Djiroun, Fatima; Bagaa, Miloud; Taleb, Tarik

A Markov Decision Process-based Collision Avoidance in IoT Applications

Published in:
2017 IEEE Global Communications Conference, GLOBECOM 2017

DOI:
10.1109/GLOCOM.2017.8253931

Published: 01/12/2017

Please cite the original version:
A Markov Decision Process-based Collision Avoidance in IoT Applications

Fatima zahra Djiroun 1/2, Miloud Bagaa 2 and Tarik Taleb 2

1 LRDSI Laboratory, Saad Dahlab Blida1 University, Blida, Algeria
2 Dep. of Communications and Networking, School of Electrical Engineering, Aalto University, Espoo, Finland

Emails: {fatima.djiroun, miloud.bagaa, tarik.taleb}@aalto.fi

Abstract—IoT covers various scales and types of wireless networks. The first constraint to be respected, for an efficient application, is to reduce as much as possible the amount of energy consumption. The idle listening process in existing Medium Access Control (MAC) protocols is a very energy consuming task. Recently, a new emerging technology based on a low power wake-up radio has shown real benefits by completely eliminating the problem of idle listening. Thanks to the use of this technology, an IoT device keeps its main radio in deep sleep until a wake-up message is received by the wake-up radio that consumes less energy. However, collision can occur among wake-up messages (i.e., wake-up plane). Collision in the wake-up plane, if not handled efficiently, leads to collision at the data plane which is more complicated. In this paper, we address this issue by modeling the wake-up decision using a Markov Decision Process (MDP). The goal is to formulate a decision policy that determines whether to send a wake-up message in the actual time slot or to report it, taking into account the time factor. Experiments have been conducted to determine the decision policies. Results of the proposed approach have been compared against those of RFIDImpulse, a CSMA/CA-based wake-up MAC protocol. The obtained results show the efficiency of the proposed approach.

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have emerged recently and attracted both academic and industry attention as an interesting research area. It retrieves its power from its combination with IoT technology, that is gaining ground with its potential on making our daily life as smart as possible. UAVs, also known as drones, have been rapidly evolved and are now deployed in diverse sectors of our daily life. This is due to their small size and capability to fly without an on-board pilot, allowing to perform a wide range of activities from delivering a package, crowd surveillance, to diving into water for a specific underwater operations [1]. The applications of UAVs can be broadly divided into civilian and military models. The former can be utilized for governmental or non-governmental purposes [2]–[6].

One interesting application is the employment of UAVs in the field of agriculture for the role of crop management [7]–[9]. Regular aerial monitoring of agricultural lands allows deep analysis of crop performance. With the help of the different IoT devices (sensors and actuators) deployed either on the drone or on the ground, the health of these crops could be studied and immediate automatic actions could be taken accordingly. In this kind of applications, the ground sensors which are forming a wireless sensor network, regularly collect relevant data related to the studied phenomenon and report them to the UAV when it is flying in the neighborhood, in order to take strategic decisions. If a given ground sensor needs to announce some urgent data such as fire detection and flood monitoring, and that no drone has been captured in the neighborhood for a precise period of time, it sends this data to some alternative nodes which are assumed to have higher probability of being regularly in contact with flying UAVs. This not only leads to healthy crop growth, but also increases their yield.

Guaranteeing a long lifetime, and thus a solid and a reliable application is strongly dependent on ensuring long operation periods of the deployed IoT devices, without any human intervention after the initial deployment. This could only be guaranteed by assuring an energy efficient communication scheme. Avoiding Idle listening is thus considered mandatory to achieve this goal [10], [11]. The integration of an additional low power wake-up radio [12] behind the IoT device’s main radio is considered as a promising solution that completely eliminates the problem of idle listening [13], [14]. This is thanks to the fact that the main radio of the device could enter into the deep sleep mode whenever there is no transmission destined to it, while the low power wake-up radio is kept listening to the channel in order to detect possible incoming transmissions. The wake-up radio is either a passive circuit powered by the energy of the wake-up signal [15] or a semi-active one consuming few µ watts of power [16].

Adopting the wake-up radio technology in the aforementioned application may have a great impact on the energy efficiency, the fact that ground sensor nodes will never need to perform duty cycling to sense whether there is some drone flying in the vicinity. Instead they could stay in sleep mode as no drone is present in the neighborhood. Once the drone is within the transmission range of the intended node, it wakes it up using a wake-up signal which will be received by the low power wake-up radio. At this time, the sensor node switches on its main radio and starts transmitting the collected data toward the drone. Once data transmission has finished, the main radio goes back to sleep mode, saving thus energy.

