Dynamic online joint energy management and sampling rate control in energy harvesting aided IoT network

Chunhui Feng¹ | Qinghai Yang¹ | Meng Qin² | Kyung Sup Kwak³

¹ State Key Laboratory of ISN, School of the Telecommunication Engineering, Xidian University, Shaanxi, China
² School of Electronics and Computer Engineering, Peking University, and Shenzhen Pengcheng Laboratory, Shenzhen, Guangdong, China
³ Graduate School of Information Technology and Telecommunications, Inha University, Nam-gu, Incheon, South Korea

Correspondence
Qinghai Yang, State Key Laboratory of ISN, School of the Telecommunication Engineering, Xidian University, No.2 South Taihai Road, Xi’an, 710071, Shaanxi, China.
Email: qhyang@xidian.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 61891365, 61971327; National Research Foundation of Korea, Grant/Award Number: MSIT-NRF-2017R1A2B2012337; Institute for Information and Communications Technology Promotion, Grant/Award Number: 2019-2014-1-00729

Abstract
Energy harvesting (EH) aided Internet of Things (IoT) network is a promising paradigm to liberate IoT network from energy deficiency. Dynamic energy and traffic scheduling in such a scenario is challenging due to temporal correlation of energy constraints and delay requirements of IoT applications. In this paper, joint energy management and sampling rate control to explore the tradeoff between network utility and delay performance are studied while maintaining the energy causality constraint. Taking into account the dynamic characteristics of EH process, channel fading and traffic arrivals, a stochastic optimisation problem is formulated to maximise the network utility. Leveraging the Lyapunov optimisation approach, combined with the idea of weight perturbation, a framework is proposed to decompose the stochastic problem into several deterministic sub-problems that can be solved separately. Based on the framework, an online resource allocation algorithm is developed to achieve two major goals: first, balancing energy consumption and energy harvesting to stabilise their data and energy queues; second, deriving the utility-delay tradeoff by adjusting the control parameter. The stability of data buffer and energy buffer in the proposed network is theoretical verified with performance analysis.

1 | INTRODUCTION

Internet of Things (IoT) as an emerging paradigm provides ubiquitous intelligence and pervasive interconnections to vast number of sensing devices. The resulting IoT applications have been widely deployed in our daily life [1–3]. To support these applications, green energy supply and usage play an essential role in improving overall network performance. Energy harvesting (EH) is considered as an efficient way to achieve long-term and self-sustainable operation for networks [4–7]. Integrating EH technology into IoT networks gives rise to a merged architecture, energy harvesting aided IoT network (EHIN), which can enable and prolong the lifetime of IoT networks.

Although more energy-efficient, EHIN still face several challenges. First, the time-varying and unreliable EH process makes energy management challenging. In this paper, we focus on the stochastic model based long term performance of IoT networks. Hence, the difficulty is imposed by the resultant temporal correlation of energy allocation, that is, depleting a sensor’s battery at a rate faster than the replenishment rate leads to sensor failure. To avoid this, “energy causality” constraints must be involved and problems involving such constraints usually be formulated as dynamic programming (DP) [8]. However, the DP typically suffers from a curse of dimensionality and requires substantial statistical knowledge of EH process. Second, improving data collection efficiency of data-gathering application is a crucial issue which can be reflected by the predefined network utility in IoT networks. For instance, the more sampling data injected to the network, the higher network utility is. However, blindly maximising the sampling rate may cause IoT devices’ data queue backlog saturated and thus leading to high latency.
which is unacceptable for the applications with strict bound on delay [9]. Hence, it is a crucial issue to improve the network utility while maintaining the delay for latency-constrained IoT applications.

To address the above-mentioned challenges, we study a utility-delay tradeoff problem in an EHIN, while considering the multiple stochastic processes, including energy supply, traffic arrival and channel state. Specifically, we first propose a joint energy management and sampling rate control framework to facilitate the design of an online algorithm. The framework is developed with the aid of Lyapunov optimisation approach to achieve a close-to-optimal network utility while guaranteeing the stability of queue backlogs. Then, based on the framework, we propose an online algorithm, named joint energy management and sampling rate control (EMSC) algorithm, which makes decisions at each slot without any priori knowledge of the stochastic processes. Summarily, the major contributions of this paper are summarised as follows.

- To maintain the “energy causality” constraint, we transform it using the Lyapunov optimisation approach combined with the idea of weight perturbation. We construct a modified Lyapunov function embedded with a weight, but carefully perturb the weight used for decision making, so as to “push” the target energy queue levels toward certain value to ensure that the energy queues always have enough energy for transmission.
- To strike a balance between network utility and delay, we incorporate the sampling rate control into our resource allocation framework. The effectiveness of sampling rate control is reflected by a control parameter. Only by adjusting the control parameter can the proposal derive the utility-delay tradeoff on demand.
- The proposed algorithm is analysed to be able to achieve an $O(1/\sqrt{V})$, $O(V^2)$-tradeoff between utility optimality and delay performance for any $V > 0$. We also compute the required battery capacity. Furthermore, simulation results show the superior performance of the proposal on delay-guaranteed.

The rest of this paper is organised as follows. Section 2 overviews the related works in the existing literature. Section 3 states the network model and the problem formulation. The resource allocation framework and the EMSC algorithm are devised in Section 4. Section 5 analyses the optimality and stability of the proposed algorithm. Simulation results are presented in Section 6, and Section 7 concludes this paper.

2 RELATED WORK

There have been extensive works investigate the resource allocation in EHINs, like sensor networks, machine-type communication (MTC) networks, based on a deterministic optimisation framework [10–13]. Reference [10] developed a utility-optimal algorithm for multihop EH-aided cognitive radio sensor network, which jointly control the sampling rate and channel access. In [11], sensors schedule the actions between spectrum sensing, data transmission and EH. By jointly optimising the duration of each action, the energy efficiency was maximised under the energy causality constraint. Reference [12] studied energy efficient resource allocation for a MTC enabled cellular network with EH via joint power control and time allocation. In particular, the authors considered a nonlinear EH model which leads to a nonsmooth objective function and nonsmooth constraints. In [13], authors aimed to optimise data gathering in a rechargeable sensor network under the assumption that the EH rate of each sensor can be estimated with high accuracy.

Due to the unpredictability and dynamics of EH process, the above deterministic optimisation framework may not capture the characteristics of EH process. Thus, the stochastic optimisation framework based resource allocation have drawn great attention in EHINs. With only the statistics and causal knowledge of EH state, EH process is modeled by a Markov chain to capture the dynamics over time in [14, 15]. Reference [14] looked at the problem of maximising the time-average data transmission rate for cognitive radio nodes which are powered by a wireless beacon with EH capability. Authors in [15] developed a balanced policy to adapt the transmission probability to the EH state, so that energy harvesting and consumption can be balanced. References [16, 17] explored the statistics of EH process by machine learning approach. Reference [16] adapted the transmission scheme to the unknown EH process via a learning approach and proposed a transmitted data maximisation algorithm. Without any distribution or future information of EH process, reference [17] proposed a suboptimal power control in the online setting. Using the tools from Lyapunov technique, the policy developed in [18] is allowed to adapt its power actions to the instantaneous EH state on each slot, while maintaining the perpetual operation for sensor network. As a common feature, the works in [14–18] ignores the stochastic dynamics of the traffic arrivals with the simplified assumption that the traffic arrival rate is inside the network capacity region. Thus, the above resource allocation scheme neglect the delay of critical data, which, is an important metric to measure the quality of service of IoT applications.

