Optimal Resource Scheduling for Energy Harvesting Communications under Strict Delay Constraint

Ibrahim Fawaz, Mireille Sarkiss, Philippe Ciblat

To cite this version:
Ibrahim Fawaz, Mireille Sarkiss, Philippe Ciblat. Optimal Resource Scheduling for Energy Harvesting Communications under Strict Delay Constraint. 2018 IEEE International Conference on Communications (ICC 2018), May 2018, Kansas City, United States. 10.1109/ICC.2018.8422088. cea-01888837

HAL Id: cea-01888837
https://hal-cea.archives-ouvertes.fr/cea-01888837
Submitted on 5 Oct 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Optimal Resource Scheduling for Energy Harvesting Communications under Strict Delay Constraint

Ibrahim Fawaz\textsuperscript{1,2}, Mireille Sarkiss\textsuperscript{1}, and Philippe Ciblat\textsuperscript{2}
\textsuperscript{1} LIST, CEA, Communicating Systems Laboratory, F-91191 Gif-sur-Yvette, France
\textsuperscript{2} LTCI, Télécom ParisTech, Université Paris-Saclay, F-75013, Paris, France
Email: ibrahim.fawaz@cea.fr, mireille.sarkiss@cea.fr, philippe.ciblat@telecom-paristech.fr

Abstract—This paper investigates the resource scheduling minimizing the packet loss when the wireless communication system operates with Energy Harvesting (EH) based devices. The packet loss occurs when the buffer is overflowed and when the queued packet is older than a certain pre-defined threshold. We consider a strict delay constraint rather than an average delay constraint. The associated optimization problem can be modeled as Markov Decision Problem (MDP) where the actions are the number of packets sent on the known channel at each slot. The optimal deterministic offline policy is exhibited through dynamic programming techniques, i.e. Value Iteration (VI) algorithm. We show the gain in the number of transmitted packets and the consumed energy is substantial compared to a naive policy which forces the system to send the maximum number of packets using the available energy in the battery.

I. INTRODUCTION

Energy harvesting communications have recently attracted considerable attention in highly connected IoT and mobile cellular networks. Indeed, the energy harvesting technology has been proposed as a promising and viable solution to substantially extend the lifetime of mobile devices while reducing their dependency on conventional grid power. Hence, this improves self-sustainability of wireless networks and limits their growing carbon footprint. In such dense networks, energy harvesting-enabled devices can gather the energy from alternative, natural or man-made, energy sources in their surrounding environment to power their communications. Depending on the environment conditions, the harvested energy arrives in intermittent amounts at random times. It can be then stored in a capacity-limited energy storage device or battery for future use. The stochastic behavior and uncertainty of available energy in addition to the time-varying channel conditions may degrade the transmission performance due to intolerable latency and packet loss. Therefore, designing efficient data transmission policies, which adapt the rate and power according to channel and energy dynamics, is essential to satisfy users quality of service (QoS) and ensure the system reliability. In particular, optimizing resource scheduling of mobile devices becomes a more challenging issue under energy harvesting constraints.

During the past decade, extensive research efforts have focused on investigating resource scheduling with EH capabilities at the transmitters: [1], [2] for overview, and [3]–[12] for original works. In these works, several performance criteria have been optimized such as throughput, completion time, average delay, outage probability, for various models of energy arrival rate, battery capacity, or fading channel. Optimal approaches have been proposed based mainly on offline and online scheduling schemes, depending whether causal or non-causal channel state information (CSI) and energy state information (ESI) are available at the transmitter. More precisely, two problems were considered in [3]: throughput maximization by a given deadline and transmission completion time minimization, under channel and energy variations. The authors proposed optimal offline policies based on directional water-filling in a non-causal setting and online policies using continuous time stochastic dynamic programming in a causal setting. The throughput maximization problem was similarly investigated in [4] but for limited energy battery and limited data buffer, allowing thus buffer overflow. The optimal solutions were proposed by decoupling energy and data problems using a new variant of directional water-filling with added energy pumps, or applying recursively the shortest path algorithm. For causal state information, the same throughput maximization problem was modeled as Markov decision process in [5] and stochastic optimization techniques were used. [6] addressed the tradeoff between energy overflow and energy shortage by maximizing the scheduling throughput for capacity-limited EH system. Online algorithms were proposed to solve this problem for static and fading channels based on a new estimation method of future energy arrivals without any prior information. Both offline and online algorithms were also provided in [7] to maximize the throughput in finite-horizon scheduling with EH transmitter. The offline solution is expressed in terms of water levels and the online solution minimizes successively the expected throughput losses with respect to the offline optimal decision. Finite-horizon optimization problem was also considered in [8] to minimize the outage probability in a EH system. A low complexity fixed threshold transmission is proposed based on the offline mixed integer linear programming solution. In [9], an average delay optimal scheduling problem was studied.

