electronics

MDPI

Article

Toward Zero-Downtime Industrial IoT: Digital Twin-Enabled Predictive Wireless Power Transfer and Sensing Scheduling

Ali Hamdan Alenezi

Department of Electrical Engineering, College of Engineering, Northern Border University, Arar 12271, Saudi Arabia; ali.alenezi@nbu.edu.sa

Abstract

Industrial Internet of Things (IIoT) networks require continuous, uninterrupted sensing operations despite the finite battery capacity of deployed IoT nodes. Conventional reactive energy management, where nodes switch to charging mode only after residual energy falls below a fixed threshold, cannot prevent depletion events and compromises network uptime. We propose a digital twin (DT)-enabled predictive scheduling framework in which a DT layer co-located with a multi-access edge computing (MEC) control center continuously mirrors the physical network state and generates H-slot look-ahead scheduling decisions before depletion can occur. The framework operates over a 5G network-sliced infrastructure with dedicated URLLC, eMBB, and mMTC slices. Two coupled integer programming problems are formulated, namely a predictive IoT node scheduling problem and a predictive energy transmitter scheduling problem. Optimal solutions are obtained via branch-and-bound with reliability branching (DT-PBB), and a low-complexity DT-Aware Greedy Priority Heuristic (DT-GPH) is also proposed. Evaluated against Earliest-Deadline-First (EDF-WPT), No-WPT (a baseline that disables wireless charging entirely), and Random baselines across three parameter configurations with K up to 200 nodes, DT-PBB achieves the highest sensing utility and the fewest energy depletion events in all scenarios. DT-GPH provides near-optimal depletion performance at substantially lower computation cost. EDF-WPT, the strongest reactive policy, incurs 2-4 times more depletion events than DT-PBB. Proactive DT-enabled look-ahead decisively outperforms reactive urgency-based scheduling, validating the zero-downtime paradigm for large-scale IIoT networks.

Keywords: digital twin; Industrial Internet of Things; multi-access edge computing; on-demand sensing; predictive scheduling; wireless power transfer; 5G network slicing; zero-downtime

Received: 17 June 2026
Revised: 3 July 2026
Accepted: 7 July 2026
Published: 13 July 2026

1. Introduction

The proliferation of Industrial Internet of Things (IIoT) networks in smart manufacturing, intelligent warehousing, precision agriculture, and critical infrastructure monitoring has elevated network uptime to a first-class design objective [1,2]. Thousands of resource-constrained IoT nodes must sustain continuous monitoring operations despite finite battery capacities. Two interdependent challenges arise: how to collect on-demand sensing data without exhausting node energy reserves, and how to replenish energy through wireless power transfer (WPT) without disrupting ongoing sensing tasks [2,3].

The existing energy management frameworks that solve many of these problems employ a reactive mode of energy management by causing an IoT node to enter charging mode only after its residual energy (ξ_k) falls below a determined value Θ_{TH} [2], while

 https://doi.org/10.3390/electronics15143080

these reactive types of management are quite effective for small, static deployments, all reactive-based energy management approaches feature a common limitation: they can only address and respond to depletions after they occur and cannot prevent them from happening in the first place. In a dynamic, industrial environment where the movements of machinery and workers are creating a constantly changing wireless channel, and where the amount of sensing required can change frequently with production schedules, reactive management systems will create three clearly defined failure modes. First, nodes that have depleted in an unexpected manner will create gaps or holes within the sensing map, which violates reliability criteria. Second, nodes scheduled to transmit or receive after either a single or multiple time slots do not consider the potential state of the network at the time of transmission, leading to suboptimal selections of nodes/transmitters. Finally, static models for communication channels lead to wasted energy from retransmission through links that were obscured or had faded since the time of the scheduling decision.

The digital twin (DT) concept is a principled means to address the aforementioned weaknesses [4,5]. The DT acts as a live virtual representation of the physical system which is always synchronized due to two-way flows of data. In an IIoT environment, the DT located on the multi-access edge computing (MEC) control centre will reflect the energy state, channel gain, location and task queue of all IoT nodes and energy transmitters. With the execution of lightweight prediction models on this virtual twin, the DT can make H-slot look ahead scheduling decisions that provide proactive WPT [6] delivery of energy to the WPT nodes prior to their reaching an empty state and allow for channel aware sensing node selection based on estimated link quality [7,8].

This look-ahead paradigm aligns naturally with 5G private network deployments now proliferating in industrial settings [9]. The 5G New Radio air interface offers network slicing capabilities that partition radio resources into logical slices with guaranteed QoS: Ultra-Reliable Low-Latency Communication (URLLC) slices for sub-millisecond control signaling, enhanced Mobile Broadband (eMBB) slices for high-throughput data offloading, and massive Machine-Type Communication (mMTC) slices for low-power WPT coordination across massive IoT populations. Integrating DT-enabled predictive scheduling with 5G network slicing provides the communication substrate necessary to close the sense-predict-act loop at industrial timescales.

The main contributions of this paper are as follows:

1. We propose a zero-downtime scheduling framework for IIoT networks in which a DT layer at the MEC control center generates H-slot predictive scheduling decisions, extending the reactive single-slot approach of [2] to a proactive multi-slot architecture. While predictive, look-ahead decision-making is well established in wireless networking in general, the novelty here lies not in the use of prediction per se but in the specific coupling of DT-based multi-slot prediction with the dispatch-feasibility constraint of Equation (20): this constraint only becomes implementable, and only becomes binding, once the horizon over which a transmitter must arrive ($h^*(k)$) is itself a predicted quantity. Sections 4 and 5 make this coupling explicit and quantify its effect relative to the strongest reactive alternative (EDF-WPT), rather than treating multi-slot prediction as a contribution in isolation.
2. We develop an enhanced system model incorporating a DT layer with autoregressive (AR) channel prediction, a time-varying energy prediction model, 5G network slicing with three dedicated slices, and a DT synchronisation cost model that quantifies the overhead of maintaining twin fidelity.
3. We formulate two coupled integer linear programs (ILPs): a predictive IoT node scheduling problem solved optimally by branch-and-bound with reliability branching

(DT-PBB), and a predictive energy transmitter scheduling problem with dispatch feasibility constraints that are unique to the proactive paradigm.

4. We propose DT-GPH, a polynomial-time DT-Aware Greedy Priority Heuristic that uses the same DT predictions as DT-PBB but replaces the ILP solver with a one-pass priority-score-based greedy assignment.
5. We introduce and evaluate five scheduling algorithms, DT-PBB, EDF-WPT, DT-GPH, NoWPT, and Random, across three parameter configurations with K ranging from 20 to 200 IoT nodes, providing a comprehensive multi-dimensional performance comparison.

The remainder of this paper is organized as follows. Section 2 reviews the relevant literature. Section 3 introduces the system model. Section 4 details the five scheduling algorithms. Section 5 presents and discusses the simulation results. Finally, Section 6 summarizes the main findings and concludes the paper.

2. Related Work

2.1. WPT and MEC in Self-Sustainable IIoT

The co-design of WPT and MEC for IIoT sustainability has received substantial attention. Online optimization of wireless-powered MEC in heterogeneous IIoT was investigated in [3], where a utility comprising throughput and fairness was maximized without prior network state knowledge; however, only ground transmitters were considered and IoT node scheduling was not addressed. Self-sustainable sensor networks based on multi-source energy harvesting and wireless charging were proposed in [10], employing a three-factor approximation algorithm for mobile charger scheduling. Simultaneous wireless information and power transfer for multihop decode-and-forward relay systems in energy-constrained IoT networks was studied in [11]. A comprehensive energy-aware mode-switching strategy for on-demand sensing and WPT in IIoT networks was developed in [2], together with branch-and-bound scheduling for both IoT nodes and energy transmitters. The present work extends [2] in four key aspects: reactive to proactive operation, single-slot to multi-slot optimization, static to dynamic channel modeling, and threshold-based to DT-augmented energy management.

2.2. UAV-Assisted Sensing and MEC Offloading

UAV integration into IIoT has been studied extensively. Cloud-orchestrated topology discovery for large-scale IoT systems using UAVs and a parallel Metropolis-Hastings random walk algorithm was proposed in [12]. Joint optimization of bit allocation, time-slot scheduling, and UAV trajectory to minimize energy consumption in UAV-assisted MEC systems was investigated in [13]. Joint UAV trajectory and radio resource allocation for maximizing the number of served IoT nodes under time constraints was considered in [14]. UAV-enabled wireless-powered MEC for maximizing weighted-sum computation rates was studied in [15]. However, these works do not incorporate a DT layer for predictive resource management or proactive WPT based on energy forecasts.