In Summary, the above mentioned works either consider deterministic optimisation framework, in which the distribution information of all the stochastic processes is known in advance, or fail to meet delay requirements for delay sensitive IoT applications. Therefore, they cannot be used into our stochastic optimisation problem and can hardly depict the utility-delay tradeoff, which is exactly the objective of our work.

3 SYSTEM MODEL

IoT applications rely on a massive number of IoT devices (IoTDs) that generate small data packets. These packets are mostly transmitted in the uplink direction, toward a control center for data storage and processing [19]. Due to this uplink-
centered nature, we consider the uplink scenario in an EHIN consisting of an access point (AP) and $N$ IoTDs indexed by $\mathcal{N} = \{1, 2, \ldots, N\}$. The concerned EHIN operates in discrete time with normalised time slot $t \in \mathcal{T} = \{1, 2, \ldots, T\}$. The spectrum bandwidth is equally divided into $K$ orthogonal subchannels, denoted as $\mathcal{K} = \{1, \ldots, k, \ldots, K\}$, and each IoTD transmits data over allocated orthogonal.

### 3.1 Subchannel assignment

To schedule the transmission, the AP needs to allocate subchannels to IoTDs. We consider the scenario that the number of IoTDs no less than that of subchannels. $\alpha_{n,k}(t)$ is defined as the subchannel assignment indicator of IoTD $n$ on the $k$th subchannel at time slot $t$, that is, $\alpha_{n,k}(t)$ is equal to 1 if subchannel $k$ is allocated to IoTD $n$ and 0 otherwise. To avoid interference among IoTDs, each subchannel can be allocated to at most one IoTD at each slot, namely

$$\sum_{n \in \mathcal{N}} \alpha_{n,k}(t) \leq 1, \forall k \in \mathcal{K}. \quad (1)$$

In addition, each IoTD can be served by at most one subchannel at each slot, we have

$$\sum_{k \in \mathcal{K}} \alpha_{n,k}(t) \leq 1, \forall n \in \mathcal{N}. \quad (2)$$

### 3.2 Admission control and data queue dynamics

In the considered EHIN, separate data queues are maintained for each IoTD to temporarily backlog the arrived data. Let $Q(t) = (Q_1(t), Q_2(t), \ldots, Q_N(t))$ represents the data queue backlog of all IoTDs at time slot $t$. $a_n(t)$ denotes the amount of random data arrivals destined for IoTD $n$ at slot $t$. Assume that $a_n(t), \forall n \in \mathcal{N}$ are independent and identical distribution (i.i.d.) over time slots according to a general distribution and are independent with respect to $n$. Furthermore, there exists a certain peak amount of traffic arrivals $a_{n,\text{max}}$ satisfying $a_n(t) \leq a_{n,\text{max}}$. In practice, the statistic of $a_n(t)$ is usually unknown to IoTD $n$, and the achievable capacity region is difficult to estimate, that is, the situation that the arrival rate is outside of the network capacity region may occur. In this situation, data queues cannot be stabilised without a admission control mechanism to limit the amount of data that is admitted.

As illustrated in Figure 1, each IoTD first makes decisions on whether the arrived data is admitted or not over slots. If not, it directly drops the data. Let $r_n(t)$ and $b_n(t)$, respectively, represent the amount of admitted data and dropped data at IoTD $n$ at slot $t$. Apparently, $r_n(t)$ should be less than the traffic arrival rate, that is,

$$r_n(t) \leq a_n(t), \forall n \in \mathcal{N}. \quad (3)$$

Denote $\mu_n(t)$ as the data transmission rate of IoTD $n$ at slot $t$. The amount of data that IoTD $n$ can transmit over channel $k$ is determined by two factors. One is the data queue occupancy, that is, the IoTD $n$ can only transmit the available data in its queue, which can be expressed as

$$0 \leq \mu_n(t) \leq Q_n(t). \quad (4)$$

Another is the channel capacity $\lambda_{n,k}(t)$, that is, if the channel $k$ is allocated to IoTD $n$, the data transmission rate is bounded by

$$\mu_n(t) \leq \sum_{k \in \mathcal{K}} \alpha_{n,k}(t) \lambda_{n,k}(t). \quad (5)$$

Besides, to depict the time-varying characteristics of channel fading, we assume $\lambda_{n,k}(t)$ randomly varies over time slots in an i.i.d. fashion and is bounded by $\lambda_{n,k}(t) \leq \lambda_{\text{max}}, \forall n \in \mathcal{N}, k \in \mathcal{K}$.

Based on the aforementioned analysis, the data queue length varies over time slot with the admitted data $r_n(t)$ as input, and the server process $\mu_n(t)$ as output, as follows:

$$Q_n(t + 1) = Q_n(t) - \sum_{k \in \mathcal{K}} \alpha_{n,k}(t) \mu_n(t) + r_n(t). \quad (6)$$

To model the impact of joint sampling rate control and energy management on the average delay and the achieved network utility, the definition of the network stability is given.

$$\lim_{{T \to \infty}} \frac{1}{T} \sum_{{t=1}}^{T} \mathbb{E}[Q_n(t)] < \infty. \quad (7)$$
3.3 | Energy supply and energy queue dynamics

The EH process is characterised by the energy supply rate \( \tau_a(t) \), which denotes the amount of energy that can be harvested by IoT-D at time slot \( t \). We model \( \tau_a(t) \) is i.i.d. over slots and is upper bounded by \( \tau_{\text{max}} \). Further, to capture the time-varying and unreliable characteristics of EH process, we assume that IoT-Ds may have no priori knowledge of energy supply rate, which is practically true when the statistical knowledge of EH process is not available.

We denote \( e_a(t) \) the amount of harvested energy by IoT-D at time slot \( t \), which is bounded by

\[
0 \leq e_a(t) \leq \tau_a(t).
\]

(8)

The energy consumption of data sensing is assumed to be a linear function of the sampling rate \( r_a(t) \) [20] and denoted by \( \kappa_s r_a(t) \). We assume all IoT-Ds use the same transmission power \( \kappa_T \) over each allocated subchannel for data transmission. Thus, the overall energy consumed by the IoT-D is

\[
P_a^{\text{total}}(t) = \kappa_s r_a(t) + \sum_{k \in \mathcal{K}} \alpha_{a,k}(t) \kappa_T, \forall n \in \mathcal{N}.
\]

(9)

Besides, we use \( P_{\text{max}} \) to denote the upper bound of \( P_a^{\text{total}}(t) \) and is expressed as \( P_{\text{max}} = \kappa_s \tau_{\text{max}} + \kappa_T \).