When drones are flying, gathering data from ground sensors, it may happen, due to several control conflicts, that more than one drone is trying to wake up the same sensor node. Receiving several wake-up signals at the same time leads to the well known problem of collision. This could also happen when some ground sensor nodes are exchanging some urgent data, due to lateness of some drones. Collision is considered
as another source of energy waste, since the energy depleted in transmission and reception of collided messages is just lost. Several retransmissions should take place, which results in supplementary energy consumption. This has been considered very costly, mainly for energy-constrained networks. However, non of the previous works have considered this problem, when implementing wake-up radio-based schemes. In this paper, we address the problem of collision at the wake-up plane, i.e between wake-up messages, by proposing an infinite horizon time Markov Decision Process (MDP). The aim is to derive a decision policy that determines whether a drone, when it is in the vicinity of the destined ground sensor node, should transmit a wake-up signal to initiate data gathering or postpone it for the next time slot. The same decision should be taken by the ground node before waking up its neighbor in case of exchanging urgent data. In addition to collision handling and energy efficiency guaranteeing, the driven policy should take the time factor into high consideration, avoiding thus large delays that may result from unnecessary postponing of decisions.

The reminder of the paper is structured as follow. Section II summarizes briefly how the relevant research work, proposed in the recent literature have handled the problem of collision in wake-up radio-based schemes. In Section III, we present the problem of collision in wireless networks and formulate the model based on MDP framework. The conducted experiments and the results analysis are presented in Section IV. Finally, the paper concludes in Section V.

II. RELATED WORK

A lot of research work has been conducted on wake-up radio in the literature [17]. However, just some of them discuss the problem of collision at the wake-up plane. In a few research work investigating the collision issue, only basic and traditional mechanisms, which are based on CSMA/CA protocol, have been adopted, such as carrier sensing [12] and random binary back-off. The work introduced in [18] was based on the IEEE 802.15.4 standard [19]. For each device in the network, an RFID reader and a tag are integrated behind the main radio. The RFID reader is used to wake up the RFID tag that represents a wake-up radio. When there is a packet to transmit, the sender, using its RFID reader starts by performing a clear channel assessment in a random chosen slot within the contention window. If the channel is found free, the wake-up message will be sent; otherwise a random back-off mechanism will be followed until the channel becomes idle, or the maximum listening duration will be reached. Authors in [20] proposed, for the wake-up radio, a scheme following the IEEE 802.11 protocol. The introduced work includes both physical and virtual carrier sensing as well as the RTS/CTS control packet exchange, before transmitting a wake-up message. Physical carrier sensing is achieved by sensing the channel to check whether it is idle or busy. Virtual carrier sensing is performed at each packet reception, in which the transmission duration is included. From this information, the receiver, that is not the intended one, could be aware of the duration for which the channel will be busy, and thus it prohibits itself from accessing the channel during that period.

In order to avoid collision at the wake-up plane, some other research work have proposed the use of multi channel techniques for the wake-up radio [21], [22]. In [21], different frequency channels have been assigned to 2-hops neighboring devices. When the transmitter has a packet to transmit to a receiver, it tunes its wake-up radio channel to the receiver’s frequency channel and monitors it for a predefined period of time. If the channel is found free, the wake-up radio starts transmitting the wake-up message; otherwise, it should wait until the channel becomes free. Adopting multi-channel schemes could not alleviate the problem of collision at the wake-up plane. This could always happen when more than one nodes try to wake up the same receiver. A different research work [23] proposed the use of TDMA-like scheduling. For each device a schedule table is managed and maintained by a central point. The latter calculates the wake-up time of each device according to the packet generation rate. Another work, following also a TDMA scheme, has been proposed for bats tracking and monitoring in [24]. Scheduling communications based on TDMA protocol may avoid collision in small networks. However, it could not be applied in large scale networks, where the scheduling task becomes more complicated due to synchronization problems which result in high energy consumption.

Adopting basic mechanisms, which are based on random operations such as the random binary back-off techniques, for collision avoidance could be considered helpful. However, it is not the most efficient way to handle collision. In fact, in contention based schemes, when a device has a packet to transmit, it initializes a contention window with a minimal value and then chooses randomly a given slot within this window to perform a back-off. After each failed transmission, the contention window will be doubled up to a maximum value. When the number of devices involved in collision is important, collision persists and its probability becomes higher. This is attributable to the fact all devices are trying to retransmit in the same fashion (i.e., increasing the contention window and choosing a back-off value randomly). The key behind this issue in contention based scheme is that initially, and regardless the size and density of the network, all devices have the same probability to access the channel (i.e., the same contention window value), which is static during the network lifetime. This raises the need to develop other solid models, which must take into account the network dynamics and should be based on deep studies related to several factors (e.g., network density and packet arrival probability) to efficiently handle the problem of collision at the wake-up plane and ensure better fairness. In contrast to all previous works, in this paper, the wake-up decision is based on robust mathematical studies. Since in real wireless networks, wireless devices take actions and then transit from one state to another in a stochastic environment, this could be exactly imitated by modeling the network using the mathematical MDP framework that allows optimizing the network’s objectives in an efficient way.
III. System Model

