Reinforcement Learning based Effective Communication Strategies for Energy Harvested WBAN

Moumita Roy[a], Dipanjana Biswas[a], Nauman Aslam[b] and Chandreyee Chowdhury[a,*]

[a]Computer Science and Engineering Department, Jadavpur University, Kolkata, India
[b]Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, UK

ARTICLE INFO

Keywords:
Body Area Network
Markov Decision Process
Reinforcement Learning
Transmission strategy

ABSTRACT

This paper proposes effective communication strategies for Wireless Body Area Networks (WBANs) that consist of wearable or implantable sensor nodes placed in, on/around the human body to send body vitals to a sink. The main research challenges for communication strategy formulation include limited energy resources and varying link conditions. Though energy harvested sensor nodes partially address the problem of energy efficiency, finding an optimal balance between the energy constraint of the nodes and communication reliability is still challenging. Since data loss in such networks may prove to be fatal, it is important to investigate the problem prior to deployment and come up with effective communication strategies for initiating post-deployment operations. Hence, in this paper, the nodes are stochastically modeled as a Markov Decision Process. There is a need to adapt to the changing ambient conditions through exploration and exploitation. So, a modified Q-learning technique is proposed for post-deployment decision-making by the WBAN nodes subject to the dynamic ambient conditions. The effectiveness of the proposed strategy is validated through extensive simulation and compared with state-of-the-art works. The performance of the proposed approach is also verified with a real-life dataset. The results demonstrate that around 90% successful data delivery to sink could be made with the proposed scheme in the real-life scenario.

1. Introduction

INTEGRATION of Internet of Things (IoT) in the healthcare sector initiates the era of Medicine 4.0 or Health 2.0 that marks a transition towards ubiquitous monitoring of the patients through early detection of disorders and the implementation of a proactive treatment plan [1]. Thus, wearable health monitoring systems have garnered significant attention nowadays to become an integral part of the medical IoT that aims to enhance the quality of life [2][3][4]. The advancements in sensors and wireless technology result in a revolutionary new paradigm in healthcare popularly known as Wireless Body Area Network (WBAN) that is an integral part of medical IoT [1][2]. Such a system involves small and intelligent, invasive, or non-invasive body sensor nodes to be deployed at vital positions of the human body to measure several essential health parameters. The nodes transmit collected data to a gateway device or sink acting as the network coordinator (could be a smart handheld) via which these are communicated to a remote medical server for effective analysis by professionals.

The key challenge for this network is the scarcity of resources [2]. The limited lifetime of battery powered body sensor nodes have motivated the researchers to develop various energy efficient approaches [4][5] for WBAN. Energy efficient routing approaches are also developed as in [6][7][8] including transmission power control mechanisms. In this context, energy harvesting is becoming a likable solution [9][10][11]. Generally, energy can be harvested from human body by converting the electrochemical substances into electrical power or from the movement of human body [12]. However, harvesting sufficient energy for all node operations is very difficult and often not feasible [10].

Research says that a body sensor node depletes most of its energy during transmission. Hence, designing energy-efficient transmission strategies has great significance to WBANs [13][14]. Optimal transmission power is essential to guarantee communication reliability (better signal-to-noise ratio) as well as to regulate energy drainage. However, the selection of an appropriate power level necessitates dynamic decisions depending on various ambient conditions. This could incur significant computation overhead for a resource constraint node. Hence, a few research [14][15] in literature could be found that focus on designing strategies prior to the deployment of the network. In reality, the network parameters vary dynamically, and thus, any policy found to be suitable for certain operating conditions may not remain optimal for other scenarios as well. This motivates towards updating strategies through learning the ambience. In this regard, Reinforcement Learning (RL) could be exploited that enables the WBAN devices to make autonomous decisions through exploring the environment and exploiting the knowledge gained [16].

In literature, few works could be found on WBAN that employ RL to address the problem of sensor access control [17], transmit power control [16], resource allocation [10], security against attack [18] to enhance the network performance as well as utility of the network in an optimized way. These works model their respective RL problem from the perspective of the coordinator where the task of decision making for WBAN nodes lies on the coordinator. Further, to

*Corresponding author
Email address: chandreyee.chowdhury@jadavpuruniversity.in (C. Chowdhury)
ORCID(s):

address the high dimensionality problem few works [19][20] could even be found to combine deep learning and reinforcement learning techniques to shrink the learning state space observed by the coordinator [17].

The concept of implementing the RL algorithms in coordinator lies from the fact that unlike the coordinator, the small size body sensor nodes have limited energy, computing power and storage buffer etc. Hence, incorporation of this kind of complex mathematical model could incur significant overhead for such a resource constraint node. However, when a coordinator node makes decision for the sensor nodes, additional control messages would be required to exchange node information and dissemination of the resultant strategy. Hence, this would consume some extra amount of resources in terms of energy and bandwidth. Moreover, this kind of centralized approach would mainly be suitable for the single hop star topology when all nodes are directly connected to the sink. But in practice, there may be the need for multi-hop communication particularly when the sink node does not remain within vicinity of a sensor node.

This motivated us to develop a distributed decision making scheme where each node would decide the suitable transmission strategy depending on its working conditions. Here, the entire work is carried out in two phases. In pre-deployment phase, each energy harvesting WBAN node is modeled stochastically using Markov Decision Process (MDP) so that it gets a semi-image of the WBAN they are part of. In the post-deployment phase, a modified Q-learning algorithm is designed for each WBAN node that utilizes the proposed MDP framework to adapt the transmission power based on ambient conditions. The irrelevant exploration in the Q table has been removed by modifying the state-action space. In addition, the algorithm has been initialized with a partially pre-computed policy using value iteration method to avoid random exploration at the beginning. Thus, less complex but effective transmission decisions could be made at runtime. Hence, in this paper our contributions are as follows:

- Energy harvesting body sensor nodes are stochastically modeled applying MDP prior to deployment of the network in order to find a balance between energy depletion and packet success rate.
- A modified Q-learning algorithm is designed for the post-deployment decision-making by the WBAN nodes. The algorithm is initialized with the representative pre-computed solution using the value iteration technique to enhance the convergence speed. It operates based on the pre-deployment MDP formulation. State-action space is modified to eliminate irrelevant exploration space.

The rest of the paper is organized as follows: The following section discusses some of the relevant state-of-the-art research in detail. Section 3 reviews some preliminaries of the proposed work. Next, Section 4 illustrates the design of the proposed communication strategies in detail. This is followed by Section 5 which presents the performance analysis and the discussion. Finally, Section 6 concludes.

2. Related Work

Efficient management of critical energy resources to cope with varying link conditions in Wireless Sensor Network (WSN) as well as WBAN has always been the point of concern for researchers. In [21] Fu et al. proposed a robust and energy-efficient scalefree double stage topology evolution model for WSN. A trade-off among latency, energy efficiency, and routing survivability is made by Fu et al. in [22] while making routing decisions in WSN. Further, energy exhaustion, hardware/software malfunctions, and impaired connectivity-based failure model have been designed for WSNs based on cellular automata by Fu et al. in [23]. Chen et al. in [6] proposed an energy-efficient routing protocol based on a fuzzy inference system that reduces power consumption. Another transmission power control protocol was developed by Kim et al. in [7] to address the problem of the lifetime and link reliability in WBANs based on both short and long-term link-state estimations.

Further, the authors in [14][9] address the problem of energy efficiency through designing transmission strategies so that Quality of Service (QoS) is maintained. Seyedi et al. in [9] developed decision policies of the energy harvesting WBAN nodes to determine the transmission mode to use at a given instant of time to maximize the quality of coverage. Roy et al. in [15] proposed an optimal transmission policy for WBAN using MDP subject to various input conditions such as battery level, event occurrence, packet transmission rate, and link quality. However, with this approach, only policies for representative input conditions could be found. Next, the authors in [14] optimize policies subject to the target ambience using a Genetic Algorithm (GA). This approach is worthy for designing strategies prior to the deployment of the network to find optimal solutions (i.e. policies) for a range of ambient conditions. However, decision-making at runtime requires adaptation of the pre-computed strategy through learning the environment.

In recent years, the RL technique [10][19] has been adopted extensively in the literature that facilitates learning through exploration and exploitation. Table 1 represents some of the relevant works on RL reported since the last decade. Most of these works are designed for WSN and thus can not be directly implemented on WBAN due to the inherent limitations [2]. In [24] Zheng et al. studied the sensor activation control for the optimization of green energy utilization in an energy harvesting WSN. Here, both energy generation and target distribution exhibit temporal and spatial diversities, and the temporal modes are adapted using RL. In addition, the authors in [25] proposed an energy management scheme based on RL that dynamically adapts its policy to a time-varying environment to improve the quality of service. Since, WSN nodes are less resource constraint than WBAN nodes, the works in [24][25][26][27] implement RL algorithm at node level.