This work has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 675891.
where the transmitter relies on hybrid energy supplies. In fact, the data transmission is mainly powered by harvested energy and resorts to power grid as a backup under energy consumption constraint. The problem was modeled as a two-dimensional Markov chain and an optimal transmission policy depending on a critical threshold of the queue length is proposed using Linear programming formulations. In [10], optimal deterministic scheduling in EH-powered network was obtained by minimizing the packet blocking probability, due to non-transmission at the transmitter, under an average delay (equivalently queue length) constraint and an average consumed energy constraint. The problem was formulated as an MDP and solved using dynamic programming Value Iteration algorithm. Moreover, [11] minimized the weighted packet loss rate under an average delay constraint in wireless sensor networks. The constrained MDP was formulated with linear value iteration approximation that locally determines the energy allocation at every EH wireless node using multilevel water-filling. Near-optimal control policy was derived by applying stochastic online learning based on post-decision state framework. In [12], MDP modeling and online post-decision learning approach were also derived to maximize the data arrival rate at the transmitter queue under delay and energy constraints. Two delay constraints were separately considered: average delay constraint and statistical delay constraint. This latter is a bounded delay with maximum acceptable delay-outage probability constraint.

In this paper, we address resource scheduling for a single user communication powered by energy harvesting. Unlike certain aforementioned works [9]–[11], we impose here a strict delay constraint on each queued packet stored in the buffer rather than an average delay constraint. This severe constraint has been introduced in our previous work [13] to find optimal scheduling policy that minimizes the average power consumption. We now incorporate energy harvesting aspects within the scheduling problem. Then, taking into account sporadic energy arrivals, random data arrivals and time-varying channel states, we minimize the packet loss rate, i.e., the average number of discarded packets, induced by strict delay constraint in addition to buffer overflow constraint. We formulate the problem as an MDP and solve it using Relative Value Iteration algorithm. We find an optimal offline stationary policy and compare it with a naive policy that performs immediate scheduling irrespective of energy and channel states.

The remainder of the paper is organized as follows. In Section II, we describe the system model. In Section III, we formulate the optimization problem as an MDP and solve it using value iteration algorithm. We provide and analyze numerical results in Section IV. Finally, we give some concluding remarks and future perspectives in Section V.

II. SYSTEM MODEL

We consider a point-to-point communication over a fading channel with an energy harvesting transmitter. The transmitter is equipped with two queues: one corresponds to a limited-capacity battery to store harvested energy from an external source and the other is a finite buffer to store data packets arriving from the upper layer. The communication is slotted into consecutive epochs of equal duration $T_c$.

A. Energy model

Due to the random nature of energy harvesting sources, we model the EH process as an independent identically distributed (i.i.d.) Poisson distribution with an average arrival rate $\lambda_e$. We assume that the energy arrives in multiple packets of energy units (e.u.) of $E_U$ Joules (J). The received energy is stored in a battery of limited capacity $B_e$, and is lost when it exceeds $B_e$. At the beginning of time slot $n$, let $e_n$ denote the harvested incoming energy (counting as a number of the energy units). Its probability distribution is given by

$$p(e_n = e) = e^{-\lambda_e} \frac{\lambda_e^e}{e!}.$$ 

We assume in this model that the processing energy is negligible compared to the transmission energy, thus the energy stored in the battery is only used for communication. We also consider the energy causality constraint where the system can only transmit if a sufficient amount of energy is available in the battery. Let $b_n$ denote the energy level of the battery at the beginning of time slot $n$, $b_n \in \{0, ..., B_e\}$, and $E_n$ the energy consumed to send packets during time slot $n$, then $E_n \leq b_n \forall n$. In addition, we suppose perfect energy state information at the transmitter (ESIT).

