Abstract—Communication system design for wireless networked control systems (WNCSs) requires strict timing, reliability and lifetime guarantees despite limited battery resources and the non-idealities introduced by wireless networking such as delays. In this paper, we introduce radio frequency (RF) energy harvesting paradigm into WNCS framework for the first time in the literature. We study the optimal power control, energy harvesting and scheduling problem with the objective of providing maximum level of adaptivity under periodicity, delay and reliability requirements. We show that the power allocation problem is separable from the scheduling problem at optimality and provide the exact expression for optimal power control. The scheduling problem is then formulated as a mixed integer linear programming (MILP) problem and proven to be NP-Hard. For the scheduling, we propose polynomial-time heuristic algorithms motivated by the analogy between scheduling sensor nodes with energy harvesting requirements over time units and jobs with sequence dependent setup times on identical machines. We prove the theoretical worst-case bound for the performance of these heuristics. We show via extensive simulations that the proposed algorithms perform close-to-optimal and significantly better than Earliest Deadline First (EDF) algorithm in terms of adaptivity, delay, reliability and average runtime.

Index Terms—Wireless networked control systems, RF energy harvesting, adaptivity, power control, scheduling.

I. INTRODUCTION

Wireless networked control systems (WNCSs) are spatially distributed systems in which sensors, actuators and controllers connect via a wireless network [2]. The usage of wireless communication in control systems results in lower cost, flexible network architectures, easy installation and agility. Consequently, WNCSs have been used in a broad range of applications, including industrial automation [3], building automation [4], unmanned aerial vehicles [5], automated highway [6] and smart grid [7], with standardization efforts of industrial organizations, such as International Society of Automation (ISA) [8] and Highway Addressable Remote Transducer (HART) [9]. Conventionally, WNCS nodes are assumed to operate on battery, therefore, most research efforts focus on energy saving mechanisms to prolong their lifetime. The time criticality of the information sent from the sensor nodes makes the battery dependent lifetime a real bottleneck, since the depleted batteries need to be replaced while keeping the system operational. Incorporating energy harvesting (EH) capability to the system can overcome this issue by prolonging the lifetime of the sensor nodes without any replacement. The sensor nodes may generate energy from natural sources, such as sun, vibration and pressure [10], [11]. However, the dependency of the harvested energy on the environmental conditions, so, the randomness in the amount and arrival time of the harvested energy may not satisfy the time-critical data delivery requirement in such networks. On the other hand, EH technologies based on inductive or magnetic resonant coupling are practically infeasible for such type of networks due to their large size and requirement of accurate calibration and alignment of the coils. Radio frequency (RF) energy harvesting networks (RF-EHN) stand out as a viable solution to support the strict delay and reliability requirements of WNCS applications due to its full control on the transferred energy, small form factor and easy installation by using the dedicated RF resources [12], [13]. The heterogeneous requirements of the sensor nodes in a WNCS necessitate the design of robust and reliable scheduling algorithms to achieve a certain level of performance despite the unfavorable network conditions [14]. The time triggered sensor nodes periodically generate data packets at a preset frequency, the scheduling algorithms for such type of transmissions should exploit this periodic transmission pattern and distribute the data transmissions as uniformly as possible over time to accommodate maximum transmissions. On the other hand, the event triggered sensor nodes generate the data based on an aperiodic event. The uniform distribution of the periodic transmissions not only helps to accommodate such event triggered transmission with minimum delay in the unallocated parts of the schedule, but also allows to add new sensors with reduced overhead and delay.

The previous scheduling frameworks proposed for WNCSs have been studied with three main objectives: to provide low deterministic end-to-end delay for real-time traffic [15]–[18], to adopt real-time scheduling algorithms used for the scheduling of both periodic and aperiodic controller tasks [14], [19], [20] and to focus on optimization addressing the trade-off between energy consumption and performance of control systems [21]–[23]. First, the scheduling algorithms designed for low deterministic end-to-end delay of real-time traffic are mainly based on recent communication standards such as WirelessHART [9] and ISA-100.11a [8]. However, these algorithms are designed for large mesh network and do not consider the periodic nature of data transmission. Second, the real time scheduling algorithms used for the scheduling of periodic controller tasks, such as Earliest Deadline First (EDF) and deadline monotonic (DM) [24], are adopted for the
scheduling of the periodic transmission of the sensor nodes in WNCs [19], [20]. However, as they allocate the transmission slot as soon as the packets arrive, these algorithms are not robust to the changes and imperfections of the wireless communication, thus, may not guarantee timely packet delivery in practical networks. Finally, the optimization problem formulations for WNCs aims to maximize the end-to-end reliability under delay guarantee [21], minimize the energy consumption of the sensor nodes under packet loss probability and delay constraints [22] and maximize control system performance by minimizing tracking error under network capacity and delay constraints [23]. However, these formulations either neglect the maximum transmit power of the sensor nodes and length of the packet as a variable [21] or ignore the dependency of the energy consumption on the transmission power of the sensor nodes [22], [23]. More recently, the optimization problems have been extended to include the dependency of the energy consumption on the transmission power of the nodes while exploiting the periodicity of the sensor node transmissions in WNCs [14], [20], [25]. These studies focus on maximizing adaptivity to packet losses, additional messaging events and changes in the requirements of sensors and network topology. The adaptivity is attained by distributing packet transmissions as uniformly as possible over time. Nonetheless, these works assume battery powered sensor nodes, and do not incorporate energy harvesting into the system.

As a plausible solution to the battery replacement, the RF-EHNs have been studied extensively for single-hop, multi-hop and multi-antenna configurations in the literature [13]. The formulations mostly aim to achieve the throughput maximization [26], total transmission time minimization [27], or fairness among the nodes considering multiple input multiple output (MIMO) system [28], under the energy causality constraint without any scheduling. On the other hand, scheduling is performed to maximize the sum throughput of the network [29], [30] or to minimize the schedule length [31]. However, none of these studies consider periodic data transmissions. To the best of our knowledge, only [32] and [33] consider periodic data transmission with the aim of maximizing the sum and average throughput of the system. However, this throughput maximization objective cannot be applied to our problem, in which the objective is to achieve maximum adaptivity through the uniform distributions of the periodic transmissions of time-triggered sensors while allocating the random transmissions of event-triggered sensors and packet transmissions with minimum delay.

In this paper, we propose a framework that incorporates RF energy harvesting paradigm into WNCs. By exploiting the periodic nature of node transmissions inherent in WNCs and accounting for the aperiodic transmissions, we study the optimal power control and scheduling algorithm with the objective of attaining maximum level of adaptivity while meeting the packet generation period, transmission delay, reliability and energy causality of the sensor nodes in a WNC. Our proposed scheduling framework provides adaptivity to the transmission of the unscheduled event triggered packets, additional messaging events and changes in the network topology by uniformly distributing the packet transmissions of the periodic transmissions of the sensors with predetermined topology and packet generation pattern over time. Moreover, the dependence of the energy consumption on the transmission power has been incorporated into the framework together with the energy causality constraint, i.e., the consumed energy should be less than or equal to the available energy.

The original contributions of the paper are listed below.