Table 1
Some of the relevant research on application of RL in solving network issues in WBAN and WSN

Year	Existing work	Description	Issues handled	Network	Target application	Computation implemented	Topology considered
2012	[28]	Decentralised RL for energy-efficient scheduling in WSN	Energy efficiency, latency, interference	WSN	Sensor node	At node level	Single hop, multi hop
2014	[26]	Energy-aware task scheduling in WSNs based on Cooperative RL	Energy efficiency	WSN	Sensor node	At node level	Multi hop
2015	[24]	Sensor activation control for the optimization of green energy utilization in an energy harvesting WSN	Energy harvesting, energy balancing	WSN	Energy harvesting node	At node level	-
2016	[27]	RL-based sleep scheduling algorithm for desired area coverage in solar-powered WSN	Energy harvesting, energy balancing, lifetime	WSN	Sensor node	At node level	Multi hop
2017	[25]	Achieving energy neutrality in WSN Using RL	Energy harvesting, energy efficiency, quality of service	WSN	Sensor node	At node level	-
2018	[18]	RL-based power control for in body sensors in WBANs against jamming	Power control, reduce transmission energy, resist jamming attack	WBAN	Sensor node	At WBAN coordinator	Single hop
2018	[17]	RL-based sensor access control for WBANs	Quality of Service, energy consumption, transmission reliability	WBAN	Sensor node	At WBAN coordinator	Single hop
2019	[29]	Cooperative communications with relay selection based on deep RL in WSNs	Energy efficiency, outage probability	WSN	Sensor node	At node level	Multi hop
2020	[10]	RL-based energy efficient resource allocation for energy harvesting powered WBAN	Energy efficiency, energy harvesting	WBAN	Energy harvesting sensor node	At hub	Single hop
2020	[19]	Joint optimization of power control and time slot allocation for WBANs via deep RL	Energy efficiency, power control, quality of service	WBAN	Sensor node	At WBAN coordinator	Single hop
2020	[20]	Deep RL-based resource scheduling strategy for reliability-oriented WBANs	Resource scheduling, reliable transmission	WBAN	Sensor node	At node level	Single hop and two hop
2021	[30]	Deep RL-based on demand charging algorithm for rechargeable WSN	Lifetime, energy replenishment	WSN	Mobile charger (MC)	At base station	Multi hop but base station-MC (Single hop)

However, the authors in [10][17][18][19] focus on implementing the computationally complex RL algorithms at WBAN coordinator which contains more resource as compared to the node.An RL-based power control scheme was proposed by Chen et al. in [18] where the coordinator evaluates the optimal strategy for in-body sensors to resist jamming attacks. A resource allocation problem was investigated by Xu et al. in [10] for energy harvesting powered WBANs (EH-WBANs) where the decisions are made by a hub with partial network information. A sensor access control scheme based on reinforcement learning was proposed by Chen et al. in [17] that enables the coordinator to choose the access time and transmit power of the sensors based on the state of the sensors. These approaches are worthy particularly for star topology WBAN where the coordinator evaluates the strategy for the nodes. Besides, control packet overhead is associated with these approaches to transmit node information to the coordinator at regular intervals. Hence, in [20] Xu et al. developed a resource scheduling strategy that maximizes the reliability of the transmission of emergency-critical sensory data. Here to reduce complexity, the authors employed deep RL to solve the optimization problem at the node level.

Herewith, in this paper, we aim at analyzing the problem prior to the deployment of the network so that the nodes could be deployed with effective strategies right from the beginning. Next, we develop a less complex modified Q learning algorithm for the nodes so as to update the pre-calculated strategies through exploiting and exploring the ambience.

3. Preliminaries

3.1. Markov Decision Process

MDP is a discrete-time state transition stochastic process that provides a mathematical model for sequential decision making when outcomes are uncertain [31, 32]. An MDP is represented as a five-tuple: (S_t, A_t, P, R, γ). The notations are explained in Table 2. Here, optimization is made either through minimizing the expected cost to reach the goal or by maximizing the expected reward. Performing an action $a_t \in A_t$ in a state s_t results in a reward $r(s_t, a_t)$ and determines the state s_{t+1} (where $s_t, s_{t+1} \in S_t$) at the next decision epoch $(t+1)$ through a transition probability function. However, the decision is made based on the present state and the action performed. It does not depend on the previous states (Markov property [32]). Sequence of rewards obtained by performing a sequence of actions give the utility that exhibit simple one-step look ahead relationships. Utility value is quantified in two ways. (i) The rewards at each predicted state starting from the current state are simply added to determine the additive utility ($U_a([r_0, r_1, r_2...]) = r_0 + r_1 + r_2 + ...$). (ii) A discount factor ($\gamma < 1$) is introduced to evaluate discounted utility ($U_d([r_0, r_1, r_2...], \gamma) = r_0 + \gamma r_1 + \gamma^2 r_2 + ...$) where sooner rewards are more significant than later rewards. Hence, discounted utility is more worthy for convergence of optimization algorithms in order to get the series of actions such that the expected discounted

Table 2
Description of frequently used terms

Terms	Description
S_t	A finite set of system states s_t
A_t	A finite set of actions a_t
P	Transition probability matrix indexed in both dimensions by states. Here $p(s_{t+1} \mid s_t, a_t) = p(s_{t+1}\|s_0...s_t, a_0...a_t)$ describes the state transitions.
R	Reward matrix where the immediate reward (or expected immediate reward) gained for state transition from s_t to s_{t+1} for carrying out action a_t is evaluated following reward function $r(s_t, a_t)$.
γ	[0,1] Discount factor that gives the importance of future reward in present reward.
$\Pi(s)$	A policy Π gives an action for each state s , $\Pi^*(s)$ is optimal policy that results in maximum expected utility if followed.
ϵ'	Exploration rate

utility gets maximized.

MDP formulation is commonly solved using value iteration technique [31]. Value iteration is a dynamic programming approach [33] that works in iterative fashion. The process is repeated for all system states i.e. $\forall s \in S$. For given state transitions (as in matrix P) and corresponding rewards (as in matrix R), value iteration gives the discounted utility U_d together with the number of iterations to converge. The corresponding non stationary policies (π) could be obtained through backward induction using finite horizon method [31]. Solution of MDP can be found by applying reinforcement learning technique [17][10] as well.

3.2. Q-learning

Q-learning is an off-policy reinforcement learning algorithm that attempts to find the best action a_t for the given present state s_t [34]. The environment parameters in terms of state transition probability distribution (as in P matrix) and expected reward (as in R matrix) are given as input to the algorithm. The solution is found through (i) a technique of discovering the environment called exploration and (ii) utilizing the knowledge and experience termed as exploitation. Here, a trade-off can be made with the ϵ' greedy search. In each decision epoch, either an action a_t is chosen in a uniform random fashion among all possible actions with probability ϵ' (also known as exploration rate) or with probability $(1 - \epsilon')$ the best action learned so far is selected. A look-up table termed a Q-table is maintained here that records the maximum expected future rewards for an action at each state in terms of Q value (Quality value). $Q(s, a)$ value for each state-action pair is initialized as [0,0] and computed iteratively following Temporal Difference update rule [34][16]. The algorithm terminates when either all Q-values are obtained or a certain number of iterations is reached.

4. Design of the proposed RL-based communication strategies

4.1. System model

An energy harvesting WBAN comprising of κ sensor nodes has been taken into account in this work. Here, each node periodically sends its sensed data to a remote medical server via network coordinator or sink. Time is considered to be slotted with intervals of unit length following TDMA based schemes [2]. Since, the scheduled-based schemes like TDMA minimize energy depletion for overhearing, contention, or idle listening and thus reduce duty cycle than the contention-based MAC schemes (such as CSMA/CA) [2], we model our system with this assumption. At each time interval t, the system is described by the following parameters.