We define \( E(t) = (E_1(t), E_2(t),..., E_N(t)) \) as the energy queue size of IoT-Ds. From the discussion above, the dynamics of \( E_a(t) \) across time slots can be expressed as

\[
E_n(t + 1) = E_n(t) - P_a^{\text{total}}(t) + e_a(t).
\]

(10)

Note that in a given time slot \( t \), the “energy causality” constraint must be met, that is, the energy consumed is no more than the energy stored:

\[
P_a^{\text{total}}(t) \leq E_n(t), \forall n \in \mathcal{N}.
\]

(11)

Furthermore, the energy charging is bounded by the battery capacity, which is denoted by \( \Omega_n \). We assume the battery capacity is the same for all IoT-Ds, so the subscript \( n \) is omitted here for simplicity. Thus, we have

\[
E_n(t) + e_n(t) \leq \Omega, \forall n \in \mathcal{N}.
\]

(12)

3.4 | Problem formulation

Based on the aforementioned models, we formulate a stochastic optimisation problem to maximise the time average aggregate network utility subject to the constraints mentioned above. We define utility function \( U(r_a(t)) \) be a function of sampling rate \( r_a(t) \) and assume it to be a strictly concave function [21]. Then, the time average aggregate network utility can be written as

\[
\mathcal{U} = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E} \left[ \sum_{a \in \mathcal{N}} U(r_a(t)) \right].
\]

(13)

To simplify the presentation, we define \( \mathbf{r}(t) \triangleq (r_1(t), r_2(t),..., r_N(t)) \) as the set of the variables to be optimised. The aggregate network utility can be maximised by optimising \( \mathbf{r}(t) \) under the following stochastic optimisation problem

\[
P1: \mathcal{U} \rightarrow \mathbf{r}(t)
\]

s.t. Equations (1) to (12).

(14)

We can see from the formulation of the above problem, on one hand, the prior knowledge on the statistical information of the multiple stochastic processes is hardly available, which limits the EHIN to obtain the maximum network utility under an off-line framework; on the other hand, the “energy causality” constraint greatly complicates the design of an efficient scheduling algorithm due to the fact that the current power allocation strategy may cause energy outage in the future and thus effect the future decision. Taking the above challenges into consideration, we design our EMSC algorithm in the following section.

4 | UTILITY-DELAY TRADEOFF ALGORITHM: EMSC

4.1 | Design principle of EMSC

The above two challenges inspire us to deal with the stochastic optimisation via the Lyapunov optimisation framework. On the one hand, Lyapunov optimisation does not require the prior knowledge on the statistical information of the system randomness. Moreover, the “energy causality” constraint can be removed leveraging the Lyapunov optimisation approach combined with weight perturbation.

Before the algorithm, we give the following design flow according to the Lyapunov optimisation framework. We first construct the modified drift function to relax the “energy causality” constraint and queue stability constraints, then the optimisation objective is mapped into a penalty function. By minimising the “drift plus penalty” term, Lyapunov optimisation can minimising the queue length while obtain the optimal utility. At this point, the original stochastic optimisation problem can be decomposed into three separated deterministic subproblems, which can be solved independently of others. Detailed design steps are given in the following subsections.
4.2 Problem transformation

To relax the “energy causality” constraint, we employ weighted perturbation based Lyapunov optimisation technique. We first define the weighted perturbation as the battery capacity $\Omega$. The concatenated vector $Z(t) = [Q(t), E(t)]$ is used to denote the network state, which captures the queue backlog. Then, a perturbed Lyapunov function is defined as

$$L(Z(t)) = \frac{1}{2} \sum_{n \in N} (Q_n(t))^2 + \frac{1}{2} \sum_{n \in N} (E_n(t) - \Omega)^2. \quad (15)$$

The intuition behind the use of weighted perturbation parameter $\Omega$ is that by keeping the Lyapunov function value small, we needed “push” $E_n(t)$ value towards $\Omega$. Thus, by choosing a proper choice $\Omega$, we can ensure there is always enough energy stored in the energy queue for sensing and transmission. Meanwhile, when minimising $L(Z(t))$, we push the data queue backlog towards zero, which is equivalent to satisfying network stability constraint (7). Then, the one-slot conditional Lyapunov drift can be defined as

$$\Delta(Z(t)) = \mathbb{E}\{L(Z(t+1)) - L(Z(t)) | Z(t)\}, \quad (16)$$

where the expectation is w.r.t. the random stochastic processes and the control actions.

To incorporate the objective function into Lyapunov drift, we map the objective function to an appropriate penalty. Instead of greedily minimising Lyapunov drift, actions are taken to minimise the drift-minus-utility $\Delta V(Z(t)) = \Delta(Z(t)) - V \mathbb{E}(U|Z(t))$, where $V$ is a non-negative control parameter to provide a tradeoff between queue backlog and network utility maximisation. Since the time average queue length is proportional to the queuing delay according to Little’s law, the control parameter $V$ provides a guideline to balance the network utility and delay. Then, we have the following lemma regarding the drift-minus-utility.

**Lemma 1.** For any optimisation decision made on slot $t$, and all possible values of $Z(t)$ the value of drift-minus-utility is bounded by

$$\Delta V(Z(t)) \leq B - \mathbb{E}\left\{ V \sum_{n \in N} U(\alpha_n(t)) | Z(t) \right\}$$

$$+ \mathbb{E}\left\{ \sum_{n \in N} (E_n(t) - \Omega) r_n(t) | Z(t) \right\}$$

$$+ \mathbb{E}\left\{ \sum_{n \in N} Q_n(t) r_n(t) - \kappa_T (E_n(t) - \Omega) | Z(t) \right\}, \quad (17)$$

where $\sum_{n \in N}$ is the simplification of $\sum_{n \in N} \sum_{k \in K^v} B = \frac{N}{2} \Omega^2 + A_{\max}^2 + B_{\max}^2 + \tau_{\max}^2$ is a positive constant.

**Proof:** See Appendix A.1.

At this point, the design principle behind the EMSC algorithm is to minimise the RHS of (17) according to the principle of opportunistically minimising an expectation in per-slot manner.

4.3 Scheduling decisions

In this subsection, we present our EMSC algorithm to address the above problem. Since the decision variables $e(t), r(t)$ and $(\alpha(t), \mu(t))$ can be decoupled with each other and are independent of the current backlog $Z(t)$, the minimisation problem can be separated into three subproblems as follows.