- We propose a novel framework for WNCs employing the RFEH mechanism for the sensor nodes to provide maximum level of adaptivity while satisfying the packet generation period, energy causality, maximum transmit power and reliability constraints.
- The formulated optimization problem is a mixed integer non-linear programming (MINLP) problem, which is hard to solve. Following the optimality analysis of the problem, we determine the optimal power for the sensor nodes and energy harvesting slot for each subframe. By exploiting the optimal power and energy harvesting slot, we transform the MINLP into a mixed integer linear programming (MILP) problem for the scheduling, which is easier to handle.
- For the scheduling problem, we prove the NP-hardness of the problem and propose two polynomial time heuristic algorithms, namely Periodic List Scheduling (PLS) Algorithm and Periodic Alternative List Scheduling (PALS) Algorithm, each of which is preferable depending on the performance measure of choice. Furthermore, we analytically derive the asymptotic worst-case bounds for the performance of the heuristics.
- Through extensive simulations, the performance of the proposed heuristics is illustrated compared to the optimal solution and EDF algorithm in terms of adaptivity, delay and computational cost for various network sizes and environments.

The rest of the paper is organized as follows. Section II describes the system model and assumptions used throughout the paper. Section III presents the objective function, constraints and formulation of the optimization problem. Section IV provides the analysis of the optimality conditions, exact expression of the optimal power allocation, and the formulation and NP-hardness of the scheduling problem. Sections V and VI present the heuristic scheduling algorithms PLS and PALS, respectively, and analyze their worst case performance bounds. Section VII gives the performance evaluation of the proposed algorithms. Finally, Section VIII concludes the paper.

II. SYSTEM MODEL AND ASSUMPTIONS

The system model and assumptions are detailed as follows. The WNC consists of $L$ wireless sensor nodes and a certain number of controllers and actuators, as illustrated in Fig. 1. The controllers transmit RF energy for the sensor nodes and by using the harvested energy, sensor nodes transmit their data to the controller on a wireless channel. Each controller controls a certain physical part of the plant and can only receive a single packet at a time. Upon reception of the sampled information, controllers compute and forward control commands to the actuators for the proper operation of the
Many controllers are collocated with the actuators, e.g., in ventilation, heat and air conditioning systems, because the control command is critical [34]. Therefore, we assume that they are connected through wires and actuators receive the commands successfully [35]. Although the actuators and sensors are located at the same plant, they are not necessarily collocated within the same plant, preventing sensor nodes from exploiting the power supply of the actuators. Incorporating RF energy harvesting to the sensor nodes not only reduces the part and maintenance cost of the wires but also leads to a more scalable network, facilitating the installation of new sensors without requiring any extra wires in the network.

Sensor nodes can be time triggered or event triggered. Time triggered sensor nodes periodically sense the data from a real-time physical system or plant, and transmit them to their controller. On the other hand, event triggered sensor nodes are activated based on the occurrence of a particular event, e.g., fire, humidity or gas leakage, and send their data about the event to the controller. The packets generated by the event triggered sensors are then allocated to the earliest available slot in the constructed schedule, i.e., an unallocated slot with length greater than or equal to the transmission time of the event triggered sensor. It is important to note that intelligent scheduling of the time-triggered sensor nodes can minimize the delay of the event triggered sensors by distributing the unallocated time as uniformly as possible.

The packet generation period of the time-triggered sensor node $i$ is fixed, which is denoted by $T_i$, for $i \in \{1, 2, \ldots, L\}$. Each sensor node $i$ prepares the data packet every $T_i$ seconds and transmits it within its allocated time slot. We assume that the data requirement of the sensor nodes is small enough such that they transmit only one packet within their corresponding packet generation period [36], [37]. The sensor nodes are assumed to have implicit delay tolerances, i.e., their transmission delay requirement is equal to their packet generation period. This implies that the nodes should complete their transmission before the next data generation. On the other hand, event-triggered sensors only generate packet upon a predetermined event acting as a trigger. These sensors wait until the first unallocated part of the schedule is large enough to accommodate their packet and then perform the transmission.

We assume that each sensor node directly communicates with the corresponding controller, i.e., it is a single-hop network. The proposed framework can be extended for multi-hop networks and concurrent transmissions by adding routing constraints and modifying the transmission model in the optimization problem, respectively. To simplify the problem in the first step and focus on incorporating EH paradigm into the power control and scheduling of WNCSSs, these extensions are out of the scope of this paper and subject to future studies. The terms node and link both indicate the connection between a sensor node and its controller, and as each node is connected to a single controller, both terms are used interchangeably. The single antenna network is considered to reduce the complexity of system and algorithm design at the first step of the study.

We consider Time Division Multiple Access (TDMA) as medium access control protocol. The total time in which the system remains operational is divided into fixed length frames and each frame is further divided into fixed length subframes, where frame length $F = \max_{i \in \{1, 2, \ldots, L\}} (T_i)$ and subframe length $S = \min_{i \in \{1, 2, \ldots, L\}} (T_i)$. The subframe length is chosen equal to the minimum packet generation period, since a smaller subframe length may not generate any feasible schedule if the length of the time slots is too large to fit in one subframe or may result in a very short unallocated time duration at the end of the subframe without allowing the allocation of additional messaging, which contradicts with the objective of this study. On the other hand, a subframe length larger than the minimum packet generation period provides no advantage, resulting in less uniform distribution of the packet transmissions. Each subframe consists of an EH slot and multiple data transmission slots.

One of the controllers is selected as a central controller, i.e., system manager in ISA 100.11a or central gateway and network manager in WirelessHART [9]. In the initialization of the network, all the controllers collect channel state information (CSI) from the sensor nodes, send it to the central controller, then, the central controller runs the scheduling and resource allocation algorithms based on the collected CSI information and includes the resulting scheduling and resource allocation decisions in the beacon transmitted to the controllers. The controllers continuously monitor the received power and the packet error rate of the sensor nodes, and inform the central controller in the case of significant changes in the channel conditions of the time-triggered sensors and new arrival or departure of a node. The central controller then either restarts the CSI collection and decision update similar to the initialization of the network, or makes some adjustments in the schedule, depending on the amount of changes and packet error rate in the network. The controllers can also send additional beacons for the scheduling of the retransmissions. In the case where there is no change in the schedule and resource allocation, the beacon is only used for time synchronization. Otherwise, the beacon also contains...
updated scheduling and resource allocation decisions. The coordination overhead is due to 1) the collection of CSI and transmission of scheduling and resource allocation decisions in the initialization of the network and in case of changes in network topology, with complexity $O(QL)$ where $Q$ is the rate of topology change, which is negligible for low mobility networks, and 2) transmission of the beacons for synchronization, with complexity $O(M)$ for $M$ subframes, which is negligible since the beacon transmission is much smaller than the whole information transmission. Since the main focus of this paper is the scheduling and resource allocation algorithm design, the mechanisms for topology discovery and synchronization are out of scope of this paper and can be found in [38].

The nodes are assumed to be equipped with a rechargeable battery with low energy storage capacity and high discharge rate, e.g., super-capacitors. Thus, all the harvested energy is assumed to be consumed or discharged within the corresponding subframe [39]. Super capacitors have a wide variety of applications, in hybrid energy storage systems, automotive industry, wireless sensor networks [27], thus, the corresponding model is widely used in the literature, e.g. [27], [40]. We assume a half duplex system in which nodes only harvest energy until they start transmitting. The energy received by node $i$ from the controller is dependent on the channel coefficient $c_i$ for the channel from the controller to sensor node $i$, the controller transmit power $p_c$ and energy harvesting circuitry model. The total harvested energy $e_{ij}$ by node $i$ in subframe $j$ is given by

$$e_{ij} = h(p_c,c_i)p_i^h,$$

where $p_i^h$ is the length of the EH slot in subframe $j$ [26], and $h(p_c,c_i)$ is the function relating the received power, i.e., $p_c c_i$, to the energy harvesting rate at sensor node $i$. In the case linear EH model is used [41]-[43], $h(p_c,c_i) = \eta_i p_c c_i$, where $\eta_i$ is the energy harvesting efficiency, which is assumed constant over the receive power of interest. In the case non-linear EH model is used, $h(p_c,c_i) = \eta_i p_c / (\Omega_i - 1)$, where $\Omega_i = (1 + e^{a_i b_i})^{-1}$ is a constant to ensure zero-input zero-output response, and $\Psi_i = M_i (1 + e^{-a_i (p_c - b_i)})^{-1}$ is the logistic function related to node $i$, the parameters $M_i$, $a_i$ and $b_i$ are positive constants determined by curve fitting [44].