4.1.1. Energy level

The remaining energy $E_{rem}(t)$ of a node at time t can be divided into $(N+1)$ discrete levels i.e. $BL_t \in \{0, 1, 2, ..., N\}$ (following most of the WBAN transceivers). The energy level of a node is measured based on the residual energy at a time instant. Here, $BL_t = 0$ represents the situation when a node can only run its circuitry with δ_0 amount of energy. No data transmission or reception could be made at this level. However, some local computations could be performed though at this level. Energy level $BL_t = i$ for $1 \leq i \leq N$ permits reception of beacon packets from sink as well as data transmission with transmission power $tx = tx_i$ where $tx \in \{tx_1, tx_2, ..., tx_N\}$ such that $tx_1 < tx_2 < ... < tx_N$. Thus, a node must have at least $(\delta_0 + \delta_{rx} + \delta_{tx_i})$ amount of E_{rem} in order to be at $BL_t = i$. Here, δ_{rx} and δ_{tx_i} indicate the amount of energy required to receive beacon from sink and transmit data packets with transmission power $tx = tx_i$ respectively. However, the next energy level allows data transmission with $tx = tx_{i+1}$, and thus requiring $\delta_{tx_{i+1}}$ (i.e. $> \delta_{tx_i}$) amount of energy for transmission as well. Hence, at any time slot t, the remaining energy of a node $E_{rem}(t)$ occupies one of these defined levels BL_t as follows:

$$BL_t = \begin{cases} i & \text{if } (\delta_0 + \delta_{rx} + \delta_{tx_i}) \leq E_{rem}(t) < (\delta_0 + \\ & \delta_{rx} + \delta_{tx_{i+1}}) \text{ for } i \in [1, N-1] \\ N & \text{if } E_{rem}(t) \geq (\delta_0 + \delta_{rx} + \delta_{tx_N}) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

4.1.2. Event occurrence

Each node generates and/or transmits a single data packet per time slot. Data transmission between a node and the sink can be taken into account as a two-state process where the action of data transmission is subject to event occurrence. A sensing event (EO_t) describes the generation of a data packet for transmission. EO_t is 1 when there is a data packet in the queue ready to be transmitted and 0 otherwise (i.e. no event). Further, if an event is generated at any time slot t (i.e. $EO_t = 1$), the probability of generation of another event in the next time slot is given by e_{on}. Conversely, the probability

of generating no events at both present and next time slots is denoted by e_{off}.

4.1.3. Link quality

The channel conditions at each time slot t can be measured in terms of link quality $LQ_t \in \{0, 1\}$ and described as a two-state process as well. LQ_t is 1 (when Link Quality Indicator $LQI > LQI_{threshold}$) for stable channel conditions whereas $LQ_t = 0$ represents adverse scenario. At any time slot t, if LQ_t is measured as 1, it is expected to be 1 as well in the next slot $(t + 1)$ with probability lq_{on}. Conversely, the probability of $LQ_t = 0$ for both present and next time slots is given by lq_{off}.

4.1.4. Energy harvesting

At each time slot t, energy harvesting $EH_t \in \{0, 1\}$ is modeled such that EH_t is 1 if energy is harvested in present slot and EH_t would be 1 in next slot as well with probability eh_{on}. However, no energy will be harvested in the next slot $(t+1)$ with probability eh_{off} when no harvesting takes place at present slot i.e $EH_t = 0$.

4.2. Pre-deployment Phase

This phase focuses on the stochastic modeling of the nodes using MDP prior to the deployment of the network.

4.2.1. Markov Decision Process Formulations

At each time slot, the data transmission mode is regulated by MDP with the objective to maximize lifetime without degrading performance.

a) Defining system states and actions: The system state at time t can be defined as a combination of essential parameters (energy level BL_t, event occurrence EO_t, link quality LQ_t and energy harvesting EH_t) that primarily regulate a node's performance after deployment. Here, each of the parameters serves as state variable as follows:

$$s_t = (BL_t, EO_t, LQ_t, EH_t) \tag{2}$$

During each time slot t, the system performs an action $a_t \in \{0, 1, 2, ..., N\}$ that drives the system state s_t to a probable next state s_{t+1}. Here, the actions are defined in terms of data transmission. Action 0 denotes no transmission of data though some local computation could be carried out. Action 1 to N represents data transmission with corresponding transmission power $tx = tx_1$ to tx_N respectively. Each tx is associated with a packet success rate ϕ_i for $1 \leq i \leq N$. Evidently, for energy level $BL_t = i$, the permitted actions are $a_t \in \{0, 1, 2, ..., i\}$ for $0 \leq i \leq N$.

b) Reward estimation: At any given time slot t, performing an action a_t on system state s_t results in reward $r(s_t, a_t)$ and the system moves to the next state s_{t+1}. Here, the obtained reward $r(s_t, a_t)$ is quantified as follows:

$$r(s_t, a_t) = \begin{cases} \frac{p_{on}^{EO}(t) \times p_{on}^{LQ}(t) \times p_{on}^{EH}(t) \times \phi_i}{\frac{tx_{min}}{tx_i}} & \text{if } E_{rem}(t) > \delta_0 + \delta_{rx} + \delta_{tx_i} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In this case, $p_{on}^{EO}(t)$, $p_{on}^{LQ}(t)$ and $p_{on}^{EH}(t)$ denote the probability of occurring event, having stable link conditions and harvesting energy respectively at time slot t. These probabilities are evaluated as follows:

$$p_{on}^{EO}(t) = EO_{t-1} \times e_{on} + (1 - EO_{t-1}) \times (1 - e_{off}) \tag{4}$$

$$p_{on}^{LQ}(t) = LQ_{t-1} \times lq_{on} + (1 - LQ_{t-1}) \times (1 - lq_{off}) \tag{5}$$

$$p_{on}^{EH}(t) = EH_{t-1} \times eh_{on} + (1 - EH_{t-1}) \times (1 - eh_{off}) \tag{6}$$

Here, the input probabilities together determine how much favorable the environment is to carry out the action a_t. Packet success rate ϕ_i for $1 \leq i \leq N$ corresponding to the power level of the chosen action gives the benefit of selecting this particular action. In general, a high transmission power provides a better packet success since the chances of data delivery enhance with an increased power level. However, the packet success rate depends on the link quality as well. For poor link quality ($LQ_t = 0$), the following condition holds:

$$\frac{\phi_i}{(\delta_0 + \delta_{rx} + \delta_{tx_i})} \leq \frac{\phi_{i+1}}{(\delta_0 + \delta_{rx} + \delta_{tx_{i+1}})}$$

On the contrary, when the link quality is stable ($LQ_t = 1$), the following condition is satisfied:

$$\frac{\phi_i}{(\delta_0 + \delta_{rx} + \delta_{tx_i})} > \frac{\phi_{i+1}}{(\delta_0 + \delta_{rx} + \delta_{tx_{i+1}})}$$

Thus, when the link quality is good, a better packet success rate could be achieved with the same transmission power level tx_i (i.e. with similar energy consumption) as compared to its adverse counterpart. Unlike ϕ_i, the ratio $\frac{tx_{min}}{tx_i}$ represents the cost of selecting the acting transmission power tx_i (corresponding to the action a_t) among available power levels $(tx_1, tx_2, ..., tx_N)$ as compared to the minimum power level tx_{min}. Hence, this ratio in turn indicates the cost of taking the action a_t. Here, the reward function serves as the objective function that aims to find a balance between successful data delivery and energy depletion. However, no reward would be given for action 0 (i.e. no data transmission). An optimal action subject to the input conditions is found by solving MDP in order to maximize the obtained reward in each iteration.

c) Transition to next decision epoch: The system state s_{t+1} in the next time slot $(t + 1)$ is represented as follows:

$$s_{t+1} = (BL_{t+1}, EO_{t+1}, LQ_{t+1}, EH_{t+1}) \tag{7}$$

Remaining energy $E_{rem}(t+1)$ of each energy harvesting node in the next time slot $(t + 1)$ is given as follows:

$$E_{rem}(t + 1) = E_{rem}(t) - l_t + g_t \tag{8}$$

Here, l_t represents the amount of energy loss for performing an action a_t in the previous time slot t.

$$l_t = \begin{cases} \delta_0 + [\delta_{rx}]I_{true}(beacon^s) + \delta_{tx_i} & \text{w.p. } [p_{on}^{EO}(t+1)]I_i(a_t) \\ & \text{if } E_{rem}(t) \geq (\delta_0 + \\ & \delta_{rx} + \delta_{tx_i}) \\ \delta_0 + [\delta_{rx}]I_{true}(beacon^s) & \text{w.p. } I_0(a_t) \\ & \text{if } E_{rem}(t) \geq (\delta_0 + \\ & \delta_{rx}) \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

where $w.p.$ stands for $with\ probability$ and $1 \leq i \leq N$. $I_i(a_t)$ and $I_0(a_t)$ are indicator functions that become 1 when the value of a_t equals to the respective subscript (i.e. i and 0) and zero otherwise. Energy loss (l_t) in each time interval t is quantified as the sum of (i) constant energy depletion (δ_0) to run the circuitry, (ii) energy consumption (δ_{rx}) when beacon is received from sink (indicated using indicator function $I_{true}(beacon^s)$) and (iii) energy expenditure δ_{tx_i} to carry out data transmission with power level tx_i corresponding to the action a_t. Evidently, for action 0, no energy would be depleted for data transmission. Further, when a node runs out of energy, no energy loss could be reported.