4.3.1 Energy management

The energy management subproblem is expressed as

$$\min_{e(t)} (E_n(t) - \Omega) r_n(t),$$

s.t. (8), (12). \quad (18)

Clearly, the solution of (18) has the following on-off structure and is given by:

$$e_n^*(t) = \begin{cases} \min(\Omega - E_n(t), \alpha_n(t)), & E_n(t) < \Omega, \\ 0, & \text{otherwise.} \end{cases} \quad (19)$$

4.3.2 Sampling rate control

The admission control subproblem is given by

$$\min_{r(t)} r_n(t) (Q_n(t) - \kappa_T (E_n(t) - \Omega)) - V U(r_n(t)),$$

s.t. (3). \quad (20)

Since the utility function $U(r_n(t))$ is a concave function and (3) is a linear constraint, the admission control problem is a convex optimisation problem. We can drive the optimal solution based on the convex optimisation theory and is given by

$$r_n^*(t) = \min \left( \max \left( (d')^{-1}(\Pi), 0 \right), a_s(t) \right), \quad (21)$$
where \( (')^{-1} \) is the inverse of the first derivative of \( U(\cdot) \) and 
\[ \Pi = \frac{U'(\cdot) - \kappa_\tau(E_n(\cdot) - \Omega)}{U'(\cdot)}. \]

### 4.3.3 Channel and transmit rate allocation

The channel and transmit rate allocation subproblem can be written as follows:

\[
\min_{\alpha(t), \mu(t)} \sum_{n, k} \alpha_{n,k}(t)\left[ -Q_n(t)\mu_n(t) - \kappa_T(E_n(t) - \Omega) \right],
\]

\[ \text{s.t.} \quad (1), (2), (4), (5). \quad (22) \]

The objective function of channel and transmit rate allocation (CTRA) subproblem consists of the product of a integer variable \( \alpha(t) \) and a continuous variable \( \mu(t) \), which makes CTRA a nonconvex mixed integer programming problem. To facilitate the design of a tractable resource allocation solution, we adjust the data queue length and modify the objective function of CTRA. Instead of solving the original CTRA, we transform it to the modified CTRA (m-CTRA) problem to find a suboptimal solution.

Here, we define the modified \( Q_n(t) \) as

\[
Q_n^{\text{mod}}(t) = Q_n(t) - \lambda_{n,k}(t)^+. \quad (23)
\]

Now, we can replace \( Q_n(t) \) in the objective function of CTRA with \( Q_n^{\text{mod}}(t) \) and construct the m-CTRA. For the sake of brevity, we define \( m_{n,k}(t) = -Q_n^{\text{mod}}(t)\mu_n(t) - \kappa_T(E_n(t) - \Omega) \), then the m-CTRA can be written as

\[
\min_{\alpha(t), \mu(t)} \sum_{n, k} \alpha_{n,k}(t)m_{n,k}(t),
\]

\[ \text{s.t.} \quad (1), (2), (4), (5). \quad (24) \]

Note that if we can remove the transmit rate variable \( \mu(t) \) and relax the constraints related to it, the m-CTRA becomes a one-to-one matching problem. Because of this, we transform the m-CTRA into an integer problem by the following two steps. First, we show that the objective function of m-CTRA is minimised if sensors transmit data at full capacity on their assigned channels. Second, we replace \( \mu_n(t) \) by the channel capacity \( \lambda_{n,k}(t) \). Thus, the continuous variable can be removed from the objective function and the m-CTRA is transformed into a one-to-one matching problem. The detailed process is shown in the following.

(1) In the following lemmas, we show that a channel \( k \) is assigned to sensor \( n \) if and only if it has a sufficient amount of data to transmit (i.e., constraint (4) can be satisfied) and it transmits the data at full channel capacity.

**Lemma 2.** The channel \( k \) is assigned to sensor \( n \) only when the data queue occupancy is larger than the channel capacity, that is, \( Q_n(t) > \lambda_{n,k}(t) \).

**Proof.** Suppose that channel \( k \) is assigned to sensor \( n \), that is, \( \alpha_{n,k}^*(t) = 1 \). Note that our objective is to minimise the objective function of m-CTRA. In the condition that \( \sum_{k\in K} \alpha_{n,k}^*(t) = 1 \), \( m_{n,k}(t) \) must be negative. With the expression of \( m_{n,k}(t) \), we can further drive that \( Q_n(t) > \lambda_{n,k}(t) \). \( \square \)

**Lemma 3.** If channel \( k \) is assigned to sensor \( n \) at time slot \( t \), it must transmit at full channel capacity, which can be expressed as

\[
\mu_n(t)^* = \begin{cases} 
\sum_{k=1}^K \alpha_{n,k}^*(t)\lambda_{n,k}(t), & \sum_{k=1}^K \alpha_{n,k}^*(t) = 1, \\
0, & \text{otherwise.}
\end{cases} \quad (25)
\]

**Proof.** Here we first consider the condition that there is no channel allocated to sensor \( n \), that is, \( \alpha_{n,k}(t) = 0, \forall k \in K \). Combined with the constraint (5) and its non-negativity, we can derive \( \mu_n(t)^* = 0 \).

Then, we consider the condition that channel \( k \) is assigned to sensor \( n \). We prove that \( \mu_n(t)^* = \sum_{k\in K} \alpha_{n,k}(t)\lambda_{n,k}(t) \) is the optimal solution to the m-CTRA. Since \( m_{n,k}(t) = [-Q_n^{\text{mod}}(t)\mu_n(t) - \kappa_T(E_n(t) - \Omega)] \), we can observe that \( m_{n,k}(t) \) is in inverse proportion to \( \mu_n(t) \). Hence, we choose \( \mu_n(t) \) as large as possible to minimise \( m_{n,k}(t) \). Recalling that \( \mu_n(t) \) is restrained by its channel capacity and queue backlog. Further, from Lemma A.1, we can see that constraint (4) is obviously satisfied. Thus, \( \mu_n(t) \) is only bounded by channel capacity constraint. We choose \( \mu_n(t)^* = \lambda_{n,k}(t) \) as the optimal solution to m-CTRA. This completes the proof. \( \square \)

(2) According to Lemma 2 and Lemma 3, we can replace the transmission rate by channel capacity and rewritten the m-CTRA as follows

\[
\min_{\alpha(t)} \sum_{n, k} \alpha_{n,k}(t)[-Q_n^{\text{mod}}(t)\lambda_{n,k}(t) - \kappa_T(E_n(t) - \Omega)]
\]

\[ \text{s.t.} \quad (1), (2). \quad (26) \]

The m-CTRA is transformed into a channel allocation subproblem, which is a one-to-one mapping problem and can be solved by the adaptive Hungarian algorithm.

Based on the above analysis, the whole process of the EMSC algorithm is summarised in Algorithm 1. The EMSC algorithm solves the energy management, sampling control and channel and transmit rate allocation subproblems in sequence. Then, update the backlog in all queues for the procedures in the following time slot.
Algorithm 1: EMSC algorithm

Input: Observe the queue state \( Z(t) \), traffic arrival \( a(t) \) and energy supply rate \( \tau(t) \).
Output: Control policy \( e^*(t) \), \( r^*(t) \), \( \mu^*(t) \) and \( \sigma^*(t) \).

At any IoT device \( n \):
1: Determine energy harvesting policy \( e^*(t) \) according to Equation (19)
2: Decide \( r^*(t) \) based on (21)

At the fusion node:
1: Choose \( \mu^*(t) \) and \( \sigma^*(t) \) by solving (26).
2: Update \( Q(t) \) and \( E(t) \) according to their respective queue dynamics.