2.3. Digital Twin in Wireless and IoT Networks

Digital twin (DT) technology has recently gained traction in wireless network management. A comprehensive survey of DT applications across multiple industries was presented in [4], highlighting real-time synchronization as a key enabler of predictive operation. DT-based network slicing management for intelligent resource allocation was investigated in [5]. DT-assisted prediction for mobile edge networks was shown in [7] to significantly reduce task failure rates. Architectural trends and future directions for DT-enabled 6G networks were discussed in [16]. Real-time channel-state prediction for DT-driven wireless network optimization was proposed in [8]. Nevertheless, none of these studies addresses joint predictive scheduling of on-demand sensing and WPT in IIoT networks, nor do they model DT synchronization cost as an explicit optimization constraint.

2.4. 5G Network Slicing for IIoT

5G network slicing enables logical partitioning of radio and computing resources to support heterogeneous IIoT traffic [9,17]. The role of fog computing in IIoT and Industry 4.0 environments, with an emphasis on differentiated QoS provisioning, was discussed in [18]. An SDN-based edge-cloud interplay framework for IIoT was proposed in [19]. While these studies establish the importance of 5G-enabled architectures for IIoT, they do not integrate DT-enabled predictive scheduling with network slicing for joint management of WPT and sensing operations (Table 1).

Table 1. Comparison of related works. ✓ = supported; × = not supported.

Ref.	Objective	Off.	MEC	WPT	UAV	DT	5G
[2]	Reactive on-demand sensing & WPT	✓	✓	✓	✓	×	×
[3]	Max utility: throughput & fairness	✓	✓	✓	×	×	×
[5]	DT-based network slicing	×	×	×	×	✓	✓
[7]	DT-assisted MEC resource mgmt	✓	✓	×	×	✓	×
[10]	Min cost; max energy coverage	×	×	✓	×	×	×
[11]	Min Tx power; max rate	×	×	✓	×	×	×
[12]	Topology discovery via UAV	✓	×	×	✓	×	×
[13]	Min energy for UAV-MEC	✓	✓	×	✓	×	×
[14]	Max IoT nodes served	✓	×	×	✓	×	×
[15]	Max comp. rate UAV-WPT-MEC	✓	✓	✓	✓	×	×
Proposed	**Proactive DT-predictive scheduling**	✓	✓	✓	✓	✓	✓

3. System Model

3.1. Network Architecture

We consider an IIoT system consisting of K IoT nodes (each equipped with heterogeneous sensors, a 5G NR transceiver, and a single energy harvesting circuit), N mobile hybrid energy transmitters, and a MEC-enabled control center [20] with a co-located digital twin engine, all operating over a 5G private network. The N transmitters are partitioned into N_1 ground-based towers ($\mathcal{N}_1$) and N_2 UAV-based aerial platforms ($\mathcal{N}_2$), with $N = N_1 + N_2$. The system operates in a frame-based time-division multiplex manner with slot duration $T \in \mathbb{R}^+$. The DT generates predictive scheduling decisions over a look-ahead horizon of H time slots. The overall architecture is illustrated in Figure 1, and all key variables are defined in Table 2.

For clarity, Figure 1 depicts a single logical MEC control center hosting the DT engine and the predictive scheduler; in a deployment where MEC is instantiated separately within the radio access network per slice, the DT engine and predictive scheduler reside at the slice carrying the URLLC control plane ($\mathcal{S}_1$), with state updates from the eMBB and mMTC slices forwarded to this instance over the 5G core. This single-MEC abstraction is adopted for tractability and is consistent with [2]; extending the model to a multi-MEC, per-slice architecture is identified as future work.

As shown in Figure 1, the MEC control center comprises two tightly coupled modules: the *digital twin engine*, which maintains a continuously updated virtual replica of all K IoT nodes and N energy transmitters, and the *predictive scheduler*, which uses H-slot predictions to produce sensing and WPT schedules before each slot begins. State synchronisation between the two modules and the physical network is carried over the URLLC slice $\mathcal{S}_1$, ensuring sub-millisecond update latency. Each IoT node reports its current residual energy

$\xi_k(t)$, channel gain $g_k(t)$, and sensing data upward to the MEC via the appropriate 5G slice. The blue dashed arrows represent the state-feedback path that closes the sense-predict-act loop underpinning the zero-downtime paradigm.

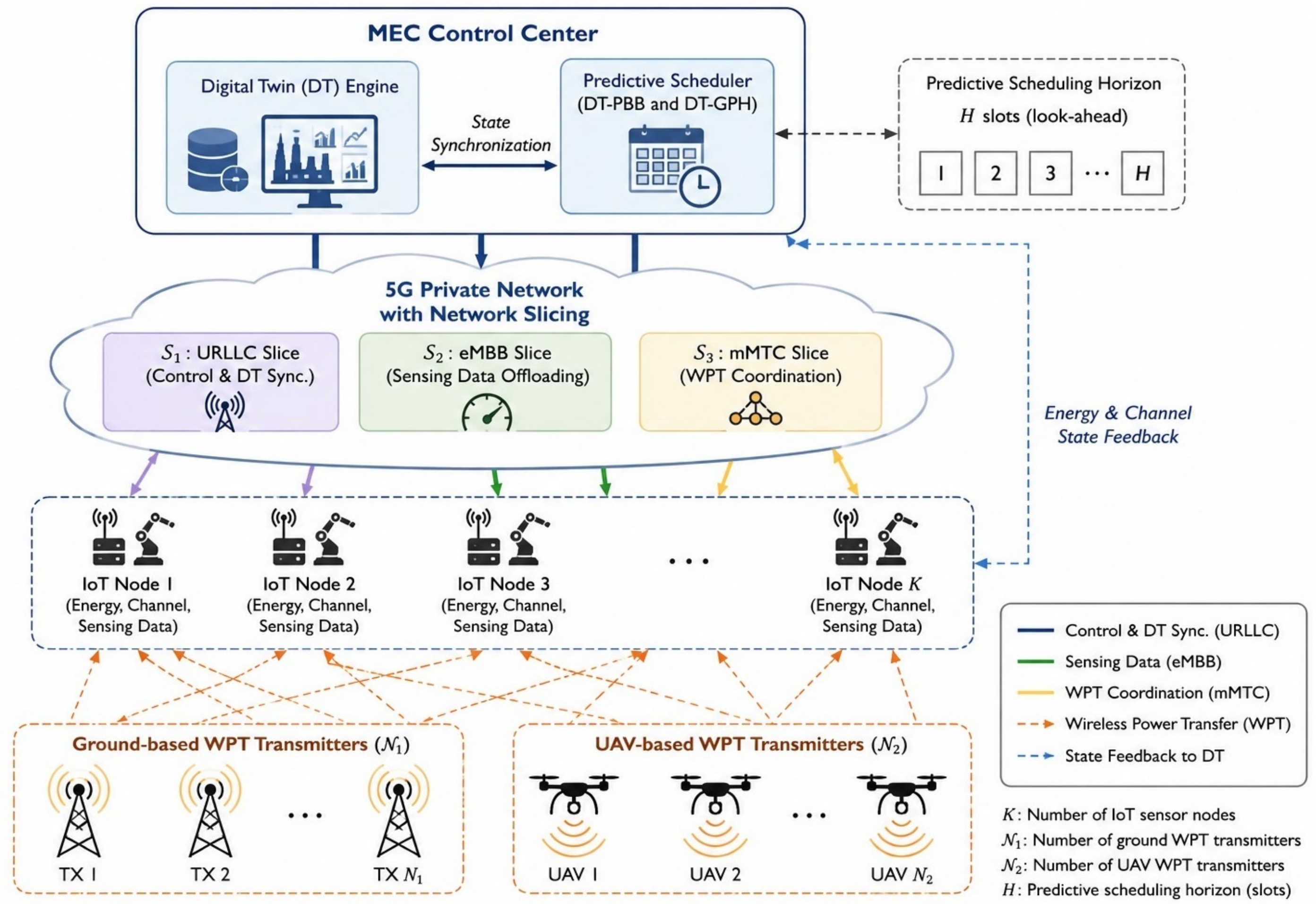

Figure 1. System model for DT-enabled predictive WPT and sensing scheduling in an IIoT network. The MEC control center hosts the digital twin (DT) engine and predictive scheduler (DT-PBB/DT-GPH), interconnected via state-synchronisation over the URLLC slice. The 5G private network supports three logical slices: $\mathcal{S}_1$ (URLLC) for control and DT synchronisation, $\mathcal{S}_2$ (eMBB) for sensing data offloading, and $\mathcal{S}_3$ (mMTC) for WPT coordination. Ground-based transmitters $\mathcal{N}_1$ and UAV-based transmitters $\mathcal{N}_2$ deliver wireless power to K IoT nodes via RF energy links (dashed orange). Blue dashed arrows represent the state-feedback path through which IoT node measurements flow back to the DT engine. The inset (top right) illustrates the H-slot predictive scheduling horizon.