Here, energy gain g_t in time slot t is measured as follows:

$$g_t = \begin{cases} \lambda & \text{w.p. } p_{on}^{EH}(t+1) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

Next, event generation EO_{t+1} in the next time slot $(t+1)$ is predicted as follows:

$$EO_{t+1} = \begin{cases} 1 & \text{w.p. } p_{on}^{EO}(t+1) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

Accordingly, link quality LQ_{t+1} in the next time slot $(t+1)$ is evaluated as

$$LQ_{t+1} = \begin{cases} 1 & \text{w.p. } p_{on}^{LQ}(t+1) \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Herewith, whether a node will harvest energy in the next time slot $(t+1)$ is predicted as

$$EH_{t+1} = \begin{cases} 1 & \text{w.p. } p_{on}^{EH}(t+1) \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

d) Formation of P matrix and R matrix: State transition matrix $[P]_{n\times n}$ (having n number of system states for each node) is constructed for each action a_t. Here, state transition probability $p^{s_t\to t+1}$ from present state s_t to probable next state s_{t+1} is quantified in terms of state transition probabilities of individual state variables as follows.

$$p^{s_t\to t+1} = p^{BL_{t\to t+1}} \times p^{EO_{t\to t+1}} \times p^{LQ_{t\to t+1}} \times p^{EH_{t\to t+1}} \tag{14}$$

where $p^{BL_{t\to t+1}}$, $p^{EO_{t\to t+1}}$, $p^{LQ_{t\to t+1}}$ and $p^{EH_{t\to t+1}}$ represent the transition probabilities between BL_t to BL_{t+1} (following Eq. 9-10), EO_t to EO_{t+1} (following Eq. 11), LQ_t to LQ_{t+1} (following Eq. 12), and EH_t to EH_{t+1} (following Eq. 13) respectively. Accordingly, the corresponding reward matrix $[R]_{n\times n}$ is formed for each action a_t (following Eq. 3-6) that records the reward gained for carrying out the action at each present state s_t.

4.2.2. Solving MDP using Value Iteration

The state value function at a state $s \in S$ for any stationary policy $\pi = (\pi^0, \pi^1, ..)$ satisfies the Bellman equation [31] as follows:

$$V^{\pi}(s) = r(s, \pi(s)) + \gamma \Sigma_y p(y|s, \pi(s)) V^{\pi}(s) \tag{15}$$

State transition matrix $[P]_{n\times n}$ and reward matrix $[R]_{n\times n}$ together with discount factor γ are given as input to a value iteration technique. Value iteration function begins with assigning an arbitrary value V_0 (generally 0.00 [31]) to each state $s \in S$. In each iteration, the value of the state $V_m(s)$ (at m^{th} iteration) is evaluated by Bellman backup. The iterations continues until ϵ convergence is achieved i.e. $max_s|V_{m+1}(s) - V_m(s)| < \epsilon$. This value iteration technique results in the optimal transmission policy π for given input conditions (i.e. e_{on}, e_{off}, lq_{on}, lq_{off}, eh_{on} and eh_{off}) together with discounted utility value U_d for each state s. The process is to be repeated each time to obtain a transmission policy for different input probability value combinations. Unlike learning techniques, the value iteration process cannot perceive the dynamic operational environment and thus, results in a fixed policy for a given input condition. Hence, in order to cope with runtime dynamic ambience, the process would require to be executed for a range of input conditions. This may incur significant computational overhead.

4.3. Post-deployment Phase

The phase focuses on the design and implementation of the proposed modified Q-learning algorithm for each WBAN node.

4.3.1. Design of the proposed modified Q-learning algorithm

In this work, we propose a modified Q-learning algorithm to solve the pre-deployment MDP formulation at runtime. The classical Q-learning algorithm is modified to enhance the convergence speed as follows:

- The action space corresponding to each state is modified.
- The state-space exploration for each state-action pair is modified.
- The Q-table is initialized following the solution obtained from the value iteration technique.

The original state-action space of the Q-learning algorithm is reduced in the proposed modified Q-learning algorithm to avoid unnecessary exploration in the Q-table. Since, the

Table 3
The relevant action space mapping table for $N = 3$ where {1***} represents the states ($s_t = \{BL_t, EO_t, LQ_t, EH_t\}$) when $BL_t = 1$

State-space	Action-space to be explored
{0***}	{0}
{1***}	{0,1}
{2***}	{0,1,2}
{3***}	{0,1,2,3}

Table 4
The relevant state space mapping table for $N = 3$ where {1***} represents the states ($s_t = \{BL_t, EO_t, LQ_t, EH_t\}$) when $BL_t = 1$

State-space	Action-space	State-space to be explored
{0***}	0	{0***}
{1***}	0	{1***}
	1	{0***} and {1***}
{2***}	0	{2***}
	1	{1***} and {2***}
	2	{1***} and {2***}
{3***}	0	{3***}
	1	{2***} and {3***}
	2	{2***} and {3***}
	3	{2***} and {3***}

permitted actions for each system state is governed by the energy level BL_t of each node, the action space A_t to be explored gets limited to {0,1,2,...,i} for the system states {i***} where $0 \leq i \leq N$. For instance, the relevant action space mapping table for $N = 3$ is presented in Table 3.

Next, execution of an action of data transmission (i.e. $a_t \neq 0$) on present state {i***}, drives the system into one of the probable states among {i***} or {$(i-1)$***} following transition probabilities recorded in $[P]_{n\times n}$ matrix where $1 \leq i \leq N$. However, since no data transmission is performed for action 0, this action when executed on states {i***} causes transition to one of the states {i***} only where $0 \leq i \leq N$. Herewith, the relevant state space S_{t+1} mapping table for $N = 3$ is presented in Table 4.

Unlike the classical Q-learning algorithm, the modified algorithm initializes the Q-table entries $Q(s, a)$ following the representative policy $\pi^{ValueIteration}$ obtained using value iteration process. The Q-value corresponding to each $\{s^i, a^j\}$ pair where action a^j (for $0 \leq j \leq N$) is recommended for state s^i (where $1 \leq i \leq n$) by $\pi^{ValueIteration}$ is initialized as $Q^0(s^i, a^j) = \psi$. Here, ψ represents a non-negative value. The rest of the Q-table entries $Q^0(s^i, a^k)$ are set to 0 such that $0 \leq k \leq N$ and $Q^0(s^i, a^j) \neq Q^0(s^i, a^k)$. This initialization process would enable the learning algorithm to make decision by exploiting some knowledge right from the beginning. Thus, data loss due to performing exploration from scratch could be prevented and faster convergence could be achieved as well.

In each episode, an action a_t for system state s_t is selected by exploration (w.p. ϵ') or exploitation (w.p. $(1-\epsilon')$). The Q-value is updated in each iteration as follows [34][16]:

$$Q^{new}(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\theta_t \tag{16}$$

where α is the learning rate and $Q^{new}(s_t, a_t)$ denotes new estimated value. θ_t is the temporal difference error at time t measured as follows [34][16]:

$$\theta_t = r(s_t, a_t) + \gamma max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \tag{17}$$

Since the value of the best next action is utilized here regardless of the policy to carry out the estimation, the algorithm is termed as off-policy. The step-by-step approach of the modified Q-learning algorithm is summarized in Algorithm 1. The algorithm comes into play with some initial knowledge about the ambience (step 4-9 of Algorithm 1) and then gradually learns the environment as well. The decision is made through exploitation and exploration (step 13-19 of Algorithm 1) accordingly. The algorithm results in the optimal transmission policy together with the Q value corresponding to each state and it ends when the outcome got stable in consecutive episodes. Hence, unlike the value iteration technique that results in a stationary policy for each given ambient condition, this algorithm is more suitable to find a solution for a dynamic environment.

4.3.2. Implementation of the proposed modified Q-learning algorithm on WBAN nodes

Herewith, the modified Q-learning algorithm (i.e. Algorithm 1) is implemented on each WBAN. To assess the input probability conditions, only a periodic beacon from the sink would suffice. Since the algorithm is initialized with the policy obtained from the value iteration technique, the algorithm can even make effective decisions for a node that recently joins a network and does not have any prior knowledge of the input conditions. Gradually, as the node learns the input probabilities, these probabilities are fed into the proposed algorithm to find the optimal policy through exploration and exploitation. However, with this approach, the node only needs to execute the algorithm again when there is a change in the input probabilities. Accordingly, the node can update its policy subject to ambient conditions after deployment.

4.3.3. Complexity analysis of the proposed modified Q-learning algorithm

The proposed modified Q-learning algorithm finds a solution based on two main search spaces i.e. action space and state space, both of which are well-bounded here. To find the best action (a_t) for each current state (s_t), the algorithm explores any action with probability ϵ' from the action space with uniform random distribution as mentioned in Step 13 of Algorithm 1 which gives result in constant time. However, when the decision is made through exploitation with probability $(1-\epsilon')$, the algorithm needs to search the Q-values in its Q-table to find the action with the highest Q-value. Here,

the complexity may polynomially increase with the action as in Step 14 of Algorithm 1. Thus, the search space never grows exponentially.