The packet generation period of a sensor is either a multiple or an aliquot of other packet generation periods, which can be given as a constraint to the control applications [25]. The choice of such a pattern allows the illustration of the robust scheduling idea more clearly and the determination of the worst case bounds for the proposed heuristic algorithms. However, the algorithms proposed in the article can be easily applied for any arbitrary integral set of packet generation periods without loss of generality.

Due to its low power and processing requirements, we use a constant rate model [45], in which each node $i$ transmits successfully at constant rate $r$ if link $i$ satisfies a fixed Signal-to-Noise-Ratio (SNR) threshold, given by

$$\frac{p_i g_i}{N_0} \geq \beta_i,$$

where $p_i$ is the transmit power of the node $i$, $\beta_i$ is the SNR level required to achieve the constant rate $r$, $g_i$ is the channel gain from the sensor node to the controller, and $N_0$ is the background noise power. The data transmission slot $t_i$ allocated to node $i$, is the ratio of the packet length of sensor $i$, denoted by $L_i$, to the data rate $r$, i.e., $t_i = L_i / r$, and is same for all subframes.

We assume block fading channels, i.e., $c_i$ and $g_i$ remains unchanged within the scheduling frame, considering that the sensors and controllers are mostly static within the plant. This is commonly assumed for minimum length scheduling in EH networks [31], [45], [46]. The CSI can be obtained through a training process and fed back to the controller at the start of each scheduling frame. The time and energy spent in this process can be considered negligible compared to the time and energy cost of data transmissions, for a static or low mobility network [47].

### III. DESCRIPTION OF OPTIMIZATION PROBLEM

#### A. Objective Function of Optimization Problem

The objective of the joint power control, energy harvesting and scheduling problem is to provide maximum level of adaptivity. In the following, we summarize adaptivity metric used in [20], [25]. The total active length of subframe $i$, denoted by $a_i$, is originally defined as the sum of the duration of all the data transmission slots allocated to a subframe. This definition is extended here to additionally include the EH slot. The objective of maximizing adaptivity is then quantified as minimizing the maximum total active length over all $M$ subframes, where $M = F/S$. The adaptivity of the schedule allows the network to accommodate the unscheduled and unplanned transmissions in the network, such as a transmission of an event-triggered sensor or an inclusion of a new sensor node, as illustrated through an example next.

Fig. 2 depicts an example for the allocation of three sensor nodes. The packet generation periods for sensor nodes 1, 2 and 3 are given as $T_1 = 1\text{ms}$ and $T_2 = T_3 = 2\text{ms}$, respectively. The data transmission slot lengths for sensor nodes 1, 2 and 3 are $t_1 = t_2 = 0.2\text{ms}$ and $t_3 = 0.3\text{ms}$, respectively. Subframe and frame lengths are then set to the minimum packet generation period, i.e., $S = 1\text{ms}$, and the maximum packet generation period, i.e., $F = 2\text{ms}$, respectively. In this allocation, the length of the EH slots for subframe 1 and subframe 2 are given as $t_1^h = 0.2\text{ms}$ and $t_2^h = 0.1\text{ms}$ for EDF and $t_1^h = t_2^h = 0.2\text{ms}$ for uniform schedule in accordance with the energy harvesting requirements of the allocated nodes.
respectively. The length of EH slots are different for EDF and uniform schedule because of different allocations in each subframe. The schedule given in Fig. 2(a) is generated by using the EDF scheduling policy, assuming that all packets are generated at the beginning of the scheduling frame of duration 2\(ms\) and the deadline of the packets is equal to their periodic packet generation requirement. Another feasible schedule that uniformly distributes the allocation of time slots over time is illustrated in Fig. 2(b). We now compare the performance of these two schedules in terms of adaptivity as follows:

- Suppose a new sensor node 4 with parameters \(T_4 = 1\ms\) and \(t_4 = 0.3\ms\), which requires 0.1\ms\ of energy harvesting time, needs to be allocated in the schedule. The allocation in Fig. 2(b) can accommodate this new sensor, whereas the schedule in Fig. 2(a) cannot.
- Suppose that an additional packet of a 0.3\ms\ time slot length requiring 0.1\ms\ of energy harvesting is generated by an event-triggered sensor node at the beginning of the scheduling frame. Then, in the schedule in Fig. 2(a), the time slot of the event-triggered packet can be allocated with a delay of 1.3\ms, whereas in the schedule in Fig. 2(b), the time slot can be allocated with a delay of 0.6\ms.

As shown through this example, the uniform distribution of the slot allocation over time allows to accommodate the new arrivals of time-triggered and event-triggered sensors. Therefore, an adaptive schedule should distribute node transmissions as uniformly as possible over time, which is mathematically formulated as minimizing the maximum total active length over all subframes.

B. Constraints of Optimization Problem

The optimization problem considers the constraints for energy causality, maximum transmit power of the sensor nodes, transmission delay and periodic data generation for individual sensor nodes. The purpose of the energy causality constraint is to make sure that the consumed energy is less than or equal to the available energy. The maximum transmit power constraint puts a limitation on the transmission power of the sensors. The transmission delay constraint guarantees the transmission of the generated packet before the next packet generation. The periodic data generation constraint requires that the data generation of all sensors occur periodically, separated by \(T_i\) time unit for node \(i\).

Let \(s_i\) be the ratio of packet generation period \(T_i\) to subframe length \(S\), i.e., \(s_i = T_i/S\). We now show that the periodic data generation and transmission delay requirements can be achieved by the allocation of fixed-length time slot once in every \(s_i\) subframes.

**Lemma 1:** The periodic data generation and delay requirements are satisfied by allocating a time slot of length \(t_i\) once in every \(s_i\) subframes to node \(i\), where \(i \in \{1, 2, \cdots, L\}\).

**Proof:** The proof of this lemma is similar to that of Lemma 1 in [20], with two main differences: 1) This lemma incorporates both the periodic data generation and delay requirements due to the assumption of implicit delay tolerance, as stated in Section II; whereas the lemma in [20] embeds only the periodic data generation into the objective function by allowing the delay requirement to be less than the packet generation period. 2) [20] assumes that nodes are arranged in a specific order within the subframe; whereas there is no such restriction in this lemma. Let the \(k\)th data generation of sensor \(i\) occur at the beginning of subframe \(j\) at time \(t^{(k)}_{i,j} = S(j-1)\). The \((k+1)\)th data generation occurs \(T_i\) units after the \(k\)th data generation at time \(t^{(k+1)}_{i,j} = S(j-1) + T_i = S(j-1) + Ss_i = S(j-1 + s_i)\), due to the periodic data generation requirement. Moreover, the \(k\)th data transmission time, defined as \(t^{(k)}_{i,j}\), must take place before the \((k+1)\)th data generation, due to the delay requirement. Therefore, we have \(S(j-1) < t^{(k)}_{i,j} < t^{(k+1)}_{i,j} + t_i = S(j-1 + s_i)\), which corresponds to allocating a time slot of length \(t_i\) for node \(i\) between subframes \(j\) and \((j + s_i)\), i.e., once in every \(s_i\) subframes.