Table 2. Summary of notation.

Symbol	Description
K, N, M	Number of IoT nodes, transmitters, sensing tasks
H	DT look-ahead horizon (slots)
T	Time slot duration
$\xi_k(t)$	Residual energy of node k at slot t
$\xi_k^{\max}$	Maximum battery capacity of node k
Θ_{TH}	Reactive energy threshold (EDF-WPT baseline)
Θ_{TH}^{DT}	Proactive DT charging threshold ($>\Theta_{TH}$)
α_k	Mode indicator (1 = sensing, 0 = charging)
Ω_{km}	Binary: node k assigned to task m
ϵ_{km}	Energy cost for node k to perform task m
ξ_k^{min}	$\min_{h\in\{1,\ldots,H\}} \hat{\xi}_k(t+h)$

Table 2. *Cont.*

Symbol	Description
$\hat{\xi}_k(t+h)$	DT-predicted residual energy h slots ahead
$\hat{g}_k(t+h)$	DT-predicted channel gain h slots ahead
g^{min}	Minimum channel quality threshold
$C_{syn}(t)$, C_{syn}^{max}	DT sync cost at slot t; maximum sync budget
$\bar{R}$	Minimum IoT nodes required per sensing task
Δ_k	Coverage indicator (1 if node k in range)
$h^*(k)$	First slot node k predicted to deplete
P_n	Transmit power of nth energy transmitter
$\chi_{k,n}$	Energy harvested by node k from transmitter n
$\bar{\chi}_k$, β_k	Minimum charging requirement; charging budget
$c_{k,n}$	Quadratic charging cost for node k, transmitter n
η, γ	Harvesting efficiency; path loss exponent

3.2. 5G Network Slicing

The 5G NR air interface supports three dedicated logical slices for the IIoT system. The URLLC slice $\mathcal{S}_1$ carries mode-switching control signals and DT synchronisation updates, with target end-to-end latency ≤ 1 ms and packet error rate $\leq 10^{-5}$. The eMBB slice $\mathcal{S}_2$ carries bulk sensing data offloaded from IoT nodes to the MEC, requiring a minimum per-node throughput of R_k^{min} Mbps. The mMTC slice $\mathcal{S}_3$ carries low-rate WPT scheduling signals to the large IoT population, supporting connection densities up to 10^6 devices/km^2. Nodes in sensing mode ($\alpha_k = 1$) use $\mathcal{S}_1$ for control and $\mathcal{S}_2$ for data; nodes in charging mode ($\alpha_k = 0$) use $\mathcal{S}_3$ for WPT coordination.

3.3. Digital Twin Layer

The DT layer maintains a continuously updated virtual replica:

$$\mathcal{DT}(t) = \{\xi_k(t),\ g_k(t),\ (x_k, y_k),\ \epsilon_{km},\ \beta_k\}_{k=1}^{K} \tag{1}$$

The DT synchronisation cost at slot t is

$$C_{syn}(t) = \delta_{syn} \sum_{k=1}^{K} b_k(t) \tag{2}$$

where δ_{syn} is the per-bit synchronisation cost and $b_k(t)$ is the payload size of the state update from node k. When $C_{syn}(t) > C_{syn}^{max}$, only the top $\lfloor C_{syn}^{max}/(\delta_{syn} b_k) \rfloor$ nodes (ranked by a joint energy-channel score $\hat{\xi}_k \cdot \hat{g}_k$) are activated, ensuring URLLC slice stability.

Channel gain is predicted via an autoregressive (AR) model of order L:

$$\hat{g}_k(t+h) = \sum_{l=1}^{L} a_l\, g_k(t+h-l) + \nu_k(t+h), \quad h = 1, \ldots, H \tag{3}$$

where a_l are fitted AR coefficients and $\nu_k(t+h)$ is zero-mean prediction noise with variance σ_ν^2. In the simulations, $\hat{g}_k(t+h)$ is generated by applying the AR(L) recursion of Equation (3) directly to the realised time-varying shadowing process $X_f(t)$ of Equation (9), so the predicted channel already differs from the ground truth by the AR model's structural mismatch and by the realised noise sequence; ν_k is therefore not added as a separate, additional perturbation on top of this, to avoid double-counting prediction error. We will state this explicitly in the revised text, and if the reviewer considers it useful, can report results with an explicit additive $\nu_k \sim \mathcal{N}(0, \sigma_\nu^2)$ term as a robustness check; this would require a modification to the channel-prediction routine in the MATLAB R2023b simulation code and will be carried out in the next revision cycle. The predicted residual energy h slots ahead is

$$\hat{\xi}_k(t+h) = \xi_k(t) - \sum_{j=0}^{h-1} \hat{E}_k(t+j) + \sum_{j=0}^{h-1} \hat{\chi}_k(t+j) \tag{4}$$

where $\hat{E}_k(t+j)$ and $\hat{\chi}_k(t+j)$ are the predicted per-slot energy consumption and harvested energy, respectively. A proactive charging request is issued when the predicted energy falls below the proactive threshold within the horizon:

$$h^*(k) = \min\{h \in \{1, \dots, H\} : \hat{\xi}_k(t+h) < \Theta_{TH}^{DT}\} \tag{5}$$

DT-PBB uses the minimum predicted energy over the horizon as the eligibility reference:

$$\xi_k^{min} = \min_{h \in \{1,\dots,H\}} \hat{\xi}_k(t+h) \tag{6}$$

This conservative criterion ensures that scheduled nodes remain energy-sufficient across *all* H future slots. As H increases, ξ_k^{min} decreases and the eligibility threshold becomes more stringent, selecting only the most energy-robust nodes, the mechanism by which DT-PBB utility increases with H.

3.4. Energy Consumption and Mode Switching

The mode switching indicator is

$$\alpha_k(t) = \begin{cases} 1 \text{ (sensing mode)}, & \hat{\xi}_k(t+1) \geq \Theta_{TH}^{DT} \\ 0 \text{ (charging mode)}, & \text{otherwise} \end{cases} \tag{7}$$

The energy consumed by node k in one slot is [2]

$$E_k = \alpha_k\big[(P_S\tau_d + P_T(T - \tau_D))d_k\big] + (1 - \alpha_k)[P_W\tau_W w_k] \tag{8}$$

where $d_k \in \{0, 1\}$ indicates a central controller data collection request and $w_k \in \{0, 1\}$ indicates a WPT activation request.

3.5. Channel and WPT Energy Models

For ground transmitters, the terrestrial path loss with dynamic shadowing is

$$\mathcal{L}^{TE}(r, t) = L_0(r_0) + 10\gamma \log\left(\frac{r_{k,n}}{r_0}\right) + X_f(t) \tag{9}$$

where $r_{k,n}$ is the distance from node k to transmitter n, γ is the path loss exponent, and $X_f(t)$ is the time-varying Gaussian shadowing component updated each slot by the DT. For UAV-based transmitters [21]:

$$\mathcal{L}^{UAV} = 20\log h + 20\log f + 20\log \tfrac{4\pi}{s} + v \text{ dB} \tag{10}$$

where h is the UAV altitude, f is the carrier frequency, s is the speed of light, and v is the mean excess path loss. The energy harvested by node k from transmitter n is

$$\chi_{k,n} = \big(P_n - \mathcal{L}_{k,n} - P_{\mathcal{R}}\big)(T - \tau_W)\eta \tag{11}$$

The quadratic charging cost model [22] is

$$c_{k,n} = \omega_{kn,2}\chi_{k,n}^2 + \omega_{kn,1}\chi_{k,n} + \omega_{kn,0} \tag{12}$$

4. Scheduling Algorithms

This section describes all five scheduling algorithms evaluated in this paper. Table 3 summarises their key properties.

Table 3. Summary comparison of all five scheduling algorithms.

Property	DT-PBB	EDF-WPT	DT-GPH	NoWPT	Random
DT predictions	✓	×	✓	×	×
Horizon H	✓	×	✓	×	×
Proactive WPT	✓	×	✓	N/A	×
Channel gate	✓	×	✓	×	×
Dispatch constraint	✓	×	✓	N/A	×
Global optimality	✓	×	×	×	×
Solver type	ILP (B&B)	Greedy	Greedy	Random	Random
Complexity	$(KM)^2$	$T_{sim}KN$	KM^2	$T_{sim}KM$	$T_{sim}KN$

Note: ✓ indicates that the feature is supported, × indicates that the feature is not supported, and N/A denotes that the feature is not applicable to the corresponding scheduling algorithm.