B. Data queue model and strict delay constraint

The transmitter receives also data packets and store them for future transmission in a data buffer of size $B_d$ packets. We model the data arrival process as an i.i.d. process following a Poisson distribution with an average arrival rate $\lambda_d$. We assume that all packets are of the same size $L$ bits. At the beginning of time slot $n$, let $q_n$ denote the queue length in the buffer, $q_n \in \{0, ..., B_d\}$, and $a_n$ the received packets with probability distribution

$$p(a_n = a) = e^{-\lambda_d} \frac{(\lambda_d)^a}{a!}.$$ 

A packet is discarded from the buffer

- if there is a buffer overflow, i.e., if the sum of packets in the queue and arrival packets exceeds the buffer size.

In that case, we discard the arrival packets in overflow;

- if there is a delay no-fulfillment, i.e., if stays in the queue more than $K_0$ slots. This can occur if the system decides not to transmit for a long period due to energy shortage or bad channel conditions.

In order to describe the delay no-fulfillment, we need to introduce a new variable $k_i(n)$ counting the time spent in the
buffer of the i-th packet at time n. By definition, we have
$k_i(n) \in \{-1, \ldots, K_0\}, \forall i, k$ and $k_i(n) = -1$ for an empty
space in the buffer (i.e., when the i-th packet does not exist). 
In Fig. 1, we provide a buffer state at time n.

Buffer of $B_d$ packets (ordered from the oldest to the newest)

\[ k_1(n) \quad \ldots \quad k_{q_n}(n) \quad -1 \quad \ldots \quad -1 \]

$q_n$ packets empty area

Fig. 1: Buffer configuration at slot n.

Notice that $k_j(n) \leq k_i(n), \forall i \leq j$.

C. Channel model and consumed energy

We consider a single user flat-fading channel with signal
bandwidth $W$ (Hz) and additive white Gaussian noise with
power spectral density $N_0$. During time slot $n$, the channel
remains constant with complex-valued amplitude $h_n$, and
varies independently across time slots. We define the channel
gain as $x_n = |h_n|^2$. We assume $\{x_n\}_n$ is an i.i.d. process
following a uniform distribution into a finite set $\mathcal{X}$. We also
assume perfect Channel State Information at the Transmitter
(CSIT).

We denote $u_n$ ($u_n \leq q_n$) the number of packets to be
transmitted during time slot $n$ of period $T_s$, through the
channel of gain $x_n$. The consumed energy to transmit these
packets is expressed as an integer multiple of the energy unit.
It is given by

\[ E(x_n, u_n) = \left\lceil \frac{P(x_n, u_n) T_s}{\mathcal{E}_U} \right\rceil \]

(1)

where

\[ P(x_n, u_n) = \frac{W N_0}{x_n} \left( 2^n u_n - 1 \right). \]

(2)

is the required power for this transmission.

III. PROBLEM FORMULATION AND RESOLUTION

Our main objective now is to ensure reliable communication by
minimizing the number of discarded packets due to strict
delay and buffer overflow constraints. This can be achieved
by finding an optimal policy that specifies the number of
packets $u$ to be scheduled at each time slot based on the past
system states and actions. The optimization problem can be
formulated as MDP problem [14]. We characterize in this
section the appropriate states, actions and reward of this MDP.

A. State Space

The state space $\mathcal{S}$ is the set of $s = (k, b, x)$ where

- $k = [k_1, \cdots, k_{B_d}]$ is the vector indicating the age of each
  packet in the data buffer,
- $b$ is the battery level, and
- $x$ is the channel gain.

Notice that in the previous works [10], [11], the queue
length $q$ describes the data buffer states. In our work, $q$ is
replaced with  $k$ due the strict delay constraint. In fact, $q$
is unnecessary when $k$ is given since

\[ q_n = \max \left\{ i \mid k_i(n) \geq 0 \right\}. \]

(3)

The state space is finite, and the total number of possible
states is $|\mathcal{S}|$ which is upper-bounded by $(K_0 + 2) B_d |B_e +
1, |\mathcal{X}|$. The state space can be significantly reduced by
assuming that packets are queued in an increasing order of time
spent in the buffer, i.e. $k_1(n) \geq k_2(n) \geq \cdots \geq k_{q_n}(n)$.
For instance, if we consider $B_d = 6, K_0 = 3, B_e = 4$ and $|\mathcal{X}| = 5$,
the upper-bound is 390625 while our system only has 5250
states by removing the impossible combination of components
in $k$.