4.4 Application and implementation architecture of the EMSC algorithm

Following the EMSC algorithm, we propose an implementation architecture for its applications in practical systems, as shown in Figure. Moreover, some descriptions are given as follows to show the favorable properties of the EMSC algorithm for providing engineering guidelines.

(a) The EMSC algorithm takes the joint channel state information (CSI), data queue state information (DQSI) and energy queue state information (EQSI) as the inputs and produces the scheduling decisions as the output on every time slot.
(b) The sampling rate control (SRC) subproblem and energy management (EM) subproblem can be decomposed for individual IoTD, such that can be distributively executed at each IoTD. Note that the joint channel and transmit rate allocation (CTA) depends on real time DQSI and EQSI fed back by all IoTDs. Hence, the optimisation of channel allocation and transmit rate can only be coordinated at the AP in a centralised manner.
(c) Both the SRC and EM subproblems have closed-form solutions, thus their complexity is negligible. The complexity of the proposed algorithm is dominated by solving CTA subproblem with computational complexity of \( O(NK + \log(\min(N,K))) \) [22]. Therefore, the complexity of the proposed algorithm increases linearly with the number of \( N \) and \( K \), and is far lower than that of algorithms designed based on Markov decision process (MDP) (the computational complexity of MDP-based algorithms increases exponentially with \( N \) and \( K \) [23]).

Next, we provide an application scenario of the proposed EMSC algorithm in transportation system. Various sensors and automated devices are installed on the vehicles to collect the task-oriented sensory data. The sensory data is further transmitted to the roadside units for data analysis. Considering some delay-sensitive tasks, the amount of admitted traffic and service traffic should be jointly controlled to guarantee the queueing delay, which is the major issue EMSC algorithm is focused on. Moreover, it is desirable to enable these sensors with green energy harvesting to realise their self-sustainability. Under this scenario, the design principle of the energy management policy can also follow that of the EMSC algorithm.

5 PERFORMANCE ANALYSIS

Here, the performance bounds of the proposed EMSC algorithm will be mathematically analysed. We first illustrate the performance in Theorem 1 and then provide some intuitional observations behind them.

Theorem 1. The EMSC algorithm with any \( V > 0 \) has the following properties.

(a) We use \( U^* \) and \( U^{Alg} \) denote the theoretical optimum value of the original problem and the network utility achieved by the proposed algorithm, respectively. Then

\[
U^{Alg} \geq U^* - \frac{B}{V},
\]

where \( B = B + NK^2 \max \).

(b) The time-average data queue length satisfies the following upper bound

\[
0 \leq Q_n(t) \leq Q_{\max}, 0 \leq t \leq T
\]

where the upper bound is given by \( Q_{\max} = \varphi_U V + r_{\max} \varphi_U \) is the maximum derivative of \( U(r_n(t)) \).

(c) With a battery capacity \( \Theta \) given by

\[
\Theta = \max \left( \frac{N V \varphi_U}{\kappa_S} + \rho_{\max} \frac{Q_{\max} \lambda_{\max}}{\kappa_T} + P_{\max} \right),
\]

the “energy causality” constraint will be guaranteed.

Proof. See Appendix A.2. □
To understand the results in Theorem 1, we provide intuitional observations.

1. Part (a) shows that the achieved network utility increases proportional to $V$, and that $U^A$ can be arbitrarily close to the optimal value by setting a sufficiently large value of $V$. Moreover, note that in Section 3, we transform the CTRA subproblem to the modified CTRA subproblem. The performance loss caused by the transformation is shown by $B$, in which $B$ is the performance gap if we do not perform the transformation.

2. Part (b) demonstrates the upper bounds of data queues and implies that the time-average data queue length increases linearly with the value of $V$. Combining this with Part (1), we can see that the EMSC algorithm achieves an $[O(1/V), O(V)]$ utility-delay tradeoff. Such quantitative result for utility-delay tradeoff will be useful for practical implementations.

3. Part (c) provides explicit characterisation of the needed energy capacity. Replacing $Q^\text{max}$ in Equation (28) with Equation (29), we observe that the value of battery capacity only depends on the known parameters and is easily to be determined. Furthermore, we see that the battery capacity is deterministically upper bounded by a constant of size $O(V)$. Combined with Part (1), an explicit capacity requirement for energy storage is provided according to the desired utility performance.

6 | SIMULATION RESULTS

In this simulations, we verify the analysis and evaluate the performance of the proposed algorithm via MATLAB simulations. We use the network topology consisting of 15 IoTDs and an AP in which the IoTDs are randomly distributed in a circular area with a radius of 30 meters and the AP is located at the center of this area. The IoTDs transmit data to the fusion node over $K = 4$ channels. Similar to [24], the utility function is given by $U(r_i(t)) = \log(1 + r_i(t))$, and its maximum derivatives is $g_i = 1$. The energy consumption rate of data sensing is $\lambda_1 = 0.1$ Watt/bit and the transmit power is set to $\lambda_T = 1$ Watt. The energy supply rate is uniformly distributed in $[0, \tau_{\text{max}}]$, where $\tau_{\text{max}} = 2$ Joule. The channel fading coefficient is expressed as $h_{a,k}(t) = d_a^{-\alpha}h_{a,k}(t)$, where $d_a$ is the distance between IoTD $a$ and the AP, $\alpha$ denotes the path loss exponent and equals to 4. Besides, $\tilde{h}_{a,k}(t)$ is the small-scale fading coefficient, which is uniformly distributed between $(0.9, 1.1)$ and i.i.d. across time slots [25]. Corresponding, the instantaneous channel capacity is modeled as $\tilde{\lambda}_{a,k} = \log(1 + \frac{\lambda_T \tilde{h}_{a,k}(t)}{N_0})$, where $N_0$ denotes the noise power and equals to $10^{-5}$ Watt. The traffic arrivals follow a Poisson distribution, and the mean traffic arrival rate is given by $\tilde{\lambda}_a = \lambda$. Note that each point of the following curves is plotted based on a $10^2$-run and averaged over these values.

6.1 | Network utility and queue dynamics

Figure 3 displays the evolution of network utility versus with the value of control parameter $V$. Since a larger $V$ means that the controller places more emphasis on network utility maximisation than on the control of queue backlog, the network utility grows linearly as the value of $V$ increases at the startup phase. Then, the utility improvement starts to diminish with excessive increase of $V$. This shows that a larger $V$ adversely aggravate the congestion of data queue and thereby limits the sampling rate. Finally, the utility gradually converges to the optimal value, which validates part (a) in Theorem 1.

Figure 4 demonstrates the data queue stability with different $V$. It can be seen that the data queue backlog increases at the startup phase and eventually converges to the time average value. Furthermore, the upper bounds of the data queue backlog increases with the value of $V$. Since data queue backlog is proportional to delay according to Little’s law, this indicates that the delay grows linearly in $O(V)$. Figures 3 and 4 together indicate that there is a tradeoff between utility and delay, and can be quantitatively depicted by $[O(1/V), O(V)]$.