In [35], Koenig et al. established that the task of reaching a goal state for the first time is $\mathcal{O}(total\ no.\ of\ actions \times total\ no.\ of\ states)$ or, alternatively $\mathcal{O}(action\ space \times total\ state\ space)$ through appropriate task representation or suitable initial Q-values. Here, the action space or the maximum number of actions (m) for a state depends on the number of transmission power levels in use which is less than the total number of states n following the system model. Hence, the action space would vary with $\mathcal{O}(m)$. In addition, in this work, the goal state represents such a state when the entire energy of a node would be drained and nothing would be left for performing data transmission. The proposed system comprises of the eight-goal states which can be represented as $\{0\ ***\}$. Following the design of our proposed modified Q learning algorithm as illustrated in Section 4.3.1 and Table 4, the effective state space for each state would be in the order of the total number of goal states i.e. $2 \times no.\ of\ goal\ states$ which is constant in this case. Hence, in the worst case, for n number of system states, any of the goal states could be reached through $((n-8)-1)$ i.e. $(n-7)$ intermediate states which can be considered as $\mathcal{O}(n)$. Herewith, the worst-case complexity to reach the goal state is $\mathcal{O}(mn)$ which is always less than $\mathcal{O}(n^2)$.

This is in-line with the results established in [35] that every Q-learning algorithm that does not know the effect of an action before it has executed it at least once has the worst case complexity of $\mathcal{O}(n^3)$. It is also reported by the authors in [35] that the complexity can be reduced to $\mathcal{O}(n^2)$ if the q-values are suitably initialized. In Algorithm 1, this is done based on the value iteration outcomes of the pre-deployment phase.

Algorithm 1: Modified Q-learning

input :
$[P]_{n\times n}$, $[R]_{n\times n}$, $\pi^{ValueIteration}$
output:
Optimal policy $\pi^*(s) : max_a Q^*(s,a)$
$episode = 0$;
Set action space $A_t = \{0, 1, 2, ...,i\}$ for states $\{i***\}$: $0 \le i \le N$;
if $(a_t = 0)$ **then**
  Set state space $S_{t+1} = \{i\ ***\} : 0 \le i \le N$;
else
  Set state space $S_{t+1} = \{\{i\ ***\}, \{(i-1)\ ***\}\}$: $0 \le i \le N$;
Initialize each $Q(s,a)$ in $[Q]_{n\times(N+1)}$ with representative $\pi^{ValueIteration}$;
Observe s_t;
Initialize $(\alpha, \gamma, \epsilon')$;
repeat
  Select a_t uniformly from A_t w.p. ϵ';
  Otherwise select $a_t \in A_t$: $Q(s_t, a_t) == [Q(s_t, a)]_{max} \forall a \in A_t$ w.p. $(1-\epsilon')$;
  Execute a_t;
  Obtain $r(s_t, a_t)$ from $[R]_{n\times n}$;
  Move to $s_{t+1} \in S_{t+1}$ following $[P]_{n\times n}$;
  Update $Q(s,a)$ in $[Q]_{n\times(N+1)}$ following equation 16-17;
  Set $s_t = s_{t+1}$;
  $episode += 1$;
until *episode* M;

5. Performance analysis

This section presents the experimental results demonstrating the performance of the proposed approach. The experiments are mostly conducted in pre-deployment phase using Python[1] with the support of 'pymdptoolbox' library providing classes and functions for the resolution of discrete time MDP. Post-deployment performance analysis is conducted using Castalia 3.2[2] open source network simulator based on OMNeT++ that is widely used for WBAN experimentation.

5.1. Pre-deployment experimentation

In this phase, several experiments are carried out to find out the best strategy based on the ambient conditions.

5.1.1. Experimental setup

Four energy levels $BL_t \in \{0, 1, 2, 3\}$ are taken into account for experiments. Event occurrence $EO_t \in \{0, 1\}$, link quality $LQ_t \in \{0, 1\}$ and energy harvesting $EH_t \in \{0, 1\}$ are implemented with conditional probabilities $\{e_{on}, e_{off}\}$, $\{lq_{on}, lq_{off}\}$ and $\{eh_{on}, eh_{off}\}$ respectively within their predefined range. Accordingly, 32 ($4\times2\times2\times2$) different states are defined following Eq. 2. Four actions $a_t \in \{0, 1, 2, 3\}$ are considered for experiments where action 0 denotes no transmission. The rest of the actions (i.e. 1, 2 and 3) corresponds to data transmission with three gradually increasing power levels tx_1, tx_2 and tx_3. However, this approach can support more number of actions as well. Packet success rate (ϕ_i) corresponding to the power level tx_1, tx_2 and tx_3 are taken into account as {0.7,0.8,0.9} respectively for stable link quality and {0.3,0.4,0.5} respectively for adverse conditions. The state transition matrix $[P]_{32\times32}$ for each action a_t is constructed considering the input probabilities $\{e_{on}, e_{off}\}$, $\{lq_{on}, lq_{off}\}$ and $\{eh_{on}, eh_{off}\}$ following Eq. 14. The corresponding reward matrix $[R]_{32\times32}$ is formulated following Eq. 3-6.

5.1.2. Results of value iteration

In the following experiments, the performance of the proposed MDP-based model is observed when the value iteration process is applied. An insight about how reward varies corresponding to each action performed at any given state following the reward function $r(s_t, a_t)$ (i.e. objective func-

[1] https://www.anaconda.com/products/individual last accessed date: 28.08.2020.
[2] https://github.com/boulis/Castalia/ last accessed date: 10.07.2020.

tion) is presented in Fig. 1. The results are plotted for states {3111}, {3101}, {2011} and {2001}. Here, two pairs of states are chosen for observation. One pair (i.e. {3111} and {3101}) having maximum energy level (i.e. $BL_t = 3$) and the other (i.e. {2011} and {2001}) with moderate energy level (i.e. $BL_t = 2$). Here, link quality varies in each pair of states. Since energy level, $BL_t = 3$ permits all four actions, rewards are obtained for performing actions 1, 2, and 3 in states {3111} and {3101}. However, more rewards could be achieved for taking the same action at a state that gives more potential for data transmission. Further, no reward could be obtained as well for carrying out any prohibited action at a given state (as in the case of action 3 in states {2011} and {2001}).

However, two different trends of gaining reward corresponding to actions could be observed subject to the link quality. For the states where link quality is stable like {3111} and {2011}, more reward is obtained for performing an action that suggests data transmission with lower transmission power. Accordingly, the reward decreases as an action of high power transmission is carried out. Unlike this, the reverse trend is found in the case of adverse link conditions (i.e. for states {3101} and {2001}). In this case, the action of high power transmission secures more rewards. Hence, for adverse link conditions, the objective is to increase the power level gradually so as to cope with the environment. However, for stable conditions, the focus is made on energy saving.

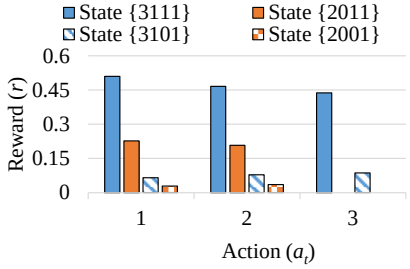

Figure 1: Rewards obtained corresponding to each action a_t at different states s_t where state {3111} represents $BL_t = 3$, $EO_t = 1$, $LQ_t = 1$, and $EH_t = 1$.

In the next experiment, the variation of utility values for different input probability combinations (e_{on}, e_{off}, lq_{on}, lq_{off}, eh_{on}, eh_{off}) are studied as shown in Fig. 2. Here, input probability combinations are chosen such that these would cover the entire horizon. The outcome for state {3111} (representing a favorable state for data transmission) is plotted here. The input probability combinations (0.9, 0.6, 0.9, 0.6, 0.9, 0.6) and (0.6, 0.9, 0.6, 0.9, 0.6, 0.9) producing the highest and the lowest utilities respectively can be regarded as the best case and the worst case scenarios. Further, it is found that, the variations of e_{on}, lq_{on}, and eh_{on} (observing outcomes for (0.9, 0.6, 0.9, 0.6, 0.9, 0.6) and (0.6, 0.6, 0.6, 0.6, 0.6, 0.6)) make more impact than the variations of e_{off}, lq_{off}, and eh_{off} (observing outcomes for (0.6, 0.6, 0.6, 0.6, 0.6, 0.6) and (0.6, 0.9, 0.6, 0.9, 0.6, 0.9)). However, an average utility obtained for probability combination (0.8, 0.6, 0.8, 0.6, 0.8, 0.6) can be taken into account as an average scenario.