B. Action Space

The action space $\mathcal{U}$ denotes the number of packets $u$ that
the transmitter can send during a time slot. This space is finite
and the number of actions is $|\mathcal{U}| = U_0 + 1$. $U_0$ is the maximal
value of scheduled packets. It is obtained from Eq. (1) with the
maximum available power $P_{\text{max}}$, the maximum capacity of the
battery $B_e$ and the best channel coefficient $x_{\text{max}} = \max_{x \in \mathcal{X}} x$.

C. Markov Decision Process

On one hand, during time slot $n$, $w_n = \max(u_n, m_n)$
packets leave the buffer, either transmitted and/or discarded
where $u_n$ is the number of transmitted packets and $m_n$ is
the number of packets with delay $K_0$ slots in the buffer. The age
of the remaining packets in the buffer is incremented by 1.
Moreover, $a_{n+1}$ new packets arrive to the buffer with age 0.
Therefore, the vector $k$ can be updated from slot $n$ to slot
$n + 1$ according to the following rule.

1: for $i = 1$ to $q_n - w_n$ do 
   $k_i(n+1) = k_{u_n}(n) + 1$
end for

2: for $i = q_n - w_n + 1$ to $q_n - w_n + a_{n+1}$ do 
   $k_i(n + 1) = 0$
end for

3: for $i = q_n - w_n + a_{n+1} + 1$ to $B_d$ do 
   $k_i(n + 1) = -1$
end for

On the other hand, during time slot $n$, $e_{n+1}$ e.u are harvested
and stored in the battery and $E_n$ e.u are removed from the
battery to schedule $u_n$ packets. Therefore, at time slot $n + 1$,
the battery state is updated according to

\[ b_{n+1} = \min \left\{ b_n - E_n + e_{n+1}, B_e \right\}. \]

(4)

We thus remark that $b_{n+1}$ (resp. $b_{n+1}$) only depends on
previous state $k_{n}$ (resp. $b_{n}$), action $u_{n}$ (resp. $E_{n}$) and external
perturbation $a_{n+1}$ (resp. $e_{n+1}$). Therefore, we can define
$p(s'|s, u)$ as the transition probability to fall in the future
state $s' = (k', b', x')$ after taking action $u$ in the current state
$s = (k, b, x)$. Assuming that the buffer, battery and channel
states are independent and channel states are not correlated, the transition probability satisfies the following equation.

\[ p(s'|s, u) = p(k'|k, b, u) p(b'|b, x, u) p(x'), \]

(5)

where \( p(x') \) is the distribution of the channel states, \( p(k'|k, b, u) \) indicates the probability transitions between buffer states, and \( p(b'|b, x, u) \) indicates the probability transitions between battery states. After tedious but simple derivations, we obtain the transitions between the buffer states and the battery states according to the following respective rules.

1. if \( u > q \) or \( k_i' > k_i + 1 \) or \( q' < q - w \) then 
   \[ p(k_i'|k_i, b, u) = 0 \]
2. else if \( k_i' \neq k_i + u + 1 \) and \( k_i + u 
eq -1 \) then 
   \[ p(k_i'|k_i, b, u) = 0 \]
3. else if \( k' > 0 \) and \( k_{i+u} = -1 \) then 
   \[ p(k_i'|k_i, b, u) = 0 \]
4. else if \( q = B_d \) and \( u 
eq 0 \) and \( k_i' > 0 \), \( \forall i \in \{q - w + 1, ..., B_d \} \) then 
   \[ p(k_i'|k_i, b, u) = 0 \]
5. else if \( q' < B_d \) then 
   \[ p(k_i'|k_i, b, u) = e^{-\lambda_d} \lambda_d^{u} u! \]
6. else 
   \[ p(k_i'|k_i, b, u) = 1 - Q(B_d - q + w, \lambda_d), \]

and

1. if \( E > b \) then  
   \[ p(b'|b, x, u) = 0 \]
2. else if \( b' < b - E \) then  
   \[ p(b'|b, x, u) = 0 \]
3. else if \( b' < B \), then  
   \[ p(b'|b, x, u) = e^{-\lambda_s} \lambda_s^{u} u! \]
4. else  
   \[ p(b'|b, x, u) = 1 - Q(B_c - b + E, \lambda_c) \]

where \( Q \) is the regularized gamma function.