Similar to the data queue, the energy queue backlog increases at the early phases and then converges to the time average value, as illustrated in Figure 5. It can be seen that the time average energy queue backlog is deterministically upper bounded by the battery capacity. Furthermore, a larger $V$ increases the time average value, which indicates that the EMSC requires energy storage devices that are of $O(V)$ sizes. This validates part (c) in Theorem 1.

6.2 | Impacts of traffic arrival rate on system performance

Figures 6 and 7 show the variation tendency of throughput with the changes in the traffic arrival rate under different $V$. Figures 6 and 7 show the throughput for the range of $\lambda = 2 : 2 : 20$ and $\lambda = 0.02 : 0.02 : 0.26$, respectively. Here, throughput refers to the maximum amount of admissible uplink traffics that can be stabilised in the system, and its time average value is of our interest, which can be expressed as $Thr = \sum_{t=1}^{\infty} \{ \sum_{a\in A} r_a(t) \}$. From Figure 6, we see that throughputs under the proposed EMSC algorithm are equal to the traffic arrival rate at first, then grow to the maximum and keep unchanged as $\lambda$ increases. More specifically, Figure 7 indicates that the maximum traffic arrival rate that system can stably carry is 0.04, 0.08 and 0.16, respectively, under different control parameter $V$. This is because the system can stably work for small traffic arrival rate, but has to start admission control for large traffic arrival rate.

Figure 8 shows that the data queue length is controlled to slowly increase with the traffic arrival rate and the increasing rate decreases with a larger traffic arrival rate, this follows from the fact that the EMSC algorithm adapts to the traffic arrivals...
attributed to the queue stability constraint in (7) and will start admission control in the heavy traffic states.

6.3 Performance comparison

Here, we compare the EMSC algorithm with a baseline algorithm, which makes greedy decisions to maximise the network utility. In each slot, the greedy algorithm chooses the schedule actions that offers the most obvious and immediate utility. Considering the channel and transmit rate allocation subproblem, the greedy algorithm first arranges the data queues in a descending order with respect to their lengths. Then, it assigns the channel with the maximum capacity to the longest data queue for data transmission and transmits under channel capacity. At last, the data queue is removed from the arrangement. Repeat the above steps until all the channels are assigned.

In the following, we divide the network operation time into five time blocks. To validate the efficacy of the EMSC algorithm, we compare the utility between the greedy algorithm and the EMSC algorithm under each time block. As shown...
FIGURE 5  Throughput for the range of $\lambda = 2 : 2 : 20$ under different $V$. 

FIGURE 6  Throughput for the range of $\lambda = 0.02 : 0.02 : 0.26$ under different $V$. 

FIGURE 7  Data queue length against traffic arrival rate $\lambda$ under different $V$. 

FIGURE 8  Utility comparison in different time block between the greedy algorithm and the EMSC algorithm ($V = 10$). 

FIGURE 9  Comparison of total data queue length between the greedy algorithm and the EMSC algorithm ($V = 10$). 

in Figure 9, the utility of the greedy algorithm is shown to be high in the first time block, then decreases to a stable value in the following several time blocks. The reason is that the greedy algorithm is more concerned about the immediate utility, so it requires as much data to maximise the utility at the beginning of operation. However, its channel allocation and transmit rate control actions cannot efficiently clear the backlog in the data queue to accommodate more coming data, such that its utility decreases drastically in the following time blocks. By contrast, although the utility achieved by the EMSC algorithm is relatively lower at the beginning, it gradually increases to a stable value higher than that of the greedy algorithm.

Further, we compare the total data queue length of the two algorithms to show the superior performance of the EMSC algorithm on delay. Figure 10 shows that the total data queue length in the EMSC algorithm is much lower than that in the greedy algorithm. Since the time-average data queue length is proportional to delay, this indicates that the EMSC algorithm outperforms the greedy algorithm in terms of delay.
CONCLUSION

This paper has focused on the joint energy management and sampling rate control problem in an EHIN. Based on the Lyapunov optimisation method, combined with the idea of weight perturbation, the original stochastic problem has been decomposed into three deterministic sub-problems. By solving each sub-problem, we present our EMSC algorithm. An $[\mathcal{O}(1/V), \mathcal{O}(V)]$ utility-delay tradeoff has been derived by the proposed algorithm which has been both verified by both performance analysis and numerical simulations. Further, The simulation results have shown the necessity of sampling rate control and have validated the advantages of the proposed algorithm in terms of utility and delay.

ACKNOWLEDGEMENTS

This work was supported in part by the NSF China under Grant 61801365 and 61971327, in part by the China Postdoctoral Science Foundation under Grant 2018M643581, in part by the National Natural Science Foundation of Shaanxi Province under Grant 2019JQ-152, and the Fundamental Research Funds for the Central Universities.