A. Problem Description

We consider the problem of collision at the wake-up plane. As has been mentioned earlier, each ground node is dotted with a wake-up radio, situated behind the main radio and responsible for waking it up when a wake-up message is received. To reduce potential collision between wake-up messages and data packets, two different frequency channels could be used, one for the main radio and the other for the wake-up radio. When a transmitter A, which could be either a drone or a ground sensor node, wants to transmit a data packet to a receiver B representing a ground sensor node, it starts by transmitting, using its main radio a wake-up message at the wake-up radio frequency channel asking the receiver to wake up and be ready to transmit or receive data. The wake-up message is a signal including the address of the sender and the receiver and could be used by the receiver’s wake-up radio to power on itself and trigger the micro-controller which in turn wakes up the main radio [12] [25]. When the same ground sensor node receives more than one wake-up signals at the same time, collision happens and the receiver cannot send back an acknowledgment since it cannot detect the transmitter. Retransmissions should then take place until an acknowledgement is received. This results in a large amount of energy consumption, mainly that wake-up messages should be transmitted using the main radio of the sender. Collision at the wake-up plane, if not handled, may lead to collision at the data plane which is more energy consuming. However, if the wake-up message is successfully received, the receiver, using its main radio, broadcasts a message to its neighborhood indicating that it is receiving data from device A. This allows device A to start data transmission and prevents all others from transmitting during a predefined period of time.

B. Model Formulation

To solve the problem of collision at the wake-up plane, we model the system as a Markov Decision Process (MDP). The proposed model decides on whether a wake-up message should be transmitted in the actual time slot or reported to the next one. The MDP decision procedure is supposed to be implemented on UAVs as well as on ground sensor nodes. To derive the wake-up decision policy, we develop an infinite horizon MDP that associates actions, corresponding transition probabilities and rewards to each state. In order to improve the overall network performance, the proposed MDP models all the network, whereby each state represents not only the device condition but all the network behavior, taking in each state the action of the device that is ready to transmit a wake-up message.

We suppose the decision process starts when a UAV reaches the communication range of the intended node. In some cases, this process starts when an urgent data packet arrives at a given ground sensor node which will forward it to one of its neighbors. In order to formulate a state space, we assume each device (either a ground sensor node or a drone) willing to transmit a wake-up message, takes into consideration the other devices’ actions. We denote by $A=(tr, w)$ the vector representing the possible actions available for the decision maker at each state. Action $tr$ designates that the agent decides to transmit a wake-up message, while $w$ indicates that a decision of waiting has been taken. Taking into account the possible actions of the other devices, five different states could be driven (Fig. 1). The first one, referred to as successful transmission, reflects the network where only the decision maker decides to transmit a wake-up message. The agent tries always to increase his chances to enter this state. The second state, called collision, represents the network where the decision maker and some other devices decide to transmit simultaneously a wake-up message to the same destination. The three remaining states, represented by $idle_1$, $idle_2$ and $idle_3$ could be reached when the action of wait has been chosen by the decision maker, while different actions have been taken by the other devices. In case only one device from the $(n-1)$ remaining devices has decided to transmit, so the others have chosen to wait, the agent is said to be in state $idle_1$. When more than one of the remaining devices have decided to simultaneously transmit, collision between them happens. This leads the system to be in state $idle_2$. State $idle_3$ represents the network where both the decision maker and the remaining devices have decided to choose the action of wait. Hence the state space is defined as $S=\{ successful\ transmission, collision, idle_1, idle_2, idle_3 \}$.

In order to derive transition probabilities, we consider the network composed of a set of $n$ devices including UAVs and ground sensor nodes. Let $p$ be the probability of packet transmission rate which follows a Poisson distribution. We denote by $Pr$ the transition probability matrix with $Pr(s'|s, a)$ indicating the transition rate between states $s$ and $s'$ in $S$ taking a given action $a$, which in the proposed scenario, represents the network channels situation passing between
successful transmission, collision or idle states, when \( n \) devices take their actions. Thus, transition probabilities can be given as presented in (1).