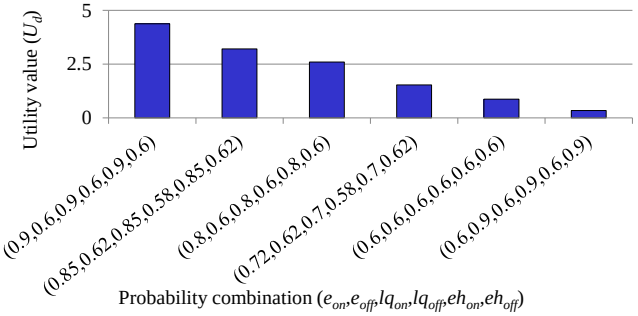

Figure 2: Utility values for different probability combinations.

Next, a vision is given in Fig. 3 about how the utility value U_d changes corresponding to each system state s_t. Here, observation is made for three representative input probability combinations denoting best, average, and worst scenarios (as mentioned in the previous experiment) such that an overview could be obtained for the entire horizon. Utility values are obtained for each state based on how favorable the state is for data transmission. Since no data transmission could be performed at states {0***}, these states receive no utility value. The highest utility is gained for the state {3111} and thus, can be regarded as the most favorable state. Evidently, more utility is obtained for the states with higher energy levels since more data packets could be sent with more energy. Besides, the states {***1} denoting the occasion of energy harvesting enhance benefits in the long run as well. Thus, the states {***1} receive more utility as compared to the states regulating event occurrence (i.e. {*1**}) or stable link quality (i.e. {**1*}) irrespective of the energy level. Moreover, a similar trend could be found for the three representative input conditions (presenting two extremes and an average situation) which indicate the proposed approach could effectively work for any input condition defined by the probability combinations. Herewith, the policy obtained for the average scenario could serve as the reference solution for the modified Q-learning technique.

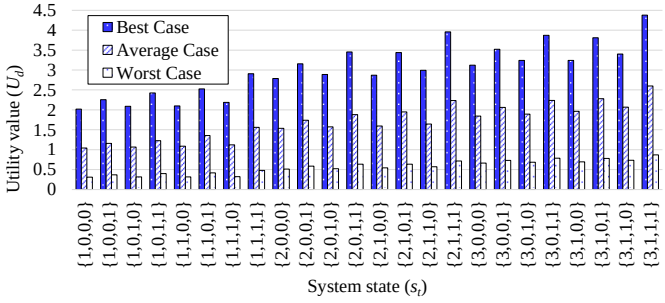

Figure 3: Utility values U_d corresponding to each state s_t where state {2010} denotes $BL_t = 2$, $EO_t = 0$, $LQ_t = 1$, and $EH_t = 0$.

5.1.3. *Performance of the modified Q-learning at the pre-deployment phase*

In the following experiments, we analyze the performance of the proposed modified Q-learning algorithm in the MDP framework when executed at the pre-deployment phase. In Fig. 4, the impact of individual input probability (each of $e_{on}, e_{off}, lq_{on}, lq_{off}, eh_{on}, eh_{off}$) on the performance of the modified Q-learning algorithm has been investigated. Results are recorded for state {3111}. Here, the Q value obtained for a given input probability combination (i.e. 0.9, 0.6, 0.9, 0.6, 0.9, 0.6) has been taken into account as reference. Next, each of the input probabilities are varied one by one keeping others unchanged and the corresponding Q values are plotted. It is found that, the variations in e_{on}, lq_{on} and eh_{on} results in more changes in Q-values as compared to the effect due to the variations in e_{off}, lq_{off} and eh_{off} (supporting Fig. 2). However, the variation in lq_{on} (by observing outcome for (0.9, 0.6, 0.9, 0.6, 0.9, 0.6) and (0.9, 0.6, 0.8, 0.6, 0.9, 0.6)) mostly influences the obtained Q-value as compared to others. Hence, this probability leaves a significant impact on the resultant policy as well.

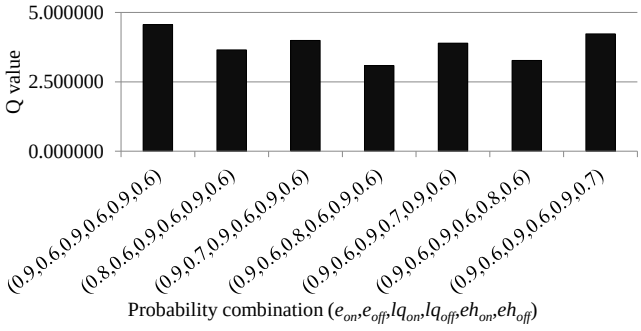

Figure 4: Q values for different probability combinations.

The effect of the number of episodes on the outcome of the proposed algorithm is illustrated in Fig. 5. Results are plotted for given input conditions (both favorable and unfavorable). The proposed modified Q-learning algorithm works in iterations or episodes. In each episode, it learns the environment and updates its Q table. Hence, the Q value increases for all states irrespective of input conditions since it gains more knowledge about the environment with more iterations. For clarity, observations are reported only for a state {3111}.

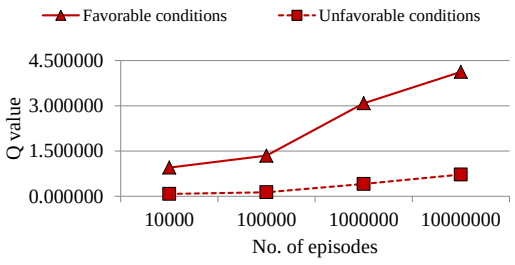

Figure 5: Q values with increasing number of episodes.

Next, we observe how the policy for a given probability combination changes each time the proposed modified Q-learning algorithm is executed. The proposed algorithm starts working on a given probability combination and in each iteration, it makes a decision either by exploring the ambience or exploiting its knowledge. Hence, it tunes the resulting policy accordingly as it learns the dynamic environment. This causes variations in the obtained outcome in each execution. Table 5 summarizes the policies obtained in 5 runs. Here, a policy gives the best action corresponding to each system state s_t in the order as presented in Table 5. Though the policies apparently vary from each other, few trends could be found by closely observing these. Action 2 or 3 i.e. the action of data transmission with higher transmission power is mostly suggested when remaining energy is at maximum level (i.e. $BL_t = 3$). Further, action 3 is mainly recommended for states {3*0*} i.e. when the energy level is high but link quality is poor. Action 1 is found to be the mostly recommended action irrespective of energy level since it is the most energy-efficient action. In addition, it is observed that often for the states having low energy levels and poor link quality i.e. {1*0*}, action 0 is suggested. This is because here the remaining energy could get insufficient to make high power transmission to overcome the adverse environmental conditions. In such a situation, no transmission and waiting for favorable conditions could become beneficial instead of data transmission with very few chances to reach the destination. In addition, no transmission (i.e. action 0) is often recommended when there is no event in the present slot (for states {*0**}).

In the following experiment, we have investigated the performance of the modified Q-learning algorithm with respect to the conventional approaches i.e. classical Q-learning algorithm, greedy approach, and random power selection approach. The Q-value (recorded after 10000000 episodes) corresponding to each state for individual technique considering given input conditions has been plotted in Fig. 6. Unlike the modified Q-learning, in classical Q-learning, there is no initial knowledge about the ambience and the algorithm starts learning the environment only through exploration. Further, there exists some unnecessary exploration of state-action space in the Q table. Hence, the Q values corresponding to each state are found to be low as compared to the proposed approach. Besides, the greedy approach only chooses the action of data transmission with maximum allowable transmission power depending on the energy level in each episode whereas, in the random power selection approach actions are selected randomly. Since both approaches do not optimize their decisions depending on the environmental conditions, the proposed algorithm is found to outperform them as well.

