gateway. We use a single 915 MHz LoRa channel to mimic a network with a large number of nodes. Regarding other parameters, we use a spreading factor of 7, a bandwidth of 125 kHz, and a coding rate of 4/5 in our experiment. We use a custom python file to model supercapacitor behavior. The experiment was conducted for 24 hours using a random day from the year-long energy trace, and each result is averaged across five runs. We only evaluate the Lookahead heuristic for physical experiments, and compare with greedy and charging-first baselines. We set $L = 5$, and $T = 10$ minutes.

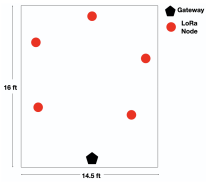


Fig. 5: Indoor Deployment

2) Results: Fig. 6a reports the average number of packets delivered per node over 24 hours. The lookahead heuristic consistently outperforms both greedy and charging-first baselines across all nodes, delivering up to 3–5% more packets, with gains becoming more visible for nodes experiencing higher traffic. Fig. 6b shows the average per-node utility over a 24-hour experiment, averaged across five runs. The lookahead heuristic consistently achieves higher utility than greedy and charging-first baselines, confirming that energy-aware scheduling improves data timeliness in practical deployments.

VI. Conclusion

In this paper, we studied sustainable IoT networks powered by supercapacitors, focusing on the trade-off between energy accumulation and data transmission under limited instantaneous power. We formulated the Charging–Transmission Co-Scheduling problem and proposed an efficient LP relaxation and a lightweight lookahead heuristic for resource-constrained devices, integrated into an LPWAN MAC protocol. Simulations and experiments show 10–25% gains over baselines, closely matching the optimal ILP, with the heuristic within 5–10% of the LP. This work provides an initial step toward resilient and autonomous supercapacitor-powered IoT systems.

Acknowledgment

The work was supported by the US National Science Foundation through grants CNS-2601685, CNS-2602744, and CNS-1948365, and by the US Office of Naval Research through grant N00014-23-1-2151

References

[1] PowerFilm Solar, "Solar power for industrial iot," https://www.powerfilmsolar.com/about-us/the-horizon-blog/2018/07/06/solar-power-for-industrial-iot, 2020.
[2] Ilika, "Low power iot ecosystem," https://www.ilika.com/about-ilika/low-power-io-t-ecosystem, 2020.
[3] S. Naeem, "A review of flexible high-performance supercapacitors for the iot and ai applications," *ETE*, 2023.
[4] A. K. Sultania and J. Famaey, "Energy-aware battery-less bluetooth low energy device prototype powered by ambient light," in *SenSys*, 2021.
[5] M. Gonzalez, P. Xu, R. Dekimpe, M. Schramme, I. Stupia, T. Pirson, and D. Bol, "Technical and ecological limits of 2.45-ghz wireless power transfer for battery-less sensors," *IoT-J*, pp. 15 431–15 442, 2023.
[6] M. Capuzzo, C. Delgado, A. K. Sultania, J. Famaey, and A. Zanella, "Enabling green iot: Energy-aware communication protocols for battery-less lorawan devices," in *MSWiM*, 2021, pp. 95–98.
[7] R. Khamitov, D. Orel, and D. Zorbas, "Predicting solar-harvested energy for resource-constrained iot devices using machine learning," in *DCOSS-IoT*, 2024, pp. 661–668.
[8] F. Maita and L. Maiolo, "Low power wireless sensor network for precision agriculture: a battery-less operation scenario," in *MetroAgriFor*, 2021, pp. 75–79.
[9] S. K. Ram, S. Chourasia, B. B. Das, A. K. Swain, K. Mahapatra, and S. Mohanty, "A solar based power module for battery-less iot sensors towards sustainable smart cities," in *ISVLSI*, 2020, pp. 458–463.
[10] S. Fahmida, V. P. Modekurthy, M. Rahman, A. Saifullah, and M. Brocanelli, "Long-lived lora: Prolonging the lifetime of a lora network," in *ICNP*, 2020.
[11] S. Fahmida, A. Chavan, V. P. Modekurthy, A. Saifullah, and M. Brocanelli, "A battery lifespan-aware protocol for lpwan," in *ICDCS*, 2024.
[12] K. Geissdoerfer and M. Zimmerling, "Bootstrapping battery-free wireless networks: Efficient neighbor discovery and synchronization in the face of intermittency," in *NSDI*. USENIX Association, 2021.
[13] ——, "Learning to communicate effectively between battery-free devices," in *NSDI*. USENIX Association, 2022, pp. 419–435.
[14] Q. Chen, Z. Cai, L. Cheng, F. Wang, and H. Gao, "Joint near-optimal age-based data transmission and energy replenishment scheduling at wireless-powered network edge," in *IEEE INFOCOM*, 2022.
[15] A. I. Petrariu, A. Lavric, E. Coca, and V. Popa, "Hybrid power management system for lora communication using renewable energy," *IoT-J*, 2020.
[16] S. Fahmida, V. P. Modekurthy, D. Ismail, A. Jain, and A. Saifullah, "Real-time communication over lora networks," in *IoTDI*, 2022.
[17] A. Jain, P. Modekurthy, and A. Saifullah, "Control over low-power wide-area networks," in *ICCPS 2024*.
[18] ——, "Real-time scheduling and control co-design over lpwan," *ACM Trans. Cyber-Phys. Syst.*, 2025.
[19] F. A. Kraemer, D. Ammar, A. E. Braten, N. Tamkittikhun, and D. Palma, "Solar energy prediction for constrained iot nodes based on public weather forecasts," in *IoT*, 2017.
[20] A. Jain, M. A. Haque, A. Saifullah, and H. Zhang, "Burst-mac: A mac protocol for handling burst traffic in lora network," in *IEEE Real-Time Systems Symposium (RTSS)*, 2024, pp. 148–160.
[21] S. Fahmida, A. Jain, V. P. Modekurthy, D. Ismail, and A. Saifullah, "Rtpl: A real-time communication protocol for lora network," *ACM Trans. Embed. Comput. Syst.*, 2024.
[22] Gurobi Optimization, LLC, "Gurobi optimizer reference manual," 2022.
[23] "Solar power data for integration studies," https://www.nrel.gov/grid/solar-power-data.html, 2021.
[24] D. Magrin, M. Centenaro, and L. Vangelista, "Performance evaluation of lora networks in a smart city scenario," in *ICC*, 2017.
[25] "Dragino gps/lora shield," https://www.dragino.com/products/lora/item/102-lora-shield.html.
[26] "Arduino uno rev3," https://store-usa.arduino.cc/products/arduino-uno-rev3.
[27] "Ettus research," https://www.ettus.com/product/.