C. Formulation of Optimization Problem

The optimal energy harvesting, scheduling and power control problem is mathematically formulated as follows:

\[
\text{minimize} \quad \frac{\max_{j \in \{1, 2, \cdots, M\}} \sum_{i=1}^{L} z_{ij}t_i + t^h_j}{(3a)} \\
\text{subject to} \quad \sum_{j=ks_i+1}^{L} z_{ij} = 1, \quad k \in \{0, 1, \cdots, M \} - 1, \quad i \in \{1, 2, \cdots, L\} (3b) \\
\sum_{j=1}^{L} z_{ij}t_i + t^h_j \leq S, \quad j \in \{1, 2, \cdots, M\} (3c) \\
z_{ij}t_i p_i \leq h(p_c \xi_j) t^h_j, \quad i \in \{1, 2, \cdots, L\}, \quad j \in \{1, 2, \cdots, M\} (3d) \\
p_i \leq p_{\text{max}}, \quad i \in \{1, 2, \cdots, L\} (3e) \\
\frac{p_i Q_i}{N_0} \geq \beta_i, \quad i \in \{1, 2, \cdots, L\} (3f)
\]

where \(p_{\text{max}}\) is the maximum allowed transmission power for the sensor nodes; \(z_{ij}\) is an integer variable that takes value 1 if sensor \(i\) is allocated to subframe \(j\) and 0 otherwise.

Eqn. (3a) aims to minimize the maximum total active length of all subframes. Eqns. (3b) represent the periodic packet generation and delay requirements. Eqn. (3c) is a feasibility condition that states that the maximum total active length over all subframes cannot exceed the subframe length. Eqn. (3d) represent the energy causality constraint, i.e., consumed energy should not exceed the available energy. Eqn. (3e) states that the transmission power of link \(i\) cannot exceed maximum allowed transmit power. Eqn. (3f) represents the SNR requirement for constant rate transmission with no concurrent transmissions. The variables of the problem are given as follows: 1) \(z_{ij}, i \in \{1, 2, \cdots, L\}, j \in \{1, 2, \cdots, M\}\) for scheduling; 2) \(p_i, i \in \{1, 2, \cdots, L\}\) for power allocation; and 3) \(t^h_j, j \in \{1, 2, \cdots, M\}\) for energy harvesting slot allocation.

D. Accommodating Network Changes

There are four possible changes that can occur in the network:
• A new periodic user joins the network.
• An existing periodic user leaves the network.
• An event triggered user arrives.
• A significant change occurs in the channel conditions.

These possible changes are addressed in the optimization framework as follows:

• If a new periodic user arrives, the central controller re-evaluates the schedule with the number of users equal to \( L + 1 \) instead of \( L \). The frame length and sub-frame length will be updated accordingly.
• If an existing periodic user leaves the network, the central controller re-evaluates the schedule with the number of users equal to \( L - 1 \) rather than \( L \).
• In case of an arrival of a new event triggered node, that node is accommodated in the first available unallocated part of the subframe. The motivation of uniformly distributing the users in all the subframes facilitates the accommodation of such new arrivals of the event-triggered sensor nodes.
• For the channel gain changes, we are considering a block fading channel, which is a very common assumption in the wireless networks and any such change will be addressed in the following frame transmissions.

In the following section, we show that the optimal power and energy harvesting time can be evaluated separately from the scheduling problem, which simplifies the optimization problem.

IV. ANALYSIS OF OPTIMALITY CONDITIONS

A. Optimal Power Allocation

Let us assume that \( \frac{\beta_i N_0}{g_i} \leq p_{\text{max}} \) is satisfied to have a non-empty feasible region for the problem. Then the lower and upper bounds of \( p_i \) value are, \( \frac{\beta_i N_0}{g_i} \) and \( p_{\text{max}} \), using Eqns. (3f) and (3e), respectively. Since the energy consumption so the lower bound for \( t_j^h \) decreases as \( p_i \) decreases by Eqn. (3d), the optimal value of power \( p_i \) is equal to its lower bound, as given by

\[
p_i^* = \frac{\beta_i N_0}{g_i}, \quad i \in \{1, 2, \ldots, L\}
\]  

B. Energy Harvesting Slot Allocation

Let us define \( h_i \) as the energy harvesting time required for the transmission of link \( i \). Let us replace \( p_i \) by its optimal value \( p_i^* \). Then, \( h_i = \frac{t_i^h}{\beta_i N_0 g_i} \) for \( i \in \{1, 2, \ldots, L\} \). Let us also define the sets \( I_j \) and \( H_j \), \( I_j = \{i | z_{ij} = 1\} \) and \( H_j = \{h_i | i \in I_j\} \), for \( j \in \{1, 2, \ldots, M\} \). In other words, \( I_j \) is the index set of the sensors assigned to subframe \( j \), and \( H_j \) is the set of the energy harvesting time values of the sensors assigned to subframe \( j \). In any feasible solution, based on Eqn. (3d), we have \( t_j^h \geq h_i \), \( i \in I_j \).

The optimal value of \( t_j^h \), denoted by \( t_j^{h^*} \), satisfies \( t_j^{h^*} \in H_j \), \( t_j^{h^*} = \max_{x \in I_j} h_i \), \( j \in \{1, 2, \ldots, M\} \).

C. Simplified Model

The simplified formulation is given as follows:

\[
\begin{align*}
\text{minimize} & \quad W \\
\text{subject to} & \quad \sum_{j=1}^{L} z_{ij} = 1, \quad k \in \{0, 1, \ldots, M - 1\}, \quad i \in \{1, 2, \ldots, L\} \\
& \quad t_j^h \geq z_{ij} h_i, \quad i \in \{1, 2, \ldots, L\}, \quad j \in \{1, 2, \ldots, M\} \\
& \quad S \geq \sum_{i=1}^{L} z_{ij} (\frac{L_i}{r}) + t_j^h, \quad j \in \{1, 2, \ldots, M\} \\
\end{align*}
\]

variables
\[
W \geq 0, \quad z_{ij} \in \{0, 1\}, \quad t_j^h, \quad i \in \{1, 2, \ldots, L\}, \quad j \in \{1, 2, \ldots, M\} 
\]

Eqn. (5a) is used to transform the objective from a nonlinear form of minimizing \( \max_{x \in \mathbb{R}} f(x) \) to a linear form and merged with Eqn. (3c) to obtain Eqn. (5d). The variables of the problem are \( z_{ij}, i \in \{1, 2, \ldots, L\}, j \in \{1, 2, \ldots, M\} \), continuous variable \( W \), and \( t_j^h, j \in \{1, 2, \ldots, M\} \). The resultant problem is mixed integer linear programming (MILP) problem.

D. NP-hardness of Optimization Problem

Theorem 1: The scheduling problem formulated in Section IV-C is NP-hard.

Proof: We reduce the NP-hard minimum makespan scheduling (MSP) on identical machines to our scheduling problem, by employing a similar methodology to [20]. Given a set of \( n \) jobs with processing times \( p_i, i \in \{1, 2, \ldots, n\} \), and \( m \) identical machines, the MSP aims to find an assignment of the jobs to the machines with the goal of minimizing the completion time of all the jobs, i.e., the makespan.