ORCID

Chunhui Feng https://orcid.org/0000-0003-1644-0177

REFERENCES

1. Ahlgren, B., Hidell, M.: Internet of things for smart cities: Interoperability and open data. IEEE Internet Comput. 20(6), 52–56 (2016)
2. Aeshad, S., et al.: Recent advances in information-centric networking-based internet of things (ICN-IoT). IEEE Internet Things J. 6(2), 2327–4662 (2019)
3. Voas, J., et al.: A closer look at IoT’s things. IEEE IT Prof. 20(3), 11–14 (2018)
4. Ko, H., Pack, S.: Neighbor-Aware energy-efficient monitoring system for energy harvesting internet of things. IEEE Internet Things J. 6(3), 5745–5752 (2019)
5. Nguyen, T.D., et al.: A distributed energy-harvesting-aware routing algorithm for heterogeneous IoT networks. IEEE Trans. Green Commun. Netw. 2(4), 1115–1127 (2018)
6. Akan, O.B., et al.: Internet of hybrid energy harvesting things. IEEE Internet Things J. 5(5), 736–746 (2018)
7. Huang, X., et al.: Software defined networking for energy harvesting internet of things. IEEE Internet Things J. 5(3), 1389–1399 (2018)
8. Liu, D., et al.: Error bounds of adaptive dynamic programming algorithms for solving undiscounted optimal control problems. IEEE Trans. Neural Netw. Learn. Syst. 26(6), 1323–1334 (2015)
9. Ye, Q., et al.: End-to-end delay modeling for embedded VNF chains in 5G core networks. IEEE Internet Things J. 6(1), 692–704 (2019)
10. Ren, J., et al.: Joint channel access and sampling rate control in energy harvesting cognitive radio sensor networks. IEEE Trans. Emerg. Topics Comput. 7(1), 149–161 (2019)
11. Wu, C., et al.: Energy utilization efficient frame structure for energy harvesting cognitive radio networks. IEEE Wireless Commun. Lett. 5(3), 488–491 (2016)
12. Yang, Z., et al.: Energy efficient resource allocation in machine-to-machine communications with multiple access and energy harvesting for IoT. IEEE Internet Things J. 5(1), 229–245 (2018)
13. Zhang, Y., et al.: Data gathering optimization by dynamic sensing and routing in rechargeable sensor networks. IEEE/ACM Trans. Netw. 24(3), 1632–1646 (2016)
14. Khoshabi Nobar, S., et al.: Cognitive radio sensor network with green power beacon. IEEE/ACM Trans. Netw. 17(5), 1549–1561 (2017)
15. Michelusi, N., et al.: Transmission policies for energy harvesting sensors with time-correlated energy supply. IEEE Trans. Commun. 61(7), 2988–3001 (2013)
16. Blasco, P., et al.: A learning theoretic approach to energy harvesting communication system optimization. IEEE Trans. Wireless Commun. 12(4), 1872–1882 (2013)
17. Wu, W., et al.: Online throughput maximization for energy harvesting communication systems with battery overflow. IEEE Trans. Mobile Comput. 16(1), 185–197 (2017)
18. Xu, W., et al.: Energy management and cross layer optimization for wireless sensor network powered by heterogeneous energy sources. IEEE Trans. Wireless Commun. 14(5), 2814–2826 (2015)
19. Ali, S., et al.: Fast uplink grant for machine type communications: Challenges and opportunities. IEEE Commun. Mag. 57(3), 97–103 (2019)
20. Tapparello, C., et al.: Dynamic compression-transmission for energy-harvesting multihop networks with correlated sources. IEEE/ACM Trans. Netw. 22(6), 1729–1741 (2014)
21. Yang, B., et al.: Energy-efficient resource allocation for time-varying OFDMA relay systems with hybrid energy supplies. IEEE J. Syst. 12(1), 702–713 (2018)
22. Ramshaw, L., Tarjan, R.E.: On minimum-cost assignments in unbalanced bipartite graphs. Tech. Rep. HPL-2012–40R1. HP Labs, Palo Alto (2012)
23. Cui Y., et al.: A survey on delay-aware resource control for wireless systems-large deviation theory, stochastic Lyapunov drift, and distributed stochastic learning. IEEE Trans. Inf. Theory 58(3), 1677–1701 (2012)
24. Wu, W., et al.: Adaptive cross-layer resource optimization in heterogeneous wireless networks with multi-homing user equipments. IEEE J. Commun. Netw. 18(5), 784–795 (2016)
25. Xu, W., et al.: Energy management and cross layer optimization for wireless sensor network powered by heterogeneous energy sources. IEEE Trans. Wireless Commun. 14(5), 2814–2826 (2015)
26. Huang, L., Neely, M.J.: Utility optimal scheduling in energy-harvesting networks. IEEE/ACM Trans. Netw. 21(4), 1117–1130 (2013)

How to cite this article: Feng C, Yang Q, Qin M, Kwak KS. Dynamic online joint energy management and sampling rate control in energy harvesting aided IoT network. IET Commun. 2021;15:351–363. https://doi.org/10.1049/iet-com.2021.12058

APPENDIX A

A.1 Proof of Lemma 1

Based on $[\max(x, y, 0) + z]^{-2} \leq x^{-2} + y^{-2} + z^{-2} + 2x(z - y)$, we can obtain (A.1) by squaring both sides of (6).

$$[Q_n(t + 1)]^2 - [Q_n(t)]^2 \leq \left( \sum_{k \in K} \alpha_{n,k}(t) \mu_k(t) \right)^2 + \left( r_n(t) \right)^2 + 2Q_n(t) \left( r_n(t) - \sum_{k \in K} \alpha_{n,k}(t) \mu_k(t) \right).$$ (A.1)
Multiplying both sides by 1/2, and defining $\hat{B} = \frac{1}{2}(\lambda^2_{\text{max}} + \tau^2_{\text{max}})$, we have

$$
\frac{1}{2}([Q_n(t) + 1]^2 - [Q_n(t)]^2) \leq \hat{B} + Q_n(t)\left(r_n(t) - \sum_{k \in K} \alpha_{n,k}(t)\mu_n(t)\right). \quad (A.2)
$$

Using a similar approach, we get that

$$
\begin{align*}
\frac{1}{2}((E_n(t) + 1)^2 - (E_n(t))^2) &\leq \frac{(P^\text{total}_n(t))^2 + (e_n(t))^2 + 2(E_n(t) - \Theta)(e_n(t) - P^\text{total}_n(t))}{2} \\
&\leq B' + (E_n(t) - \Theta)(e_n(t) - P^\text{total}_n(t)),
\end{align*}
\quad (A.3)
$$

where $B' = \frac{1}{2}(P^2_{\text{max}} + \tau^2_{\text{max}})$.

Taking conditional expectations of (A.2)-(A.3) and summing over all $n$ gives a bound on $\Delta(Z(t))$. Adding the penalty term to both sides proves the results.

**A.2 | Proof of Theorem 1**

**A.2.1 | Proof of part (a)**

We prove the part (a) by comparing the Lyapunov drift with any other stationary and randomized algorithm denoted by ALT. According to [26], for any $\delta > 0$, there exists an i.i.d. algorithm ALT that satisfies

$$
E\left[\sum_{n \in N} U(r_n(\text{ALT}(t)))\right] \leq U^* + \delta, \quad (A.4)
$$

$$
E\left[\sum_{n \in N} (e^\text{ALT}_n(t) - E_{n}(t))\right] - \sum_{k \in K} \alpha_{n,k}(t)\mu_n(t) \leq \gamma_1\delta, \quad (A.5)
$$

$$
E\left[\sum_{n \in N} (P^\text{total}_n(t) - P^\text{total}_{\text{ALT}}(t))\right] \leq \gamma_2\delta, \quad (A.6)
$$

where $r^\text{ALT}(t)$, $e^\text{ALT}(t)$, $\alpha^\text{ALT}(t)$, $P^\text{total}_{\text{ALT}}(t)$ and $\mu^\text{ALT}(t)$ are the resulting values under ALT algorithm, and $\gamma_1$ and $\gamma_2$ are constant scalars.

We first show that the suboptimal algorithm approximately minimizes the RHS of (17). However, the solution procedure show that the suboptimal algorithm indeed minimizes the following function

$$
D(t) = \sum_{n \in N} (E_n(t) - \Omega)r_n(t) + \sum_{n \in N} Q_n(t)r_n(t) - \kappa N^2 + \Omega - VU(r_n(t))
$$

$$
+ \sum_{n \in N} \alpha_{n,k}(t)[-Q_n^{\text{mod}}(t)\mu_n(t) - \kappa T(E_n(t) - \Omega)],
$$

where $Q_n^{\text{mod}}(t) = [Q_n(t) - \lambda_{\text{max}}]^+$. Define $\tilde{D}(t)$ as follows

$$
\tilde{D}(t) = \sum_{n \in N} (E_n(t) - \Omega)\eta_n(t) + \sum_{n \in N} \alpha_{n,k}(t)[-Q_n^{\text{mod}}(t)\mu_n(t) - \kappa P(E_n(t) - \Omega)],
$$