The pre-deployment phase ends here. The analysis of the performance of the proposed modified Q-learning algorithm in the pre-deployment phase gives an estimate of its operation and behavior in the post-deployment phase. However, the representative policy obtained using the value iteration technique in this phase is passed as input to the modified Q-

Table 5
Policy obtained for a given probability combination at different execution where each '0', '1', '2' or '3' in a policy '000...322' (from left to right) represents the best action corresponding to the state {0000} to {3111} respectively

Run number	Obtained policy (Recommended action a_t for each state s_t in the following order)																														
	{0000}	{0001}	{0010}	{0011}	{0100}	{0101}	{0110}	{0111}	{1000}	{1001}	{1010}	{1011}	{1100}	{1101}	{1110}	{1111}	{2000}	{2001}	{2010}	{2011}	{2100}	{2101}	{2110}	{2111}	{3000}	{3001}	{3010}	{3011}	{3100}	{3101}	{3110}	{3111}
1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	2	1	1	1	2	1	1	1	2	1	2	2	2	2	2
2	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	1	2	0	1	2	1	1	1	1	2	0	1	2	3	2	2
3	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	1	1	2	1	1	2	1	1	1	1	2	0	1	3	2	1	1
4	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	2	1	1	1	2	2	2	1	2	1	2	3	2	1	1
5	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	2	0	1	2	2	2	1	1	2	1	1	2	3	1	2

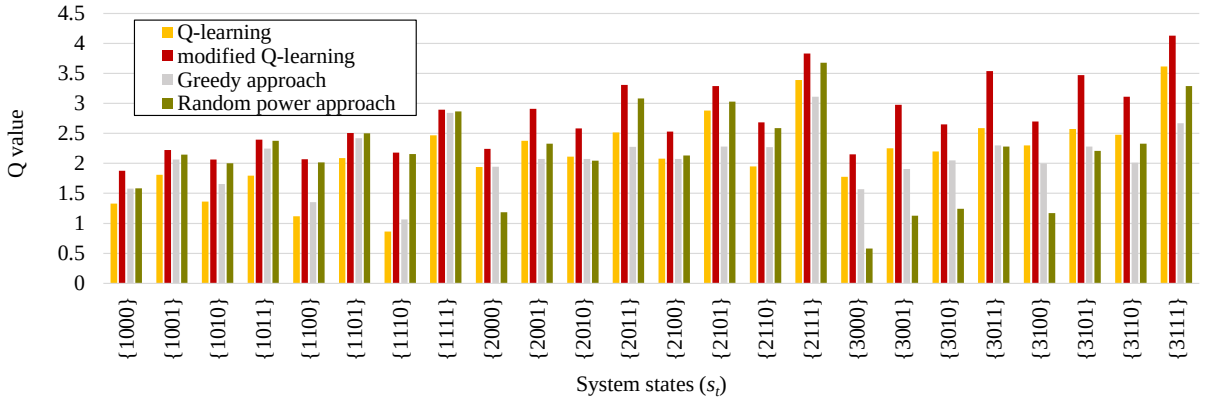

Figure 6: Q value corresponding to each system state for a given input condition.

learning algorithm implemented in the next phase.

5.2. Post deployment analysis

This section analyzes the performance of the proposed scheme from various aspects after deployment. A WBAN is simulated in Castalia-3.2[2]where the proposed modified Q-learning algorithm is implemented on each node for execution at runtime. Seven energy harvesting nodes are placed all over the human body (around 1m × 1.9m area) together with a sink residing at the waist to measure the body vitals. Experiments are carried out by taking into account three transmission power levels -15dBm, -12dBm, and -10dBm (supported by most of the WBAN transceivers) corresponding to action 1, 2, and 3 respectively. The transmission range corresponding to each power level is regulated with BANRadio[2]. Simulation has been made for 10000 sec following ZigBeeMAC (IEEE 802.15.4 standard) protocol. Energy depletion of each node as well as the network is measured using the Resource Manager module of Castalia[2] which keeps track of the energy spent by the node. This module takes into account the radio model in place and the initial energy of the individual nodes. It also holds some node-specific quantities such as the clock drift and the baseline power consumption. It has a complete view of the total power drawn depending on which the energy consumption is calculated.

5.2.1. Comparative analysis of the proposed approach

In the following experiment, a comparative study is carried out among the performance of the proposed modified Q-learning algorithm both when executed before deployment (only the obtained policies are incorporated into the nodes) as well as at runtime with respect to the state-of-the-art approaches EEFR [6] and LSE-TPC [7] in similar a simulation setup. Results are depicted in Fig. 7. Though in both cases (i.e. pre-deployment and post-deployment) the proposed modified Q-learning algorithm starts with the same initial knowledge (gained from the results of the value iteration technique), learning the environment through the assessment of input probability conditions becomes more accurate at runtime. This enables to make more appropriate transmission decisions at runtime which is reflected in the performance.

Unlike the proposed approach, transmission power regulation takes place based only on distance as part of routing decisions in EEFR [6]. Hence, the effect of environmental conditions remains unaddressed in the decision-making. Though in LSE-TPC [7], transmission power is tuned according to the link state, the other significant issues such as the impact of sensing rate, remaining energy are not taken into account here. Hence, the proposed scheme is found to exhibit more successful data delivery to the sink as compared to the existing approaches.

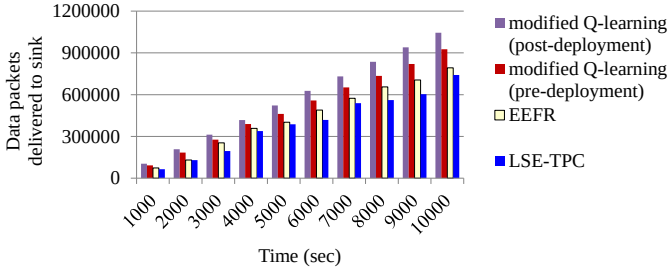

Figure 7: Variation of successful data delivery to sink with respect to time.

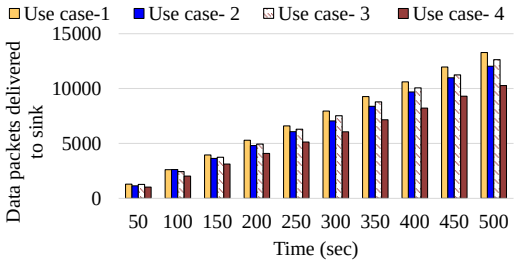

Figure 8: Performance of the proposed strategies when Use case- 1: only the reference node transmits, Use case- 2: the nearby nodes of the reference node also transmit with the proposed strategies, Use case- 3: the nearby nodes of the reference node transmit with the least cost policy, and Use case- 4: the nearby nodes of the reference node transmit with the greedy approach.

5.2.2. *Effect of multi-node transmissions on the proposed strategies*

In the next experiment, we observe the impact of the transmissions made by other nodes in the proximity on the performance of a node that follows the proposed strategy. For this experiment, we have chosen a node in the network as a reference node that is in the close communication range of the other nodes. Here, we have designed four use cases for analysis and the observations are plotted in Fig. 8. The first use case i.e. Use case- 1 represents the situation when only the reference node transmits and the other nodes in the proximity make no transmission. The next use case i.e. Use case- 2 depicts the scenario when the nearby nodes make transmission decisions with the proposed strategies. Use case- 3 and Use case- 4 describe the situations when the nodes in the proximity of the reference node only transmit with the least cost policy (use minimum power level) and the greedy approach (use maximum power level) respectively.

It is evident from the outcome that with the proposed strategy performance increases when no other nodes in the proximity perform transmission (Use case- 1), or the nodes in the proximity follow the least cost policy i.e. transmit with minimum power level (Use case- 3). This is because the channel conditions remain stable and favorable for transmission in these scenarios. On the contrary, the channel conditions become adverse when the nearby nodes follow the greedy approach i.e. transmit with the highest power level (Use case- 4). However, a consistent and balanced performance can be achieved when the nearby nodes follow the proposed strategy (Use case- 2).

5.2.3. *Performance of the proposed strategies based on real WBAN dataset*

The next experiment evaluates the effectiveness of the proposed strategies based on the real WBAN dataset[3]. Here, we experimented with a scenario where a sensor node incorporated with the proposed modified Q-learning algorithm is placed at the right foot and the sink is located at the waist. We have utilized the RSSI values[3] of a total of 30 minutes as perceived by the sensor node from the sink to analyze the variations in the channel conditions. The obtained RSSI values[3] range from -42dBm to -94dBm during this period while receiving periodic beacons from the sink at every 20 ms. Here, a value of -70dBm is considered as the threshold. This is because below this RSSI value the evaluated packet success rate ϕ given by [36] $e^{k_{data}log(1-BER)}$ for this environment becomes in the range of 0 to 0.5 satisfying the condition for poor link quality. Here k_{data} is the size of a packet in bits, and the BER corresponding to a node for the given RSSI values are calculated as in [37]. The RSSI values less than the threshold indicates the adverse channel conditions and vice versa. A window of 120 sec (depending on the rate of RSSI variation) has been considered to estimate the conditional probabilities related to the channel conditions. The modified Q-learning algorithm implemented in the node updates its policy in every 120 sec with the variations in input.

Performance of the node is plotted in terms of Packet Delivery Ratio (PDR) with the variations of lq_{on} and lq_{off} as presented in Fig. 9. PDR is defined as the ratio between total data packets received by the sink to the total data packets sent by the sensor nodes $\forall i \in \kappa$.

$$PDR = \frac{data^{s}_{rec}}{\sum_{i=1}^{\kappa} data^{i}_{sent}} \tag{18}$$

It is evident from the outcome that a reasonable PDR i.e. above 0.9 (i.e. 90%) is achieved with time which follows moderately similar pattern of the variations of lq_{on}. This findings support the observations of Fig. 4 that the variations of lq_{on} mostly influences the obtained Q-values and the resultant policies. Hence, this result also validates our proposed formulations.