Let us define a problem instance where we are to schedule \( n + 1 \) sensors with packet generation periods \( T_2 = T_3 = \ldots = T_{n+1} = mT_1 \), where \( m \) is an integer greater than 1, and energy harvesting requirements \( h_1 \geq h_i, i \in \{2, 3, \ldots, n + 1\} \). Since frame and subframe lengths are equal to maximum and minimum of the packet generation periods, the number of subframes is \( m \). Obviously, the sensor with packet generation period \( T_1 \) is allocated a time slot of length \( t_1 \) in each subframe. Since that particular sensor is allocated to all subframes and has the maximum EH requirement among all sensors, \( t_j^h = h_1, j \in \{1, 2, \ldots, M\} \), based on the optimality condition of EH slot allocation derived in Section IV-B. Also, let \( t_{i+1} = p_{t_i}, i \in \{1, 2, \ldots, n\} \). The problem is to allocate \( n \) sensors of different time slot lengths to \( m \) subframes such that the maximum total active length of all subframes is minimized. Assuming that the optimal solution to this problem is less than \( S \), it is equal to \( h_1 + t_1 \) plus the optimal solution of aforementioned MSP. Since we can reduce the NP-hard MSP to an instance of the scheduling problem that was formulated in Section IV-C, the scheduling problem in Section IV-C is also NP-hard.
E. Solution Strategy

While LP-relaxation may be considered to attempt solving an MILP, it is not only suboptimal but also cannot guarantee feasibility. Because the scheduling problem formulated in Section IV-C is NP-hard, obtaining an optimal solution introduces exponential computational cost. Hence, we will propose two polynomial-time heuristic algorithms in Sections V and VI.

- We establish an analogy between our scheduling framework and task scheduling on identical machines with sequence-dependent setup times. Setup times and minimizing makespan correspond to the energy harvesting requirements of the nodes relative to each other, and minimizing the maximum total active length of the subframes in our framework, respectively.

- Motivated by this analogy, PLS and PALS are inspired by the list scheduling [48] and alternative list scheduling algorithm [49] proposed in the literature for MSP, respectively.

- PLS assigns the time slots to the subframe with the smallest total active length and repeats the allocation periodically. PALS, on the other hand, assigns the time slots to the subframe in which they will finish transmission the earliest. For zero or equal-length EH slots, PLS and PALS yield the same schedule. In general, however, they produce schedules with different maximum total active lengths.

V. Periodic List Scheduling (PLS) Algorithm

In this section, we propose a heuristic for problem (5) based on an algorithm proposed for MSP on identical machines by exploiting the analogy between them. For this purpose, we adopt a concept existing in machine scheduling problems, called sequence-dependent setup times, or shortly, setup times. In machine scheduling problems, sequence-dependent setup time \( s_{ij} \) is the required setup time of a machine when task \( j \) is to be executed immediately following task \( i \). Then, the modified processing time of a task \( j \) following task \( i \) on a machine is given by the sum of its processing time \( t_j \) and the setup time \( s_{ij} \). On the other hand, if task \( j \) is the first task to be assigned to a machine, its modified processing time is equal to the sum of its processing time \( t_j \) and machine dependent initial setup time \( s_{0j} \). Similarly, in our scheduling framework, the processing time of node \( j \) is its time slot length \( t_j \) plus the additional harvesting time it requires, which depends on the other nodes that have been scheduled priorly on the same subframe. \( s_{ij} \) corresponds to the additional harvesting requirement of sensor \( j \) with respect to sensor \( i \). If sensor \( j \) is assigned to subframe \( k \) where sensor \( i \) has already been assigned, sensor \( j \) will require additional \( s_{ij} \) time units of energy harvesting upon assignment, i.e., \( s_{ij} = \max (0, h_j - h_i), i \neq j, s_{0j}, j \in \{1, 2, \cdots, L\}, \) is defined to be the inherent energy harvesting requirement of sensor \( j \), and consequently set to \( h_j \). Thus, if no node has been allocated to the subframe so far, then the processing time of sensor \( j \) is formulated as \( p_j^k = t_j + h_j \). Otherwise, the processing time of sensor \( j \) on subframe \( k \) is given by \( p_j^k = t_j + \min_{i \in A_k} s_{ij} \).

![Fig. 3: Allocation of four nodes by PLS algorithm](image)

To complete this analogy, the list scheduling (LS) algorithm is proposed for the MSP on parallel identical machines with sequence-dependent setup times. The LS algorithm schedules the next job on a given list to the next available machine for processing [48], i.e., machine with the minimum current load at that time. In the following, we propose the scheduling algorithm PLS that similarly assigns sensors to the subframe with the smallest total active length and perform the analysis for the performance of PLS.

A. Description of PLS

PLS algorithm, as given in Algorithm (1), is described in detail as follows. At the initialization, the nodes in the list are ordered in increasing packet generation periods (Line 2). The next node \( j \) on the list is assigned to the subframe with the minimum total active length, which is indexed by \( k \) (Line 5). The corresponding processing time \( p_j^k \) is then calculated by including both the transmission time and the additional harvesting time required for the transmission of node \( j \) on subframe \( k \) (Line 6). Finally, the time slot assignment is repeated every \( s_j \) subframes, to satisfy the periodicity requirement (Line 7). The allocation of four nodes by the PLS algorithm is depicted in Fig. 3. The parameters are given as \( T_1 = 1 ms, t_1 = 0.1 ms, h_1 = 0.1 ms, T_2 = 2 ms, t_2 = 0.1 ms, h_2 = 0.4 ms, T_3 = 2 ms, t_3 = 0.2 ms, h_3 = 0.1 ms, T_4 = 4 ms, t_4 = 0.2 ms, h_4 = 0.4 ms \). The scheduling order of the nodes is \( 1-2-3-4 \). For this network, the frame and subframe lengths are given by \( F = 4 ms \) and \( S = 1 ms \), respectively. Initially, all subframes are empty. Sensor 1 is allocated to subframe 1 with its energy harvesting time \( s_{01} = h_1 = 0.1 ms \) and extended periodically. Then, all subframes have the same length and sensor 2 is allocated to subframe 1 with its additionally required harvesting time \( s_{12} = \max (0, h_2 - h_1) = 0.3 ms \) and extended periodically. Afterwards, sensor 3 is allocated to the subframe with the smallest total active length, subframe 2, with \( s_{13} = \max (0, h_3 - h_1) = 0 \) and extended periodically. Finally, sensor 4 is allocated to the subframe with the smallest total active length, subframe 2, with \( \min_{i \in \{1,3\}} s_{i4} = s_{14} = \max (0, h_4 - h_1) = 0.3 ms \) and extended periodically. The resultant schedule yields a maximum active length of \( a_2 = h_1 + s_{13} + s_{14} + t_1 + t_3 + t_4 = 0.9 ms \).
The overall complexity of PLS is $O(LM + L^2)$. The allocation of each sensor requires determining the subframe of the smallest total active length, periodic repetition of the allocation over the frame, and update of the subframe total active lengths, each of which is of $O(M)$ complexity. Finding the harvesting time additionally required by the sensor to be allocated to the subframe with the smallest total active length is also of $O(L)$ complexity.

Algorithm 1: PLS Algorithm

B. Performance Analysis of PLS

In this section, first, we analyze the properties of the PLS algorithm. Then, we define two auxiliary optimization problems, to derive the worst case theoretical bound of the PLS algorithm using the worst case bound of the LS algorithm for jobs with sequence-dependent setup times on identical machines, provided in [48].

We now continue with the properties of the PLS algorithm.