$\tilde{D}(t)$ is the function within the expectation on R Ho the drift-minus-utility (17). By comparing $\tilde{D}(t)$ with $D(t)$, we have

$$
D(t) \leq \tilde{D}^\text{ALT}(t). \quad (A.9)
$$

Since the suboptimal algorithm minimizes $\tilde{D}(t)$. Here we use superscript SUB to denote the suboptimal algorithm. Then, we derive

$$
\tilde{D}^\text{SUB}(t) \leq \tilde{D}^\text{ALT}(t). \quad (A.10)
$$

Now we plug Equation (A.9) into the above Eq. and get

$$
\tilde{D}^\text{SUB}(t) - \sum_{k=1}^{K} \alpha_{n,k}(t)\lambda_{\text{max}}\mu_n(t) \leq \tilde{D}^\text{ALT}(t) - \sum_{k=1}^{K} \alpha_{n,k}(t)\lambda_{\text{max}}\mu_n(t). \quad (A.11)
$$

Since $0 \leq \sum_{n \in N} \sum_{k=1}^{K} \alpha_{n,k}(t)\lambda_{\text{max}}\mu_n(t) \leq NK(\lambda_{\text{max}})^2$, we have

$$
\tilde{D}^\text{SUB}(t) \leq \tilde{D}^\text{ALT}(t) + NK\lambda_{\text{max}}^2. \quad (A.12)
$$

That is, the value of $\tilde{D}(t)$ under the suboptimal algorithm is no greater than that under any other alternative policy plus a constant. Now using the definition of $\tilde{D}(t)$, (17) can be written as

$$
\Delta(t) - VU\left[\sum_{n \in N} (r_n(t)|Z(t)|)\right] \leq \tilde{B} + E[\tilde{D}^\text{SUB}(t)|Z(t)].
$$

Using the relationship between $\tilde{D}^\text{SUB}(t)$ and $\tilde{D}^\text{ALT}(t)$, we have

$$
\Delta(t) - VU\left[\sum_{n \in N} (r_n(t)|Z(t)|)\right] \leq \tilde{B} + E[\tilde{D}^\text{ALT}(t)|Z(t)],
$$

(A.13)
where \( B = B + N K \lambda_\text{max}^2 \). Now we plug (A.4) into the above Equation (A.13) and get
\[
\Delta(t) - V E \left[ \sum_{u \in \mathcal{N}} V U(r_u(t)) | Z(t) \right] \\
\leq B + (\gamma_1 + \gamma_2 + 1) \delta - V U^*.
\]
(A.14)

By letting \( \delta \) be zero and summing up the inequality over \( t \), we have
\[
\frac{L(T - 1) - L(0)}{T} = \sum_{t=1}^{T} \left[ \sum_{u \in \mathcal{N}} V U(r_u(t)) \right] \\
\leq B - V U^*.
\]
(A.15)

Letting \( T \to \infty \) we directly have (27).

**A.2.2 Proof of part (b)**

We use induction to prove the Equation (28).

1. First we prove it in the condition that the \( n \)-th IoTD does not collect any information from the monitoring area, then \( Q_n(t + 1) \leq Q_n(t) \leq \varphi_U V + r_{\text{max}} \) is clearly true;
2. Then we prove it when the \( n \)-th IoTD collects information with sampling rate \( r_u^*(t) \) given in Equation (21). According to Equation (21), we can derive \( V U^*(r_u^*(t)) = Q_n(t) - \kappa_\tau (E_n(t) - \Omega) \). Because the term \( E_n(t) - \Omega \) is negative, we have \( Q_n(t) \leq V U^*(r_u^*(t)) \). Since \( V U^*(r_u^*(t)) \leq \varphi_U V \), then \( Q_n(t) \leq \varphi_U V \). Furthermore, \( r_n(t) \) is upper bounded by the maximum sampling rate \( r_{\text{max}} \), then \( Q_n(t + 1) \leq Q_n(t) + r_{\text{max}} \leq \varphi_U V + r_{\text{max}} \) is satisfied.

**A.2.3 Proof of part (c)**

Here we first derive an expression for \( \Omega \) in such a way that IoTD does not sense any data if the available energy is less than the maximum energy consumption. Since the utility function \( U(r_u(t)) \) is concave, we get \( r_u(t) \) are inversely proportional to \( U(r_u(t)) \). Combined with the expression of sampling rate in Equation (21). We get that IoTD does not sense any data, i.e., the sampling rate is zero, if
\[
(u')^{-1} \left( \frac{Q_n(t) - \kappa_\tau (E_n(t) - \Omega)}{V} \right) \leq 0.
\]
(A.16)

Note that the maximum derivative of \( U(r_u(t)) \) is \( \varphi_U \) and equals \( U'(0) \), we have \( (u')^{-1}(\varphi_U) = 0 \), substituting the above formula into Equation (A.16), we derive that \( (u')^{-1}(\varphi_U - \kappa_\tau (E_n(t) - \Omega)) \leq (u')^{-1}(\varphi_U) \), recalling that \( U(r_n(t)) \) and \( r_n(t) \) are inversely proportional, we have
\[
\frac{Q_n(t) - \kappa_\tau (E_n(t) - \Omega)}{V} \geq \varphi_U
\]
(A.17)

Then, we rearrange Equation (A.17) to \( \Omega \geq \frac{V \varphi_U - Q_n(t) + E_n(t)}{\kappa_\tau} \), and with the fact that \( Q_n(t) \geq 0 \), we can derive \( \Omega \geq V \frac{\varphi_U}{\kappa_\tau} + E_n(t) \). To satisfy that the sensor cannot sense any data when \( E_n(t) < P_{\text{max}} \), \( \Omega \) can be written as \( \Omega \geq V \frac{\varphi_U}{\kappa_\tau} + P_{\text{max}} \).

By supposing no channel can be assigned to the \( n \)-th IoTD if the available energy is less than the maximum energy consumption, we can derive another value of \( \Omega \). As we can see from the objective function of channel allocation, no channel can be allocated to the \( n \)-th IoTD if
\[
Q_n^\text{mod}(t) \lambda_k n_k(t) - \kappa_\tau (E_n(t) - \Omega) \geq 0.
\]
(A.18)

Rearrange the Equation (A.18) to
\[
\Omega \geq \frac{Q_n^\text{mod}(t) \lambda_k n_k(t)}{\kappa_\tau} + E_n(t)
\]
(A.19)

Since \( Q_n^\text{mod}(t) \leq Q_{\text{max}} \) and \( \lambda_k n_k(t) \leq \lambda_{\text{max}} \), \( \Omega \) in Equation (A.19) can be written as \( \Omega \geq Q_{\text{max}} \lambda_{\text{max}} / \kappa_\tau + E_n(t) \). Then, we choose the maximization of the \( \Omega \), such that
\[
\Omega = \max \left( \frac{V \varphi_U}{\kappa_\tau} + P_{\text{max}} \frac{Q_{\text{max}} \lambda_{\text{max}}}{\kappa_\tau} + P_{\text{max}} \right).
\]
(A.20)