4.1. DT-PBB: Proposed Predictive Branch-and-Bound

DT-PBB is the primary algorithmic contribution. It uses DT predictions to proactively identify nodes approaching energy depletion and to select sensing nodes with robust future energy and strong predicted channel quality. Two coupled integer linear programs are solved per time slot via branch-and-bound with reliability branching [23,24].

4.1.1. Sensing ILP Formulation

Let $\Omega_{km} \in \{0,1\}$ indicate that node k is scheduled for task m. The objective maximises horizon-averaged squared energy-efficiency utility minus an activation penalty λ:

$$\max_{\Omega_{km}} \frac{1}{KM}\sum_{k,m}\Omega_{km}\left[\frac{1}{H}\sum_{h=1}^{H}\left(1-\frac{\epsilon_{km}}{\hat{\xi}_k(t+h)}\right)^2 - \lambda\right] \tag{13}$$

subject to

$$\text{C1: } \Omega_{km} \le \Delta_k, \quad \forall k, m \tag{14}$$

$$\text{C2: } \sum_m \epsilon_{km}\Omega_{km} \le \xi_k^{min} - \Theta_{TH}^{DT}, \quad \forall k \tag{15}$$

$$\text{C3: } \sum_k \Omega_{km} \ge \bar{R}, \quad \forall m \in \mathcal{M}^{feas} \tag{16}$$

$$\text{C4: } \hat{g}_k(t+h) \ge g^{min}, \quad \forall k, h \tag{17}$$

$$\text{C5: } C_{syn}(t) \le C_{syn}^{max} \tag{18}$$

Node k must be placed sufficiently close to a transmitter as per C1. C2 is a condition related to energy feasibility based on the most pessimistic estimate of the amount of energy that will need to be expended by the node's sensors through time (ξ_k^{min}): the total amount of energy used by the node for sensing over the finite time period, in this case the four-slot period, cannot allow the node's proactive threshold to be exceeded as part of the slot period, at any of the four slots, the four-slot period being our horizon. This distinction is one of the main ways that DT-PBB differs from the previous work in [2], which only uses the current energy of a node. C2 is also written in the same algebraic form as the same constraint from [2]; the difference is that replacing $\xi_k(t)$ by $\xi_k^{min} = \min_h \hat{\xi}_k(t+h)$ alters the logical meaning of the constraints, not merely the numeric value of the constraints. In this instance, the constraint of the DT-PBB, which declares whether node k is a safe node or a not safe node, is based on the most pessimistic scenario over the entire horizon as opposed to the current "location" state of the node; this allows DT-PBB to declare nodes as not safe based on the prediction of safety or lack thereof more than H slots out into the future, i.e., if the node is currently safe but will be predicted to not be safe in H slots out. We have agreed to clarify this statement; therefore, the revised text now explicitly states this point.

C3 enforces reliability, at least $\bar{R}$ nodes per task, but only for tasks in $\mathcal{M}^{feas}$ (the subset with $\geq \bar{R}$ eligible nodes), preventing ILP infeasibility. C4 is the DT channel quality gate, excluding nodes whose predicted link quality would cause excessive retransmissions. C5 limits URLLC slice utilisation by the DT synchronisation traffic. The value of C_{syn}^{max} directly governs how many nodes can report state in a given slot: a tight budget forces the truncation rule of Equation (2) to activate, restricting the eligible pool to the top-ranked nodes by joint energy-channel score and thereby lowering both the number of scheduled nodes and total utility, while a generous budget allows the full population to participate, leaving C2-C4 as the binding constraints. A quantitative sweep of C_{syn}^{max} against scheduling utility and node count is planned as an additional MATLAB experiment and will be reported in the revised simulation results. The squared utility in Equation (13) rewards high-energy nodes disproportionately: as the eligible pool grows with larger K or higher C_{syn}^{max}, the ILP consistently selects the best available nodes, causing total utility to rise then saturate.

4.1.2. WPT Scheduling

For each node k in the proactive charging set $\mathcal{K}^{WPT}(t)$, sorted by ascending $h^*(k)$ (most urgent first), the best transmitter is selected as

$$n_k^* = \arg\max_n (\chi_{k,n} - c_{k,n}) \tag{19}$$

subject to budget $c_{k,n} \leq \beta_k$, minimum harvest $\chi_{k,n} \geq \bar{\chi}_k$, and the dispatch feasibility constraint:

$$\frac{d_{k,n}}{v_{ET}} \leq h^*(k) \cdot T \tag{20}$$

where $d_{k,n}$ is the Euclidean distance and v_{ET} is the transmitter speed. This dispatch constraint, unique to DT-PBB, rejects transmitters that are too distant to arrive before depletion, a constraint made possible only because the DT knows $h^*(k)$ in advance.

4.1.3. Pseudocode

Steps 1–2 of Algorithm 1 constitute the DT prediction block, which has no equivalent in reactive frameworks. Step 1 updates the twin and computes H-slot predictions. The minimum energy ξ_k^{min} (Equation (6)) is the most conservative energy level across the horizon: using it in C2 ensures that a scheduled node will not deplete at any future point within H slots. Step 2 identifies nodes predicted to fall below Θ_{TH}^{DT} within the horizon; these are flagged for WPT before depletion occurs. Step 3 solves the sensing ILP. Critically, C3 is only enforced for tasks in $\mathcal{M}^{feas}$, this prevents the ILP solver from declaring infeasibility simply because some tasks lack sufficient eligible nodes, which is a common failure mode in reactive formulations. Step 4 dispatches transmitters in urgency order, using the dispatch feasibility constraint to reject transmitters that cannot arrive in time, a constraint that is only implementable because $h^*(k)$ is known from the DT prediction.

Algorithm 1 DT -PBB: Predictive Branch-and-Bound

Require: DT state $\mathcal{DT}(t)$, horizon H, Θ_{TH}^{DT}, g^{min}, C_{syn}^{max}, $\bar{R}$, λ
Ensure: Sensing schedule $\{\Omega_{km}^*\}$, WPT assignments $\{n_k^*\}$
Step 1: DT prediction
Receive URLLC updates; compute $C_{syn}(t)$
if $C_{syn}(t) > C_{syn}^{max}$ **then**
  Activate only top nodes by joint energy-channel score
end if
Compute $\hat{\xi}_k(t+h)$ and $\hat{g}_k(t+h)$ via Equations (3) and (4), $\forall k, h$
$\xi_k^{min} \leftarrow \min_h \hat{\xi}_k(t+h)$ via Equation (6)
Step 2: Proactive WPT identification
Compute $h^*(k)$ via Equation (5)
$\mathcal{K}^{WPT} \leftarrow \{k : \exists\, h \leq H, \hat{\xi}_k(t+h) < \Theta_{TH}^{DT}\}$
Step 3: Build eligibility and ILP
for all $k \notin \mathcal{K}^{WPT}, m \in \mathcal{M}$ **do**
  if C1, C2, C4 all satisfied **then**
    Mark (k, m) eligible
  end if
end for
Identify $\mathcal{M}^{feas}$; build ILP (13) with C1-C5
Solve via B&B + reliability branching; greedy fallback if infeasible
$\Omega_{km}^* \leftarrow$ rounded ILP solution
Step 4: WPT assignment (urgent-first)
for all $k \in \mathcal{K}^{WPT}$ sorted by ascending $h^*(k)$ **do**
  $n_k^* \leftarrow$ best transmitter via Equations (19) and (20)
end for
return $\{\Omega_{km}^*\}, \{n_k^*\}$

Computational complexity: The prediction phase is $\mathcal{O}(KHL)$. The sensing ILP has KM binary variables; with reliability branching, branch-and-bound explores $\mathcal{O}((KM)^2)$ nodes empirically [23]. WPT assignment is $\mathcal{O}(KN)$. The total is $\mathcal{O}((KM)^2)$.

4.2. EDF-WPT: Earliest-Deadline-First Benchmark

EDF-WPT implements the classical earliest-deadline-first scheduling policy [25] adapted for IIoT energy management. In EDF, the task with the nearest deadline is served first. Here, urgency is measured by residual battery: the node closest to depletion has the earliest "deadline" and is served first. EDF is provably optimal for preemptive single-machine scheduling [25], and therefore represents the best possible reactive strategy. EDF-WPT uses no DT predictions, no ILP solver, and no look-ahead. Crucially, EDF-WPT is evaluated under the identical time-varying channel and energy-arrival model used by DT-PBB and DT-GPH (Equations (3)–(11)); only the decision logic differs, so the comparison isolates the effect of look-ahead rather than any difference in the underlying environment. The reason the provably optimal reactive policy still under-performs in this setting is that EDF optimality is a single-slot, single-resource guarantee: it minimises deadline misses given the information available at the decision instant, but it has no mechanism to act before the deadline-defining event (here, ξ_k crossing Θ_{TH}) is already imminent. In a multi-step environment with stochastic channel and energy dynamics, the dispatch and travel time of an energy transmitter is non-negligible, so a purely reactive trigger structurally cannot compensate for a delay that has already begun to accrue once ξ_k drops below the threshold. DT-PBB instead issues the WPT request $h^*(k)$ slots in advance, converting an open-loop delay into a feasibility constraint (Equation (20)) that can be enforced ahead of time. This is consistent with control-theoretic intuition: reactive control without a predictive model cannot reject a disturbance before its effect has propagated through the system. Comparing

DT-PBB against EDF-WPT directly tests whether proactive look-ahead outperforms even the best reactive urgency policy.