Lemma 2: The subframes to which a sensor is allocated by PLS have the same total active lengths just before the scheduling of that sensor.

Proof: The proof follows from Lemma 2 in [20]. It is based on a contradiction argument that assumes two subframes to which sensor $i$ is assigned have different total active lengths. Then, there must be a previously assigned higher priority sensor $k$ with period $T_k$ assigned to one of the subframes to which sensor $i$ is assigned but not to the other. However, because of the separation of the allocation of sensor $i$ by $s_i$, the subframes to which $i$ is allocated to one of the subframes, it must also be allocated to the other, which yields a contradiction.

Before we move on to the remaining properties, we define two auxiliary optimization problems, P1 and P2, to help us use the bound provided in [48] in establishing the bound of PLS.

Definition 1: Let us define two problems, P1 and P2. P1 is defined as the optimization problem (5), which corresponds to the optimal scheduling of $F/T_i$ jobs with processing times $t_i$ for $i \in \{1, 2, \cdots, L\}$ and sequence-dependent setup times $s_{ij}$ on $F/S$ identical machines.

We first show the relation between the optimal values of the problems P1 and P2 when there are no energy harvesting (and setup time) requirements, and then take a closer look into the outcomes of PLS and LS on these problems.

Lemma 3: Denote the optimal value of P1 and P2 by $OPT_{P1}$ and $OPT_{P2}$, respectively, for the case where $s_{ij} = 0$, $h_i = 0$, $i \in \{1, 2, \cdots, L\}$. Then, $OPT_{P2} \leq OPT_{P1}$.

Proof: P1 and P2 are equivalent except for the additional requirement of the allocation once in every $s_i$ subframes for every node $i \in \{1, 2, \cdots, L\}$ in P1. Then, $OPT_{P2} \leq OPT_{P1}$ because P1 has an additional constraint that narrows down its feasible region.

Lemma 4: Let $C_{P1,PLS}$ and $C_{P2,LS}$ denote the maximum total active length of the schedule of P1 produced by the PLS algorithm and the makespan of P2 produced by the LS algorithm proposed for solving MSP with sequence-dependent setup times [48], respectively. Then, $C_{P1,PLS} \leq C_{P2,LS}$.

Proof: The LS algorithm proposed for solving MSP with sequence-dependent setup times assigns the next job on the list to the machine with the minimum current load, along with the required setup time of the task. The PLS similarly finds the subframe of minimum total active length for the assignment of each sensor, along with the additionally required harvesting time. The periodic extension of the time slot allocation also assigns to the subframes of minimum total active length due to Lemma 2. Thus, the only difference between PLS and LS is the way of determining the required setup time. In LS, when task $j$ follows task $i$, the setup time is $s_{ij}$ by definition. In PLS, however, we require the EH slot of the corresponding subframe to be sufficient for all the nodes assigned to the corresponding subframe. Therefore, the additionally required harvesting time should be determined by comparing the harvesting requirement of sensor $j$ ($h_j$) to the maximum of the harvesting requirements of the nodes that have been allocated to the corresponding subframe ($\max_{\epsilon \in I_k} h_i$). Therefore, the corresponding setup time is less than the setup time considering only the preceding node in the subframe allocation, which was the case for the LS algorithm. Hence, in each assignment, PLS yields a better schedule in terms of maximum total active length compared to the makespan produced by LS.

Now, we are ready to establish the worst case bound of PLS.

Theorem 2: Let $C_{P1,PLS}$ be the maximum total active length of the schedule obtained by PLS and $C^*$ be the optimal value of optimization problem (5), i.e., P1. Also, assume that $s_{ij} \leq t_j$ for all $i, j \in \{1, 2, \cdots, L\}$. Then, $C_{P1,PLS} \leq 4 - \frac{2}{T_i}$.

Proof: In [48], the approximation bound is provided as $C_{P2,LS} \leq 4 - \frac{2}{T_i}$. By Lemma 3, $OPT_{P2} \leq OPT_{P1}$. By Lemma 4, $C_{P1,PLS} \leq C_{P2,LS}$. Moreover, $OPT_{P1}$ is a lower bound on the optimal value of the optimization problem (5) since it considers the same problem without the harvesting times, i.e. $OPT_{P1} \leq C^*$. Then, $\frac{C_{P1,PLS}}{C^*} \leq \frac{C_{P1,PLS}}{OPT_{P1}} \leq \frac{C_{P2,LS}}{OPT_{P2}} \leq 4 - \frac{2}{T_i}$. ■
VI. PERIODIC ALTERNATIVE LIST SCHEDULING (PALS) ALGORITHM

Periodic Alternative List Scheduling (PALS) Algorithm is inspired from the alternative list scheduling (ALS) algorithm proposed for MSP. The ALS algorithm [49] schedules the next job on a given list to the machine on which it will be completed first. List scheduling and alternative list scheduling are the same for machines with no setup times yet produce different schedules when setup times are involved.

In the following, we propose the scheduling algorithm PALS that assigns the sensor to the subframe on which it will complete transmission earliest, and perform its performance analysis.

A. Description of PALS

The only difference between PALS and PLS is that PALS assigns the next job on the ordered list to the subframe on which it will finish transmission the earliest, instead of the shortest subframe. PALS allocates the next sensor on the list to the subframe on which its transmission will be completed the earliest, taking into account both the total active length of the corresponding subframe, its own time slot length and additionally required harvesting time in case it is allocated to the corresponding subframe. If we allocate the same four nodes discussed in PLS description presented in Section V-A by using the PALS, the allocation of the first three sensors yields the same schedule as PLS. However, sensor 4 is allocated to the subframe on which it will complete transmission the earliest (subframe 1), instead of the subframe with the smallest total active length (subframe 2). The resultant schedule yields a maximum total active length of \( a_1 = h_1 + s_{i2} + s_{24} + t_3 + t_2 + t_4 = 0.8\text{ms} \).

The overall complexity of PALS is \( O(LM + L^2M) \). The allocation of each sensor requires determining the subframe on which its time slot will complete the earliest. Finding the additionally required harvesting time for each sensor is of \( O(L) \) complexity, which yields \( O(LM) \) complexity for selecting the subframe for allocation. The periodic repetition of the allocation over the frame, and update of the subframe total active lengths are of \( O(M) \) complexity. The complexity results from the repetition of these operations for \( L \) sensors.

B. Performance Analysis of PALS

In this section, we analyze the properties of PALS. With a similar method followed in Section V-B, we derive the worst case theoretical bound of the PALS algorithm.

Lemma 5: The subframes to which a sensor is allocated by PALS have the same total active lengths just before the scheduling of that sensor.

Proof: The same arguments in the proof of Lemma 2 hold for PALS.

To proceed with establishment of the worst case bound of PALS, we benefit from the methodology followed in [48], which derives the worst-case bound of LS algorithm for MSP on identical machines with sequence-dependent setup times bounded by the processing times of tasks.

Lemma 6: Let \( C_{P2,PALS-ap} \) be the makespan of the schedule obtained by PALS-ap, the counterpart of PALS without the periodic allocation property. Also, let \( M_h \) be the set of tasks already scheduled on machine \( h \), i.e the counterpart of \( I_k \), and the setup time incurred for task \( j \) on machine \( h \) be \( \max(0, h_j - \max_i \in M_h, h_i) \). Also, assume that \( s_{ij} \leq t_j \), for all \( i, j \in \{1, 2, \cdots, L\} \). Then, \( \frac{C_{P2,PALS-ap}}{C_{P2,OPT-ap}} \leq 4 - \frac{2}{3} \).