For \( t \in T \), let \( s_t, a_t \), and \( r_t \) denote state, action and reward at time \( t \) of the proposed MDP, respectively. Let \( P_{(s,s')}^a=pr[s_{(t+1)}=s'|s_t=s, s_{(t+1)}=s', a_t=a] \) indicates the transition probabilities, and \( R_{(s,s')}=E[r_{(t+1)}|s_t=s, s_{(t+1)}=s'] \) designates the expected reward associated to the transitions. A policy \( \pi \), denoted as \( a_t=\pi(s_t) \), is defined as a mapping that associates for each state \( s \) a given action \( a \). Consequently, a policy \( \pi=\pi_1, \pi_2, \pi_3, ...\pi_T \) represents the sequence of taken decision rules at all decision epochs. When the agent is in a given state, it has to decide to transmit a wake-up message in this time slot using action \( tr \) or waiting for the next time slot following action \( w \). For each transition, given the state \( s \) where the agent actually is, the state \( s' \) where the agent will move on, and the chosen action \( a \), a reward is achieved. This reward associates to each action at each state a precise value that is a function of the outcomes received from the actual and the next states, respectively.

Let \( C(s) \) be the function that associates for each state a cost. When a successful transmission occurs, i.e when a certain constant \( g \) and the cost needed to transmit a wake-up message. At the other side, if a collision happens, this means that the transmitting device has only wasted its energy, and thus the cost at this state is the amount of energy consumed when transmitting a wake-up message. Since the energy is just wasted, so the cost will be a negative value indicating that the device has lost by entering this state. Since \( idle_1, idle_2 \) and \( idle_3 \) represent states in which the device has preferred not to transmit, and in order to take into consideration the time factor and avoid extra delay, we consider the device executing this state is receiving a negative cost represented by the amount of time needed to transmit a wake-up packet. This is for the goal of limiting the waiting time, as choosing the action wait may lead to time loss. Taking into consideration the fact that state \( idle_1 \) assumes that a successful transmission happens in the other side of the network (between the \( (n-1) \) remaining devices) and so, the cost of this state is the sum of the time lost by choosing the action \( w \) (by the decision maker) and a certain percentage (represented by \( \alpha \)) of the gain achieved when the \( (n-1) \) devices perform a successful transmission. In the same way of reasoning, in state \( idle_2 \), a collision happens between the \( (n-1) \) devices. Since this has an impact on the whole network, so the cost when entering this state is always the time lost by the decision maker summed to a certain ratio (3) when collision happens between \( (n-1) \) devices. In state \( idle_3 \), and since all devices choose not to transmit, this may affect the delay. Therefore, the cost is the time lost by the actual agent added to a certain ratio (\( \delta \)) of the time lost by the others. The possible costs are given as follows:

\[
C(s) = \begin{cases} 
   g - E & s = \text{successful transmission} \\
   -E & s = \text{collision} \\
   -I + \alpha (g - E) & s = \text{idle}_1 \\
   -I - \beta E & s = \text{idle}_2 \\
   -I - \delta I & s = \text{idle}_3 
\end{cases}
\]

where \( E \) is the energy consumed when transmitting a wake-up packet. It is expressed as:

\[
E = M_{\text{length}} \times T_b \times c_t \times h
\]

where \( M_{\text{length}} \) is the length of the wake-up message, \( T_b \) is the time needed to transmit one byte, \( c_t \) is the current drawn when the radio is in transmit mode, and \( h \) is the supply voltage. From (2), \( I \) is the time needed to transmit a wake-up message. It is defined as:

\[
I = M_{\text{length}} \times T_b
\]

It is clear that \( I < E \), and so there is always a gain when being in an idle state, in comparison to being in collision state. \( \alpha, \beta \) and \( \delta \) are thresholds used to improve the network performance. \( \alpha, \beta \) and \( \delta \in [0,1] \).

The reward function is thus a mapping that associates for each tuple \((s, a, s')\) a value \( v \in \mathbb{R} \). It is defined as the sum of the cost received when being in the actual state \( s \) and the cost obtained when moving to the next state \( s' \), taking action \( a \). The reward is given as follows:

\[
r(s, a, s') = \sum_{i \in (s, s')} C(i)
\]

Given an initial state \( s \) and a discount factor \( \gamma \in [0,1] \), the expected total discounted reward for the policy \( \pi=\pi_1, \pi_2, \pi_3, ...\pi_T \) is given as follows:

\[
V_\pi^* = \lim_{T \to \infty} \frac{E_{\gamma}}{T} \{ \sum_{t=1}^{T} \gamma^{t-1} r_t \}
\]

Let \( V^*(s) = \max_{\pi \in \Pi} V^*(s) \) denote the maximum discounted total reward, given the initial state \( s \). From [26], the optimality equations are given by:
In this paper, we introduced WupMDP, a MDP-based model for collision handling at the wake-up plane in wireless sensor networks. The proposed model permits the determination of efficient policies that allow to avoid collision between wake-up messages and thus conserve the energy wastage due to this issue. By implementing this model, a device can decide whether to transmit a wake-up message in the actual time slot.
Successful transmission gain

References