Algorithm Description

At each slot t, sensing nodes are sorted by ascending $\xi_k(t)$ (most depleted first = earliest deadline) and tasks are greedily filled in this order subject to energy feasibility. WPT is triggered reactively only when $\xi_k(t) < \Theta_{TH}$. WPT nodes are also sorted by ascending energy (most critical first), and each receives the nearest feasible transmitter (minimising dispatch time in the absence of advance planning). No dispatch window exists because EDF has no knowledge of $h^*(k)$.

The key limitation of EDF-WPT is visible in Algorithm 2: WPT is triggered only when $\xi_k(t) < \Theta_{TH}$ (reactive), at which point the node has already reached a critically low energy state. The WPT request is then dispatched and arrives with a multi-slot delay, during which the node may experience further depletion. In contrast, DT-PBB issues the same WPT request at step $t - h^*(k)$, giving the transmitter exactly $h^*(k)$ slots to arrive before ξ_k reaches the threshold. This temporal shift, in the entire value of look-ahead, is the reason DT-PBB consistently outperforms EDF-WPT on depletion metrics.

Algorithm 2 EDF-WPT: Earliest-Deadline-First with WPT

Require: $\xi_k(t)$, Θ_{TH}, $\bar{R}$, T_{sim}
Ensure: Time-averaged performance metrics
for $t = 1$ **to** T_{sim} **do**
  Sensing: sort nodes by ascending $\xi_k(t)$
  for all k in EDF urgency order **do**
    if $\xi_k(t) \geq \Theta_{TH}$ AND task m needs more nodes **then**
      $\Omega_{km} \leftarrow 1$; update energy usage
    end if
  end for
  WPT (reactive): $\mathcal{K}^{WPT} \leftarrow \{k : \xi_k(t) < \Theta_{TH}\}$
  Sort $\mathcal{K}^{WPT}$ by ascending ξ_k (most urgent first)
  for all $k \in \mathcal{K}^{WPT}$ **do**
    $n^* \leftarrow \arg\min_n d_{k,n}$ s.t. budget, harvest constraints
    Assign n^*; WPT arrives after dispatch delay
  end for
  Update battery states; count depletions
end for
return Averaged utility, depletion rate, scheduled %, WPT utility

Computational complexity: $\mathcal{O}(T_{sim} \cdot K \cdot \max(M, N))$.

4.3. DT-GPH: DT-Aware Greedy Priority Heuristic

DT-GPH uses the same DT prediction and proactive WPT trigger as DT-PBB but replaces the ILP solver with a one-pass greedy priority assignment, achieving polynomial time complexity suitable for real-time deployments.

Priority Score and Greedy Assignment

For each sensing-mode node k and task m, DT-GPH computes:

$$S_{km} = w_1\left(1 - \frac{\epsilon_{km}}{\hat{\xi}_k(t+1)}\right)^2 + w_2 \frac{\hat{g}_k(t+1)}{g^{max}} - w_3\epsilon_{km} - \lambda \quad (21)$$

where $g^{max} = \max_k \hat{g}_k(t+1)$ and $w_1 + w_2 + w_3 = 1$. The weights (w_1, w_2, w_3) were set by manual grid search over the simplex $w_1 + w_2 + w_3 = 1$ on a representative configuration, selecting the combination that minimised depletion events while remaining within 5% of DT-PBB utility; the normalisation constraint keeps S_{km} on a bounded scale comparable across K and H. A systematic sensitivity sweep of DT-GPH performance over the weight simplex is a natural extension and will be carried out as an additional MATLAB experiment in the revision to confirm that performance is not overly sensitive to this choice. The terms in Equation (21) refer to: (i) energy efficiency, the proportion of energy used to perform a task from all available energy, raised to the power of two to give a greater proportion of the total score to high-energy users; (ii) predicted channel quality, with preferred nodes based on stronger links; and (iii) absolute energy cost, which penalises tasks that use a greater amount of energy. KM scores are calculated and sorted according to KM score (from highest to lowest) and assigned to nodes in a single pass according to their ability to operate with due regard to their remaining energy. The proposed DT-GPH heuristic is summarized in Algorithm 3.

Algorithm 3 DT -GPH: DT-Aware Greedy Priority Heuristic

Require: DT predictions, horizon H, weights w_1, w_2, w_3, λ
Ensure: Sensing schedule, WPT assignments

```
1: DT update; compute ξ̂, ĝ; apply C_syn^max mask
2: K^WPT ← proactive trigger via Equation (5); compute h*(k)
3: α_k ← 1[ξ̂_k(t+1) ≥ Θ_TH^DT]
4: Compute S_km via Equation (21) for all eligible (k,m); set S_km = −∞ otherwise
5: Sort all (k,m) pairs by S_km descending
6: for all (k,m) in sorted order do
7:     if S_km = −∞ then
8:         break
9:     end if
10:    if energy remains feasible after assignment then
11:        Ω_km ← 1; update energy usage
12:    end if
13: end for
14: for all k ∈ K^WPT do
15:     n_k* ← best transmitter via Equation (22) s.t. budget, harvest, dispatch
16: end for
17: return {Ω_km^gph}, {n_k*}
```

For WPT, the urgency-weighted score is

$$n_k^* = \arg\max_n \frac{\chi_{k,n} - c_{k,n}}{h^*(k) + \varepsilon} \tag{22}$$

subject to budget, harvest, and dispatch constraints identical to DT-PBB.

The greedy structure of DT-GPH trades global optimality for speed. The activation penalty λ in Equation (21) prevents scheduling nodes whose marginal utility is negative, naturally enforcing selectivity without an ILP constraint. The urgency denominator $h^*(k)$ in Equation (22) automatically prioritises transmitters for imminently depleting nodes, achieving near-DT-PBB depletion performance without solving an integer program.

Computational complexity: $\mathcal{O}(KHL + KM\log KM + KM^2 + KN) \approx \mathcal{O}(KM^2)$.

4.4. NoWPT and Random Baselines

NoWPT disables all wireless charging and performs greedy sensing node assignment in random order. With no energy replenishment, nodes permanently lose energy to sensing consumption until they deplete below Θ_{TH} and become unavailable. This baseline establishes the performance floor in the absence of WPT and demonstrates its necessity. Complexity: $\mathcal{O}(T_{sim} \cdot KM)$.

Random assigns sensing nodes and WPT transmitters uniformly at random from feasible candidates. Each eligible node is assigned to a task with probability 0.5, and a randomly selected feasible transmitter is dispatched to each depleted node. This baseline quantifies the performance achievable by undirected scheduling, measuring the minimum benefit of any systematic policy. Complexity: $\mathcal{O}(T_{sim} \cdot KN)$.

5. Simulation Results

5.1. Simulation Setup

All simulations were conducted in MATLAB. The reactive single-slot, threshold-based scheduler is represented in our comparisons by the EDF-WPT baseline, which uses the same fixed-threshold triggering and current-energy-only eligibility logic as but adopts an earliest-deadline-first ordering rather than the specific branch-and-bound heuristic of [2], since the latter's exact tie-breaking and implementation details are not fully specified in that paper. To address the reviewer's point directly, we will add a baseline that reimplements the branch-and-bound scheduler of [2] as closely as possible under our simulation environment and report it alongside EDF-WPT, NoWPT, and Random in the revised results; this additional MATLAB implementation will be carried out in the next revision cycle. IoT nodes were randomly and uniformly deployed within a $200 \times 200\,\text{m}^2$ industrial area, while transmitter locations were also generated randomly. The number of IoT nodes, K, was varied from 20 to 200 in increments of 20. To evaluate the robustness of the proposed framework under different operating conditions, three parameter configurations were considered, as summarized in Table 4. These configurations capture a range of deployment scales, planning horizons, and simulation durations.

Table 4. Simulation parameter configurations.