Proof: Let us start by establishing some facts about \( OPT_{P2} \), which will be useful in establishing the worst case bound. Since \( OPT_{P2} \) is the optimal value of \( P2 \) when \( s_{ij} = 0 \) for all \( i, j \in \{1, 2, \cdots, L\} \), by definition, we can write

\[
OPT_{P2} \geq \max_{i \in \{1, 2, \cdots, L\}} t_i \tag{6}
\]

\[
OPT_{P2} \geq \frac{1}{M} \sum_{i=1}^{L} t_i \tag{7}
\]

Eqn. (6) states that \( OPT_{P2} \) is greater than or equal to the processing time of the job with maximum processing time, which holds since the job with maximum processing time has to be assigned to a machine. Eqn. (7) states that \( OPT_{P2} \) is greater than or equal to the average of the processing times of all jobs, which refers to the case where the sum of the processing times of all jobs were allowed to be evenly distributed on the machines.

We now continue with a direct implication of the scheduling rule of ALS. Let \( n \) be the last job on the list to be scheduled by ALS. Also, let \( H_k \) be the time at which the processing stops on machine \( h \) before scheduling \( n \). As stated, ALS schedules the next job on the list to the machine on which it will complete the earliest. This implies that if job \( n \) was assigned to another machine than the one it has been assigned to, it would have been completed at a later time. This would yield a makespan greater than or equal to the current makespan. Then, for all \( i \in M_h \),

\[
H_h + s_{in} + t_n \geq \min_{i \in M_h} s_{in} + t_n \geq C_{P2,OPT-ap} \tag{8}
\]

for every machine \( h \), where the left-hand side is a mathematical fact due to \( s_{in} \geq \min_{i \in M_h} s_{in} \), the middle expression represents the makespan if job \( n \) was assigned to the machine \( h \), and the right-hand side represents the current makespan.

By combining the left-hand and right-hand sides of Eqn. (8) and using the relation \( s_{ij} \leq t_j \), we can write

\[
H_h \geq C_{P2,OPT-ap} - 2t_n \tag{9}
\]

Since \( s_{ij} \leq t_j \), a job can take at most \( 2t_j \) amount of time on a given machine. Then, the sum of the processing and setup times of all the tasks on all machines is less than or equal to two times the sum of the processing times of these tasks, as given by

\[
\sum_{h=1}^{M} H_h \leq 2 \sum_{i=1}^{n-1} t_i \tag{10}
\]

Next, we manipulate the expressions above to obtain the worst-case bound. First, extracting \( H_M \) and adding \( t_n + s_{in} \) to both sides in Eqn. (10), we have
Second, without loss of generality, let us assume that $C_{P2,PALS-ap}$ is given by machine $M$. Then, after the allocation of job $n$, there are two possibilities: 1) If job $n$ is allocated to machine $M$, $H_M + t_n + s_{in} = C_{P2,PALS-ap}$. 2) If job $n$ is allocated to a machine other than $M$, $H_M + t_n + s_{in} \geq C_{P2,PALS-ap} = H_M$. Using this observation, Eqns. (9) and (11),

\[
2 \sum_{i=1}^{n} t_i \geq \sum_{h=1}^{M-1} H_h + H_M + t_n + s_{in} \geq (M-1)(C_{P2,PALS-ap} - 2n) + C_{P2,PALS-ap}. \tag{12a}
\]

Then, combining Eqns. (6), (7) and (12), we have

\[
2MOPT_P2 \geq 2 \sum_{i=1}^{n} t_i \geq C_{P2,PALS-ap} + (M-1)(C_{P2,PALS-ap} - 2OPT_P2). \tag{13a}
\]

Finally, by rearranging the left-hand and right-hand sides of Eqn. (13), we obtain $C_{P2,PALS-ap} \leq 4 - \frac{2}{M}$. \hfill \blacksquare

Next, we compare the maximum total active length and makespan produced by PALS and $PALS-ap$ on $P1$ and $P2$, respectively.

**Lemma 7**: Let $C_{P1,PALS}$ be the maximum total active length of the schedule of $P1$ produced by PALS. Also, let $C_{P2,PALS-ap}$ be the makespan of $P2$ produced by the $PALS-ap$ algorithm described in Lemma 6. Then, $C_{P1,PALS} = C_{P2,PALS-ap}$.

**Proof**: Both algorithms make the assignment of a new task upon completion of the previous task in a given list. Moreover, the setup time needed for the next job is the same for both algorithms. The periodic extension of the sensor allocation in PALS also assigns to the subframes on which it will complete the earliest, due to Lemma 5. Hence, in each assignment, PALS yields the same maximum total active length as the makespan produced by $PALS-ap$. \hfill \blacksquare

The worst-case bound of PALS is then derived as follows.

**Theorem 3**: Let $C_{P1,PALS}$ be the maximum active length of the schedule obtained by PALS, and $C^*$ be the optimal value of the optimization problem (5), i.e., $P1$. Also, assume that $s_{ij} \leq t_j$, for all $i,j \in \{1,2,\cdots,L\}$. Then, $\frac{C_{P1,PALS}}{C^*} \leq 4 - \frac{2}{M}$. \hfill \blacksquare

**Proof**: By Lemma 6, $\frac{C_{P2,PALS-ap}}{C^*} \leq 4 - \frac{2}{M}$. By Lemma 7, $C_{P1,PALS} = C_{P2,PALS-ap}$. Moreover, $OPT_P1$ is a lower bound on the optimal value of problem (5) (since it considers the same problem without the harvesting times, i.e $OPT_P1 \leq C^*$). Then,

\[
\frac{C_{P1,PALS}}{C^*} \leq \frac{C_{P1,PALS}}{OPT_P1} \leq \frac{C_{P2,PALS-ap}}{OPT_P1} \leq \frac{C_{P2,PALS-ap}}{OPT_P2} \leq 4 - \frac{2}{M}. \tag{14a}
\]

Note that the worst case approximation bounds for PLS and PALS algorithms are established under the condition that setup times are bounded by time slot lengths. Let us present a channel condition where the worst case bounds are guaranteed to hold.

**Lemma 8**: If the channel gains of node $j$ satisfy $g_{ij}g_{j} \geq g_{ij}N_0$, the setup times are upper bounded by the time slot lengths, i.e., $s_{ij} \leq t_j$, for $i \in \{1,2,\cdots,L\}$.

**Proof**: Clearly, if $h_j \leq t_j$, $s_{ij} \leq t_j$ holds for $i \in \{1,2,\cdots,L\}$. Then, expanding the expression for $h_j$, we get

\[
\frac{h_j}{t_j} = \frac{\sum_{i \in \{1,2,\cdots,L\}} g_{ij}N_0}{\sum_{i \in \{1,2,\cdots,L\}} g_{ij}t_j} \leq 1. \tag{15}
\]

**Claim**: Let $\eta = 0.7$. Simulation of the medium access protocols, PLS, PALS and EDF, is performed in an event-based simulator developed in MATLAB, called ADAPTIVE_EH_SCHEDULER, and simulation of the optimal algorithm, OPT, is performed in MATLAB using IBM CPLEX solver. Both are publicly available in [52].