D. Markov Decision Problem and its Resolution

In the context of infinite-horizon MDP, we consider time-averaged cost, where at a given time slot \( n \in \{0, \cdots, N\} \), the system state is denoted by \( s_n = (k_n, b_n, x_n) \) and \( \mu(s_n) = u_n \) is the action deciding the number of packets to be transmitted. We aim at finding the optimal policy \( \mu^* \) that minimizes the average number of dropped packets. The cost function of this infinite-horizon MDP problem is given by

\[ \mathcal{D}(\mu) = \lim_{N \to \infty} \frac{1}{N} E^\mu \left[ \sum_{n=1}^{N} \left( \varepsilon_d(s_n, u_n) + \varepsilon_o(s_n, u_n) \right) \right], \]

(6)

where \( E \) is the expectation with respect to the policy \( \mu \) and where \( \varepsilon_d(s_n, u_n) \) is the instantaneous number of discarded packets due to delay no-fulfillment and \( \varepsilon_o(s_n, u_n) \) is the instantaneous number of discarded packets due to buffer overflow.

At a given slot \( n \), when the system state is \( s_n \) and the performed action is \( u_n \), the number of discarded packets due to delay no-fulfillment is given by

\[ \varepsilon_d(s_n, u_n) = \begin{cases} 
0 & \text{if } m_n = 0 \text{ or } m_n < u_n \\
|s_n| - u_n & \text{otherwise}.
\end{cases} \]

(7)

The buffer overflow occurs when \( q_n - w_n + a_{n+1} > B_d \), thus the number of discarded packets due to buffer overflow is obtained as follows

\[ \varepsilon_o(s_n, u_n) = \sum_{a=B_d-q_n+w_n+1}^{\infty} (q_n - w_n + a - B_d) e^{-\lambda_d} (\lambda_d)^{a} \frac{1}{a!} \]

\[ = \lambda_d (1 - Q(B_d - q_n + w_n, \lambda_d)) + (q_n - w_n - B_d) \times \left( 1 - Q(B_d - q_n + w_n + 1, \lambda_d) \right). \]

(8)

Finally, our MDP optimization problem can be stated as 

Problem 1: 

\[ \mu^* = \arg\min_{\mu} \mathcal{D}(\mu) \]

(9)

To solve this optimization problem, we resort to the offline dynamic programming approach using the so-called VI algorithm [14]. Exploring a priori knowledge of energy arrival and data arrival dynamics and channel states at the EH transmitter, the offline approach can accurately model the state transition probabilities of the MDP and provide an optimal solution. The optimal offline deterministic policy, exhibited by Algorithm 1, consists in a one-to-one mapping from the state space \( S \) to the action space \( U \), performing a unique action \( u \) whenever a state \( s \) is visited.

**Algorithm 1 VI algorithm**

1: Initialization
   
   Set \( v^0(s) = 0 \quad \forall s \in S \)

   Fix a tolerance parameter \( \varepsilon > 0 \)

   Set \( n = 1 \)

2: For each \( s \in S \) compute

\[ v^n(s) = \min_{u \in U} \left[ c(s, u) + \sum_{s' \in S} P(s'|s, u) v^{n-1}(s') \right] \]

(10)

\[ h^n(s) = v^n(s) - v^n(s_0) \]

(11)

where \( s_0 \) is a fixed state chosen arbitrarily.

3: If \( sp \left( v^n - v^{n-1} < \varepsilon \right) \) then \( \mu^* = \max_{\mu} \mathbb{E}_{s \in S} \left[ v(s) \right] \), let \( \pi^* \) be the resulting policy that solves equation (10), \( v^{*} = v^* \) and stop; else set \( n = n + 1 \) and go to step 2.