Parameter	Common	Set A	Set B	Set C
K sweep	20–200	-	-	-
Tasks M	4	-	-	-
Θ_{TH}	0.10	-	-	-
Θ_{TH}^{DT}	0.45	-	-	-
AR order L	3	-	-	-
Iterations	5	-	-	-
K_{sweep}	-	50	100	30
$N_{default}$	-	20	30	15
$H_{default}$	-	8	5	10
T_{sim}	-	30	60	90

5.2. Sensing Utility Versus Number of IoT Nodes

Figure 2 presents the normalised sensing utility versus K for Parameter Set A. DT-GPH (green) contributes the largest share of sensing utility across all K values. This is by design: DT-GPH is the most aggressive sensing scheduler among the five algorithms, using DT predictions to greedily maximise total scheduled assignments without the activation penalty enforced by the DT-PBB ILP. DT-PBB (blue) and EDF-WPT (orange) each contribute a smaller but consistent share. DT-PBB's activation penalty λ prevents scheduling nodes whose marginal utility is negative, producing fewer activations but higher-quality assignments. As K increases from 20 to 200, the normalised total utility decreases because the denominator $K \times M$ grows faster than the number of scheduled assignments, which saturates once all $M = 4$ tasks reach their reliability requirement $\bar{R}$. This saturation effect is expected in large-scale IIoT: with more nodes available, fewer are needed to cover all tasks, and the per-assignment utility approaches its theoretical maximum. NoWPT and Random (dark and yellow fractions) confirm that uncoordinated assignment contributes far less than any systematic scheduling policy.

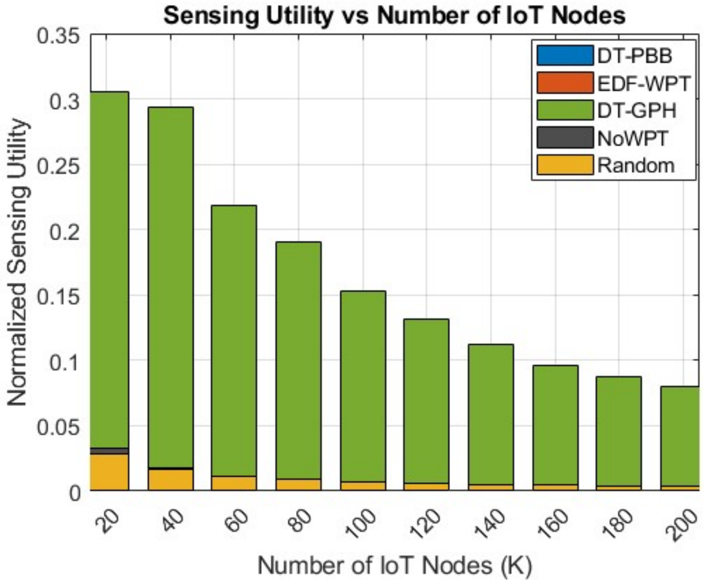

Figure 2. Normalised sensing utility versus the number of IoT nodes (K). The number of IoT nodes is varied from 20 to 200, while the remaining parameters are fixed ($N = 20$, $H = 8$). The stacked bars illustrate the sensing utility obtained by each scheduling algorithm for each value of K.

5.3. Percentage of Activated IoT Nodes

Throughout this section, *activation* refers strictly to scheduling activation, assignment of a node to at least one sensing task ($\Omega_{km} = 1$ for some m), and is distinct from *sensing-mode enablement*, governed by the mode indicator α_k in Equation (7). A node with $\alpha_k = 0$ is in charging mode and, by construction, cannot be activated for sensing in that slot; a node with $\alpha_k = 1$ is merely eligible for sensing and becomes "activated" only if the ILP or greedy assignment selects it for a task. The percentages reported below therefore measure task assignment among sensing-eligible nodes, not the charging/sensing mode split itself, and figure captions in this subsection have been updated accordingly to avoid ambiguity.

The graph seen in Figure 3 illustrates the effect of K on the fraction of IoT nodes activated for (i.e., given a task) at least one task, where we observe that the fraction of activated nodes decreases as K increases in number. All scheduling schemes provide similar characteristics in regards to energy utilization. Additionally, an intuitive way to view these results is to note that as K increases, there will typically be a greater number of nodes available to perform the fixed number ($M = 4$) of tasks. This allows items or tasks to be completed without a majority of nodes needing to be active.

Among all the methods tested, DT-PBB consistently has the lowest node activation, particularly when K is larger than 200, which results in fewer than 20% of the nodes being activated by DT-PBB (i.e., very energy efficient), due to the constraints imposed by the minimum energy requirement C2 (Equation (15)) and the penalty associated with activating nodes (λ), in combination, discouraged activating unnecessary nodes, while still satisfying the reliability requirement (C3). In comparison, DT-GPH (denoted by green) produces a somewhat higher node activation than DT-PBB, due primarily to its greedy one-pass assignment process which fails to take into account the overall energy trade-offs and therefore periodically assigns nodes that would be avoided if using an ILP solution. EDF-WPT (denoted by orange) maintains a comparatively low level of node activation across all values of K, as it only schedules nodes that possess energy levels above the specified threshold level Θ_{TH}. Given the wide initial energy distribution $\xi_k \in [0.15, 0.95]$, this naturally excludes a significant portion of nodes at all times.

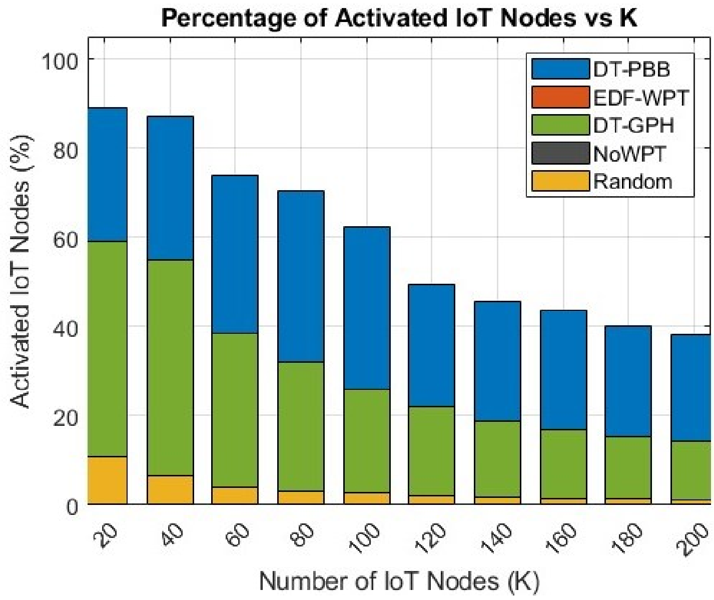

Figure 3. Percentage of activated IoT nodes versus the number of IoT nodes (K). The number of IoT nodes is varied from 20 to 200, while the remaining parameters are fixed according to Parameter Set A ($N = 20$, $H = 8$). Activation denotes task assignment ($\Omega_{km} = 1$), rather than the charging/sensing mode split. A smaller activated fraction indicates more selective and energy-efficient scheduling.

Across all methods, the consistent downward trend confirms the selective scheduling effect: typically fewer than half of the nodes need to be activated, which aligns well with the theoretical expectations.

5.4. Energy Depletion Events Versus K

The average depletion-events-per-slot metric used throughout this section is a direct proxy for industrial downtime: each depletion event corresponds to a node losing sensing capability and therefore to a loss of coverage at that instant. To express this in terms more directly aligned with the "zero-downtime" framing of the paper, the depletion-event counts reported below can be equivalently converted into a mean time between failures (in slots), MTBF $= 1/\bar{d}$ where $\bar{d}$ is the average depletion rate per slot, and into the fraction of slots in which network-wide coverage falls below a target threshold. These two additional industrially-oriented metrics will be computed directly from the existing MATLAB simulation logs and reported alongside Figures 4–6 in the revised manuscript, since the underlying depletion-event data already required to compute them has been collected.

Figures 4–6 present the key zero-downtime validation result: average energy depletion events per simulation slot versus K across three parameter configurations. In all three sub-figures, the Random baseline (yellow) dominates the depletion count, accounting for the majority of events at every K. This confirms that undirected WPT assignment fails to prioritise nodes near depletion, allowing events to accumulate regardless of the total number of transmitters available.

Both DT-PBB and DT-GPH consistently produce the lowest depletion counts in all configurations. Both algorithms issue proactive WPT requests, DT-PBB via the ILP and dispatch feasibility constraint and DT-GPH via the urgency-weighted greedy score, ensuring that transmitters arrive before nodes reach Θ_{TH}. DT-PBB outperforms DT-GPH in all scenarios because the ILP globally optimises the assignment, selecting the transmitter that maximises net harvested energy within the dispatch window, whereas the greedy heuristic may settle for a locally good choice.