The attenuation of the links are determined by considering both large and small-scale statistics. The large scale-statistics is modeled as $P_{Ld}(d) = P_{d0} + 10\log_{10}(d/d_0) + Z$, where $P_{d0}$ is the path loss at distance $d$, $d_0$ is the path loss at reference distance, $d_0 = 1$ m, $\alpha$ is the path loss exponent and $Z$ is a Gaussian random variable with zero mean and standard deviation $\sigma_z$ [53]. The small scale fading has been modeled as Rayleigh fading with scale parameter $\Omega$ set to the mean power level determined by the large scale statistics.
The simulation parameters are $N_0 = 10^{-8}$ W/Hz, $p_s = 4$ W, and $r = 2$ Mbps, $\alpha = 3.5$, and $\sigma_z = 4$ dB, which are based on the parameters of existing RF implementations, such as the Powerharvester [54], and practical values provided in [13].

A. Maximum Total Active Length

Fig. 4 shows the maximum total active length of the subframes for different numbers of nodes for PLS, PALS, EDF and OPT algorithms. To highlight the importance of optimal power allocation, we also provide the results for suboptimal power allocation, in which each sensor transmits at the maximum power level $p_{max}$ instead of the optimal power for PLS and PALS, which are denoted by $PLS$ at $p_{max}$ and $PALS$ at $p_{max}$, respectively. The performances of PLS and PALS are very close to the optimal solution with maximum approximation ratios of 1.0556 and 1.0410, respectively. The approximation ratios are much lower than those proved in Theorems 2 and 3. Moreover, PLS and PALS both outperform EDF, since EDF schedules tasks as they arrive, resulting in the non-uniform distribution of the transmissions over time. Furthermore, the proposed algorithms produce significantly more adaptive schedules at optimal power allocation when compared to the suboptimal $p_{max}$ allocation, since the lower bound for the energy harvesting slot length is directly proportional to the transmission power.

Figs. 5a and 5b show the maximum total active length of the algorithms for different path loss characteristics for two distinct network sizes. As the path loss exponent increases, the received signal power of the nodes decreases due to the lossy environment. This increases the amount of the energy harvesting time required for the transmission of the nodes, thus, the maximum total active length. This effect becomes more significant as the network size increases.

B. Maximum Aperiodic Delay

Fig. 6 shows the performance of the scheduling algorithms in terms of maximum delay experienced by an aperiodic packet. PLS and PALS outperform EDF with a close-to-optimal performance. EDF schedules the tasks as they arrive, leaving almost no room in some of the subframes for the allocation of additional packets. As the number of nodes increases, the number of fully or near-to-fully allocated subframes that the additional packets have to wait to transmit increases. On the other hand, PLS and PALS distribute the allocations as uniformly as possible over the subframes, allowing the allocation of additional messaging.

C. Average Runtime

Fig. 7 shows the average runtime of the proposed algorithms with respect to the optimal algorithm. The average runtime of both heuristics is negligible due to their polynomial complexity when compared to that of the optimal, which exhibits exponential increase as the number of nodes increases.

When we collectively analyze all the performance and runtime figures, we observe that PALS perform better than PLS algorithm at the cost of slightly higher complexity.

VIII. CONCLUSION

In this paper, we study the optimal power control and scheduling of an energy harvesting wireless networked control system. The objective of the problem is to provide maximum level of adaptivity to additional messaging events and changes in the requirements of sensors and network topology. The constraints of the problem include the packet generation period, transmission delay, and energy harvesting requirements of the individual sensor nodes. We demonstrate that the power allocation problem can be separated from the scheduling and energy harvesting problem at optimality. The resultant simplified scheduling problem is then formulated as MILP, and proven to be NP-hard. We first propose a scheduling framework based on the uniform distribution of the transmissions of the sensor nodes. We then propose two polynomial time scheduling heuristic algorithms inspired by the list scheduling algorithms proposed for job scheduling on identical machines by using the analogy between sequence-dependent setup times and schedule dependent energy harvesting requirements of the sensor nodes. We derive the worst case approximation bounds for these heuristics and elaborate on the channel conditions under which these bounds hold. Finally, through simulations, we illustrate the very close-to-optimal performance and superiority of the proposed heuristics compared to the commonly used earliest deadline first and optimal algorithms.

REFERENCES

Fig. 5: Maximum total active length of algorithms for different path loss exponents

(a) Number of sensor nodes = 12

(b) Number of sensor nodes = 60

Fig. 6: Maximum aperiodic delay of algorithms for different number of devices.

Fig. 7: Average runtime of algorithms for different number of devices.


Goksu Karadag received the B.S. degree in Electrical and Electronics Engineering from Middle East Technical University in 2016. She received the M.Sc. degree in Electrical and Electronics Engineering from Koc University in 2019, in the Wireless Networks Laboratory under the supervision of Prof. S. Coleri. She has a patented invention on resource allocation in cellular networks. She is currently a Cyber Security Specialist in Cyber Strategy & Security by Design department in Vodafone TR. She has recently won a “Customer Excellence” award from Vodafone Global, with her project on radio interface security. Her research interests include communication protocol design for M2M communications, optimization of energy harvesting wireless networks, GSM & Network security and Quantum Cryptography.

Muhammad Shahid Iqbal received the B.S degree in telecommunication engineering from Government College University, Faisalabad, in 2009 and the MS degree in electrical engineering from National University of Sciences and Technology (NUST), Islamabad, Pakistan in 2013. Currently he is pursuing his Ph.D. degree in electrical and electronics engineering from Koc University, Istanbul. He has been a lecturer with the Department of Electrical Engineering, The University of Lahore, from 2013 to 2014 and with the University of Gujarat from 2014 to 2016. He received the higher education commission (HEC) Pakistan Fellowship in 2016 for full time Ph.D. studies. His research interests include wireless communication design, wireless powered communication networks, and sensor networks.
Sinem Coleri received the BS degree in electrical and electronics engineering from Bilkent University in 2000, the M.S. and Ph.D. degrees in electrical engineering and computer sciences from University of California Berkeley in 2002 and 2005. She worked as a research scientist in Wireless Sensor Networks Berkeley Lab under sponsorship of Pirelli and Telecom Italia from 2006 to 2009. Since September 2009, she has been a faculty member in the department of Electrical and Electronics Engineering at Koc University, where she is currently Professor. Her research interests are in wireless communications and networking with applications in cyber-physical systems, machine-to-machine communication, sensor networks and intelligent transportation systems. She received TUBITAK (The Scientific and Technological Research Council of Turkey) Incentive Award and IEEE Vehicular Technology Society 2020 Neal Shepherd Memorial Best Propagation Paper Award in 2020, College of Engineering Outstanding Faculty Award at Koc University and IEEE Communications Letters Exemplary Editor Award as Area Editor in 2019, Outstanding Achievement Award by Higher Education Council and Academician of the Year Award by ANTIKAD (Antalya Businesswoman Association) in 2018, IEEE Communications Letters Exemplary Editor Award and METU- Prof. Dr. Mustafa Parlar Foundation Research Encouragement Award in 2017, IEEE Communications Letters Exemplary Editor Award and Science Heroes Association - Scientist of the Year Award in 2016, Turkish Academy of Sciences Distinguished Young Scientist (TUBA-GEBIP) and TAF (Turkish Academic Fellowship) Network - Outstanding Young Scientist Awards in 2015, Science Academy Young Scientist (BAGEP) Award in 2014, Turk Telekom Collaborative Research Award in 2011 and 2012, Marie Curie Reintegration Grant in 2010, Regents Fellowship from University of California Berkeley in 2000 and Bilkent University Full Scholarship from Bilkent University in 1995. She has been Area Editor of IEEE Communications Letters and IEEE Open Journal of the Communications Society since 2019, Editor of IEEE Transactions on Communications since 2017 and Editor of IEEE Transactions on Vehicular Technology since 2016.