IV. NUMERICAL RESULTS

We evaluate numerically the optimal policy obtained by solving Problem 1. We consider a system as described in Section II with the following characteristics: the slot duration is \( T_s = 1 \) ms, the channel states \( x \) takes 5 possible values from the finite set \( X = \{-5.41, -1.59, 0.08, 1.42, 3.18\} \) dB with equal probabilities. The noise power spectral density is \( N_0 = -87 \) dBm/Hz and the allocated bandwidth is \( W = 5 \) MHz. Data arrivals follow a Poisson distribution with mean \( \lambda_d \) packets, where packets are of equal size \( L = 5000 \) bits. Data packets are stored in a buffer of size \( B_d = 6 \)
packets. The maximum delay is $K_0 = 3$ (i.e., in absolute time $K_0 T_s = 3$ ms). Energy arrivals follow a Poisson distribution with mean $\lambda_e$ e.u per slot, where $\mathcal{E}_U = 100$ nJ. Energy units are stored in a battery of size $B_e = 4$ e.u. The maximum available power at the transmitter is $P_{\text{max}} = 2$ mW. According to Eq. (2), this system allows to transmit up to 8 packets per slot. Limited by the queue size and the capacity of the battery, we fix $U_0 = 6$ packets per slot.

In Fig. 2, we plot the average rate of discarded packets versus the number of iterations for evaluating the optimal policy within the VI algorithm for various energy arrival rates $\lambda_e$ where the data arrival rate $\lambda_d$ is fixed to 1.5. We show that the VI algorithm converges rapidly within a few hundreds iterations for most of cases. We can also notice that as $\lambda_e$ increases, the average number of discarded packets considerably decreases. Indeed, when the energy available from the surrounding environment is in larger quantities, the system will be able to send more packets, reducing thus the number of discarded packets.

In Fig. 3, we display the percentage of discarded packets versus the data arrival rate $\lambda_d$ for different energy arrival rates and both policies, namely the optimal one exhibited in this paper and the naive one which forces the transmitter to send the maximum number of packets using the available energy in the battery. As we can observe, the proposed optimal policy gives significantly better performance than the naive one in terms of percentage of discarded packets. In fact, this policy enables to adapt the transmission rate according to the channel conditions. The number of discarded packets increases when the data arrival rate $\lambda_d$ increases because the buffer overflow could happen more often. On the other hand, when the energy available to scavenge is low (small $\lambda_e$), an efficient energy management becomes crucial to ensure the sustainability of the system, and the gap between both policies increases. On the contrary, when a large amount of energy is available (large $\lambda_e$), the system can survive even without controlling relevantly the energy consumption which leads to similar performance between the optimal and naive policies.

In Fig. 4, we show the percentage of discarded packets due to delay among the total number of discarded packets for different values of the energy arrival rate $\lambda_e$ and the data arrival rate $\lambda_d$. As explained before, a packet can be discarded due to delay or buffer overflow. When the data arrival rate increases, the probability to discard a packet due to overflow increases, resulting in a lower contribution of the delay in discarding packets. On the other hand, when the energy rate decreases, the percentage of discarded packets due to the delay slightly increases because, in average, a packet remains more often in the buffer since there is not enough energy to transmit it. Hence, it is flushed from the buffer for latency’s purpose.

In Fig. 5, we plot the average consumed energy versus the data arrival rate $\lambda_d$ for different energy arrival rates $\lambda_e$. We observe that the optimal proposed policy consumes less energy while sending more packets because it adapts the number of transmitted packets per slot to the channel conditions and the battery state and thus, the transmission is done according to the energy it consumed.

In Fig. 6, we show the average battery state versus the packet arrival rate $\lambda_d$ for different energy arrival rates $\lambda_e$. As the proposed policy offers a lower energy consumption (see Fig. 5), the battery is less used and its energy level is thus higher. This ensures a better sustainable communication with less number of discarded packets.

V. CONCLUSION

We have addressed resource scheduling problem under energy harvesting capabilities with strict delay constraint. More precisely, we have solved the packet loss optimization problem using MDP framework and dynamic programming techniques. The optimal policy adjusted the number of transmitted packets according to the channel conditions and the available energy in the battery, such that the number of discarded packets...
is minimized. As an extension to this work, we aim i) to focus on online programming instead of offline, ii) to include offloading capabilities, where the system can choose to execute packets locally, offload it to nearby servers or base stations having more resources according to the system and channel conditions.

REFERENCES