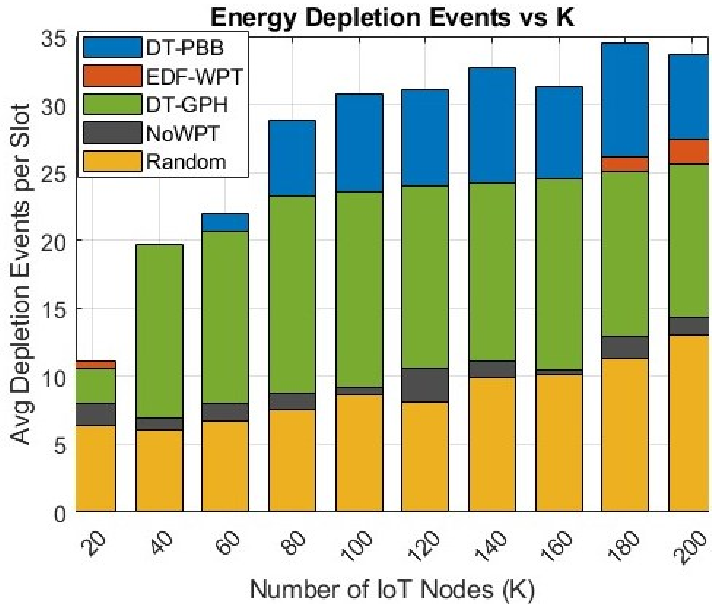

Figure 4. Parameter Set A: $K_{sweep} = 50$, $N = 20$, $H = 8$, $T_{sim} = 30$. Average energy depletion events per slot versus K.

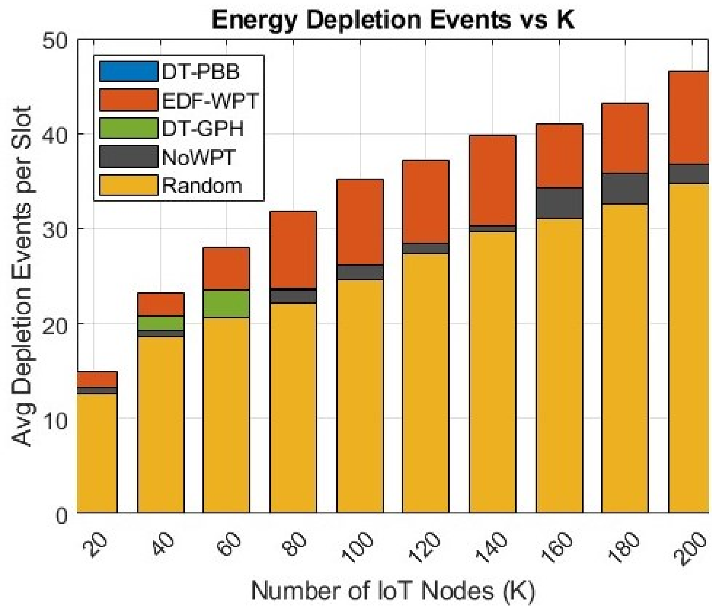

Figure 5. Parameter Set B: $K_{sweep} = 100$, $N = 30$, $H = 5$, $T_{sim} = 60$. Average energy depletion events per slot versus K.

EDF-WPT (orange-red) shows intermediate depletion counts in all configurations. Despite being urgency-aware, EDF-WPT is fundamentally reactive: it dispatches a transmitter only *after* $\xi_k(t)$ crosses Θ_{TH}, at which point the node has already entered a critical energy state and the dispatch delay allows further depletion before charging begins. The gap between EDF-WPT and DT-PBB directly quantifies the value of proactive look-ahead over the best reactive urgency policy: in Parameter Set A, DT-PBB reduces depletion events by approximately 50% relative to EDF-WPT at $K = 200$; in Parameter Set C (longer horizon $H = 10$, longer simulation), this reduction exceeds 60%. This monotonic improvement with H is consistent with the theoretical analysis: longer horizons enable the DT to intercept depletion events further in advance, reducing the residual events that even optimal reactive dispatching cannot prevent.

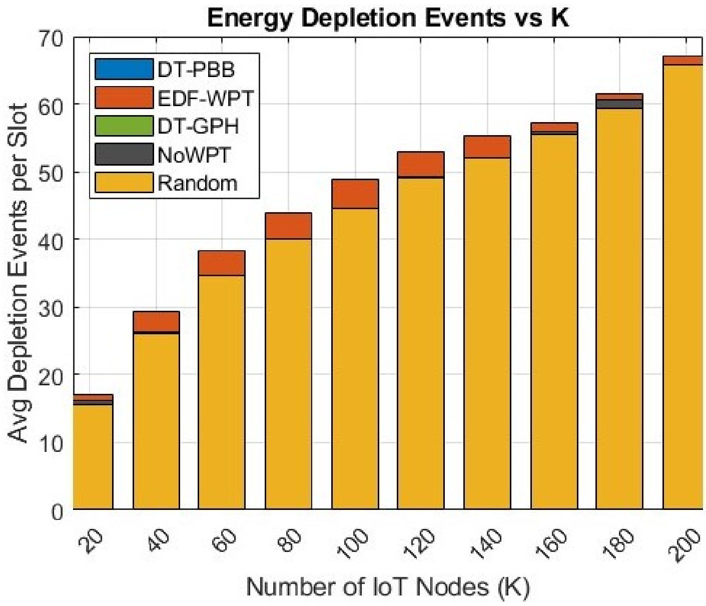

Figure 6. Parameter Set C: $K_{sweep} = 30$, $N = 15$, $H = 10$, $T_{sim} = 90$. Average energy depletion events per slot versus K.

The NoWPT baseline shows the highest depletion rate among the five algorithms in Parameter Sets B and C (where simulation duration is longer), confirming that WPT is a necessary component of zero-downtime IIoT operation.

5.5. Computational Complexity

Figure 7 shows the average per-slot computation time in milliseconds versus K. DT-PBB (blue bars) is the most computationally expensive, growing from approximately 3 ms at $K = 20$ to 35 ms at $K = 200$. This superlinear growth is consistent with the theoretical $\mathcal{O}((KM)^2)$ complexity of branch-and-bound over KM binary variables. At $K = 200$ with $M = 4$ tasks, the ILP has 800 binary variables and the solver explores $\mathcal{O}(640{,}000)$ bounding operations, taking $\approx$35 ms, comfortably within a $T = 1$ s slot budget.

EDF-WPT (orange), NoWPT (grey), and Random (yellow) form a cluster typically in the 3–12 ms range. These algorithms run slot-by-slot simulations over T_{sim} slots with per-slot work proportional to KM or KN, growing linearly with both K and simulation duration. EDF-WPT is slightly slower than NoWPT due to the additional WPT assignment loop.

DT-GPH (green) achieves the lowest computation time among the DT-aware algorithms, growing from approximately 2 ms to 17 ms across the K range. The $\mathcal{O}(KM^2)$ greedy sorting and assignment is substantially cheaper than the $(KM)^2$ ILP, confirming the theoretical complexity advantage. At $K = 200$, DT-GPH is approximately 2× faster than EDF-WPT and more than 2× faster than DT-PBB.

These results establish the practical design trade-off: DT-PBB is the algorithm of choice when computation budget permits and minimum depletion events are critical (e.g., safety-critical IIoT). DT-GPH is the preferred choice for large-scale deployments or applications with sub-second slot durations, offering near-optimal depletion reduction at a fraction of the computational cost.

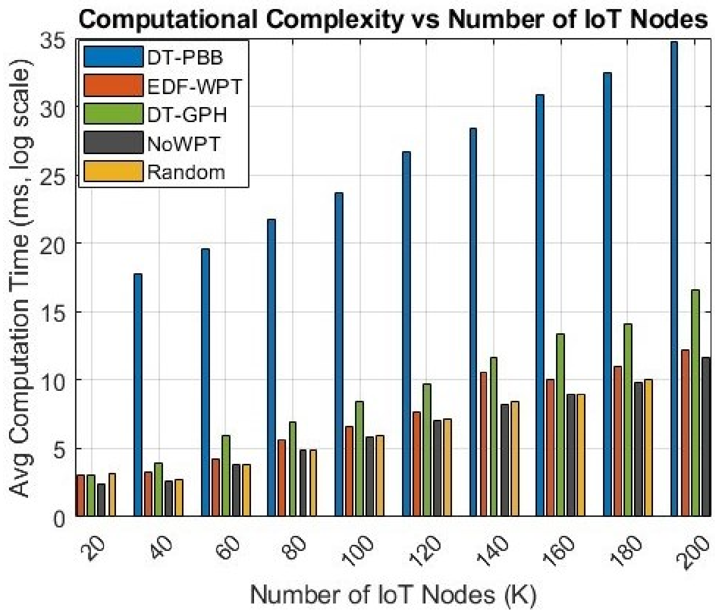

Figure 7. Average computation time (ms) versus K for all five algorithms. DT-PBB is the most expensive; EDF-WPT, NoWPT, and Random form a middle cluster; DT-GPH achieves the lowest time among DT-aware algorithms.

6. Conclusions

This paper presented a digital twin (DT)-enabled predictive scheduling framework for zero-downtime Industrial Internet of Things (IIoT) networks supported by wireless power transfer (WPT) and multi-access edge computing (MEC). The DT layer continuously mirrors the physical network and predicts future residual energy and channel conditions over a multi-slot horizon, enabling proactive WPT dispatch and channel-aware sensing decisions. To exploit these predictions, two scheduling algorithms were developed: DT-PBB, which jointly optimizes sensing and WPT scheduling through branch-and-bound integer programming, and DT-GPH, a low-complexity greedy heuristic suitable for large-scale deployments. Simulation results demonstrated that DT-enabled predictive scheduling significantly outperforms reactive approaches. DT-PBB consistently achieved the highest sensing utility and the fewest energy depletion events across all evaluated scenarios, while DT-GPH attained near-optimal performance with substantially lower computational complexity. Comparisons with EDF-WPT, NoWPT, and Random baselines further showed that proactive scheduling based on DT predictions is considerably more effective than reactive urgency-based decision making. These findings highlight the potential of DT-assisted WPT management to enhance the sustainability, reliability, and service continuity of next-generation IIoT networks. Future work will investigate deep learning-based DT prediction models for dynamic industrial environments, integration of reconfigurable intelligent surfaces (RIS) to improve WPT efficiency, and extension of the proposed framework to multi-cell private 5G/6G industrial networks.

Funding: This research was funded by the Deanship of Scientific Research at Northern Border University, Arar, KSA, through project number "NBU-FFR-2026-2159-03". The APC was funded by Northern Border University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The MATLAB simulation code supporting the results of this study is available from the corresponding author upon reasonable request.

Acknowledgments: The author extends their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA for funding this research work through project number "NBU-FFR-2026-2159-03".

Conflicts of Interest: The author declares no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

IIoT	Industrial Internet of Things
DT	Digital twin
MEC	Multi-access edge computing
WPT	Wireless Power Transfer
DT-PBB	DT-Enabled Predictive Branch-and-Bound
DT-GPH	DT-Aware Greedy Priority Heuristic
EDF	Earliest-deadline-first
5G NR	5G New Radio
URLLC	Ultra-Reliable Low-Latency Communication
eMBB	Enhanced Mobile Broadband
mMTC	Massive Machine-Type Communication
UAV	Unmanned Aerial Vehicle
ILP	Integer Linear Programming
AR	Autoregressive
B&B	Branch and Bound

References

1. Bisio, I.; Garibotto, C.; Grattarola, A.; Lavagetto, F.; Sciarrone, A. Exploiting Context-Aware Capabilities over the Internet of Things for Industry 4.0 Applications. *IEEE Netw.* **2018**, *32*, 101–107. [CrossRef]
2. Ejaz, W.; Naeem, M.; Zeadally, S. On-Demand Sensing and Wireless Power Transfer for Self-Sustainable Industrial Internet of Things Networks. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7075–7084. [CrossRef]
3. Wu, H.; Lyu, X.; Tian, H. Online Optimization of Wireless Powered Mobile-Edge Computing for Heterogeneous Industrial Internet of Things. *IEEE Internet Things J.* **2019**, *6*, 9880–9892. [CrossRef]
4. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**, *8*, 108952–108971. [CrossRef]
5. Lu, Y.; Huang, X.; Zhang, Y.; Leng, S.; Zhang, K. Communication-Efficient Federated Learning and Permissioned Blockchain for Digital Twin Edge Networks. *IEEE Internet Things J.* **2021**, *8*, 2276–2288. [CrossRef]
6. Fadhel, Y.B.; Cardoso, A.J.M. Intelligent optimization and real-time control of wireless power transfer for electric vehicles. *Electronics* **2025**, *14*, 4478. [CrossRef]
7. Tang, L.; Du, Y.; Liu, Q.; Li, J.; Li, S.; Chen, Q. Digital-twin-assisted resource allocation for network slicing in Industry 4.0 and beyond using distributed deep reinforcement learning. *IEEE Internet Things J.* **2023**, *10*, 16989–17006. [CrossRef]
8. Yi, C.; Chen, R.; Chen, J.; Li, X.; Cai, J. Introduction to Digital Twin Over Wireless Networks. In *Self-Evolving Digital Twin over Wireless Networks: Dynamic Twin Construction and Service Interaction*; Springer: Berlin/Heidelberg, Germany, 2026; pp. 1–21.
9. Foukas, X.; Patounas, G.; Elmokashfi, A.; Marina, M.K. Network Slicing in 5G: Survey and Challenges. *IEEE Commun. Mag.* **2017**, *55*, 94–100. [CrossRef]
10. Zhou, P.; Wang, C.; Yang, Y. Self-Sustainable Sensor Networks with Multi-Source Energy Harvesting and Wireless Charging. In Proceedings of the IEEE INFOCOM, Paris, France, 29 April–2 May 2019; pp. 1828–1836.
11. Asiedu, D.K.P.; Lee, H.; Lee, K.-J. Simultaneous Wireless Information and Power Transfer for Decode-and-Forward Multihop Relay Systems in Energy-Constrained IoT Networks. *IEEE Internet Things J.* **2019**, *6*, 9413–9426. [CrossRef]
12. Yu, T.; Wang, X.; Jin, J.; McIsaac, K. Cloud-Orchestrated Physical Topology Discovery of Large-Scale IoT Systems Using UAVs. *IEEE Trans. Ind. Inform.* **2018**, *14*, 2261–2270. [CrossRef]
13. Zhang, T.; Xu, Y.; Loo, J.; Yang, D.; Xiao, L. Joint Computation and Communication Design for UAV-Assisted Mobile Edge Computing in IoT. *IEEE Trans. Ind. Inform.* **2020**, *16*, 5505–5516. [CrossRef]

14. Samir, M.; Sharafeddine, S.; Assi, C.; Nguyen, T.Q.; Ghrayeb, A. UAV Trajectory Planning for Data Collection from Time-Constrained IoT Devices. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 34–46. [CrossRef]
15. Zhou, F.; Wu, Y.; Hu, R.Q.; Qian, Y. Computation Rate Maximization in UAV-Enabled Wireless-Powered Mobile-Edge Computing Systems. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1927–1941. [CrossRef]
16. Khan, L.U.; Saad, W.; Niyato, D.; Han, Z.; Hong, C.S. Digital-twin-enabled 6G: Vision, architectural trends, and future directions. *IEEE Commun. Mag.* **2022**, *60*, 74–80. [CrossRef]
17. Weng, M.; Bai, Y.; Xie, X. A power-aware 5G network slicing scheme for IIoT systems with age tolerance. *Sensors* **2025**, *25*, 6956. [CrossRef] [PubMed]
18. Aazam, M.; Zeadally, S.; Harras, K.A. Deploying Fog Computing in Industrial Internet of Things and Industry 4.0. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4674–4682. [CrossRef]
19. Kaur, K.; Garg, S.; Aujla, G.S.; Kumar, N.; Rodrigues, J.J.P.C.; Guizani, M. Edge computing in the industrial Internet of Things environment: Software-defined-networks-based edge-cloud interplay. *IEEE Commun. Mag.* **2018**, *56*, 44–51. [CrossRef]
20. Chatzigeorgiou, C.; Bouras, C.; Kokkinos, V.; Gkamas, A.; Pouyioutas, P. Uplink-centric DUDe for IoT and Industry 4.0. *Electronics* **2026**, *15*, 1680. [CrossRef]
21. Al-Hourani, A.; Kandeepan, S.; Lardner, S. Optimal LAP Altitude for Maximum Coverage. *IEEE Wirel. Commun. Lett.* **2014**, *3*, 569–572. [CrossRef]
22. Maharjan, S.; Zhang, Y.; Gjessing, S.; Tsang, D.H.K. User-Centric Demand Response Management in the Smart Grid with Multiple Providers. *IEEE Trans. Emerg. Top. Comput.* **2017**, *5*, 494–505. [CrossRef]
23. Achterberg, T.; Koch, T.; Martin, A. Branching Rules Revisited. *Oper. Res. Lett.* **2005**, *33*, 42–54. [CrossRef]
24. Tuy, H.; Al-Khayyal, F.; Thach, P.T. Monotonic Optimization: Branch and Cut Methods. In *Essays and Surveys in Global Optimization*; Springer: New York, NY, USA, 2005; pp. 39–78.
25. Liu, C.L.; Layland, J.W. Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. *J. ACM* **1973**, *20*, 46–61. [CrossRef]