6. Conclusion

The objective of this paper was to propose communication strategies for energy harvesting WBAN nodes to address the problem of finding an optimal balance between communication reliability and energy depletion subject to a dynamically changing environment. Accordingly, in this paper, RL is used to make decisions at the node level through exploring and exploiting the ambience. A modified Q-learning algorithm is proposed to solve the pre-deployment MDP formu-

[3]https://www.kaggle.com/guanslong/wban-rssi-dataset Accessed: 10.04.2021

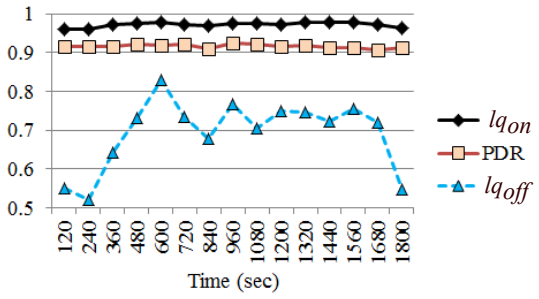

Figure 9: Variation of PDR with the variation of lq_{on} and lq_{off} obtained from real dataset.

lation at runtime so that a node can adapt its transmission decisions through learning the ambience. The effectiveness of the proposed modified Q-learning algorithm is also analyzed when employed in the pre-deployment phase. However, it is found that, with the proposed approach, the performance increases as compared to the pre-deployment strategy adaptation. The effect of multi-node transmission on the strategy has been explored. The performance of the proposed scheme is also validated based on the real WBAN dataset. For future work, we seek to implement the proposed approach on a real health monitoring platform.

References

[1] Y. A. Qadri, A. Nauman, Y. B. Zikria, A. V. Vasilakos, S. W. Kim, The future of healthcare internet of things: a survey of emerging technologies, IEEE Communications Surveys & Tutorials 22 (2) (2020) 1121–1167.

[2] S. Movassaghi, M. Abolhasan, J. Lipman, D. Smith, A. Jamalipour, Wireless body area networks: A survey, IEEE Communications Surveys & Tutorials 16 (3) (2014) 1658–1686.

[3] R. Gravina, G. Fortino, Wearable body sensor networks: State-of-the-art and research directions, IEEE Sensors Journal 21 (11) (2020) 12511–12522.

[4] M. Nekoui, L. Chu, A. Eslami, Ssiot: Energy-efficient optimal admission control for body area networks, Transactions on Green Communications and Networking (2020).

[5] M. Salayma, A. Al-Dubai, I. Romdhani, Y. Nasser, Reliability and energy efficiency enhancement for emergency-aware wireless body area networks (wbans), IEEE Transactions on Green Communications and Networking 2 (3) (2018) 804–816.

[6] Y. M. Chen, Y. PENG, Energy efficient fuzzy routing protocol in wireless body area networks, International Journal of Engineering 4 (1) (2013) 59–63.

[7] S. Kim, D.-S. Eom, Link-state-estimation-based transmission power control in wireless body area networks, IEEE journal of Biomedical and Health Informatics 18 (4) (2013) 1294–1302.

[8] N. Kaur, S. Singh, Optimized cost effective and energy efficient routing protocol for wireless body area networks, Ad Hoc Networks 61 (2017) 65–84. doi:https://doi.org/10.1016/j.adhoc.2017.03.008.

[9] A. Seyedi, B. Sikdar, Energy efficient transmission strategies for body sensor networks with energy harvesting, IEEE Transactions on Communications 58 (7) (2010) 2116–2126.

[10] Y.-H. Xu, J.-W. Xie, Y.-G. Zhang, M. Hua, W. Zhou, Reinforcement learning (rl)-based energy efficient resource allocation for energy harvesting-powered wireless body area network, Sensors 20 (1) (2020) 44.

[11] R. Zhang, J. Yu, Y. Guan, J. Liu, L. Tang, A dominating set-based sleep scheduling in energy harvesting wbans, IEEE Transactions on Vehicular Technology (2021).

[12] S. Basagni, M. Y. Naderi, C. Petrioli, D. Spenza, M. Conti, S. Giordano, I. Stojmenovic, Wireless sensor networks with energy harvesting., Mobile ad hoc networking 1 (2013) 701–736.

[13] W. Badreddine, C. Chaudet, F. Petruzzi, M. Potop-Butucaru, Broadcast strategies and performance evaluation of ieee 802.15.4 in wireless body area networks wban, Ad Hoc Networks 97 (2020) 102006. doi:https://doi.org/10.1016/j.adhoc.2019.102006.

[14] M. Roy, C. Chowdhury, N. Aslam, Designing ga based effective transmission strategies for intra-wban communication, Biomedical Signal Processing and Control 70 (2021) 102944.

[15] M. Roy, C. Chowdhury, N. Aslam, Designing transmission strategies for enhancing communications in medical iot using markov decision process, Sensors 18 (12) (2018) 4450.

[16] R. Kazemi, R. Vesilo, E. Dutkiewicz, R. Liu, Dynamic power control in wireless body area networks using reinforcement learning with approximation, in: 2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications, IEEE, 2011, pp. 2203–2208.

[17] G. Chen, Y. Zhan, G. Sheng, L. Xiao, Y. Wang, Reinforcement learning-based sensor access control for wbans, IEEE Access 7 (2018) 8483–8494.

[18] G. Chen, Y. Zhan, Y. Chen, L. Xiao, Y. Wang, N. An, Reinforcement learning based power control for in-body sensors in wbans against jamming, IEEE Access 6 (2018) 37403–37412.

[19] L. Wang, G. Zhang, J. Li, G. Lin, Joint optimization of power control and time slot allocation for wireless body area networks via deep reinforcement learning, Wireless Networks (2020) 1–10.

[20] Y.-H. Xu, G. Yu, Y.-T. Yong, Deep reinforcement learning-based resource scheduling strategy for reliability-oriented wireless body area networks, IEEE Sensors Letters (2020).

[21] X. Fu, P. Pace, G. Aloi, W. Li, G. Fortino, Toward robust and energy-efficient clustering wireless sensor networks: A double-stage scale-free topology evolution model, Computer Networks 200 (2021) 108521.

[22] X. Fu, G. Fortino, P. Pace, G. Aloi, W. Li, Environment-fusion multipath routing protocol for wireless sensor networks, Information Fusion 53 (2020) 4–19.

[23] X. Fu, Y. Yang, Analysis on invulnerability of wireless sensor networks based on cellular automata, Reliability Engineering & System Safety 212 (2021) 107616.

[24] J. Zheng, Y. Cai, X. Shen, Z. Zheng, W. Yang, Green energy optimization in energy harvesting wireless sensor networks, IEEE Communications Magazine 53 (11) (2015) 150–157.

[25] F. A. Aoudia, M. Gautier, O. Berder, Learning to survive: Achieving energy neutrality in wireless sensor networks using reinforcement learning, in: 2017 IEEE International Conference on Communications (ICC), IEEE, 2017, pp. 1–6.

[26] M. I. Khan, B. Rinner, Energy-aware task scheduling in wireless sensor networks based on cooperative reinforcement learning, in: 2014 IEEE International Conference on Communications Workshops (ICC), IEEE, 2014, pp. 871–877.

[27] H. Chen, X. Li, F. Zhao, A reinforcement learning-based sleep scheduling algorithm for desired area coverage in solar-powered wireless sensor networks, IEEE Sensors Journal 16 (8) (2016) 2763–2774.

[28] M. Mihaylov, Y.-A. Le Borgne, K. Tuyls, A. Nowé, Decentralised reinforcement learning for energy-efficient scheduling in wireless sensor networks, International Journal of Communication Networks and Distributed Systems 9 (3-4) (2012) 207–224.

[29] Y. Su, X. Lu, Y. Zhao, L. Huang, X. Du, Cooperative communications with relay selection based on deep reinforcement learning in wireless sensor networks, IEEE Sensors Journal 19 (20) (2019) 9561–9569.

[30] X. Cao, W. Xu, X. Liu, J. Peng, T. Liu, A deep reinforcement learning-based on-demand charging algorithm for wireless rechargeable sensor networks, Ad Hoc Networks 110 (2021) 102278.

[31] M. L. Puterman, Markov decision processes: discrete stochastic dynamic programming, John Wiley & Sons, 2014.

[32] L. Kleinrock, Queueing systems. volume i: theory (1975).
[33] R. A. Howard, Dynamic programming and markov processes. (1960).
[34] R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.
[35] S. Koenig, R. G. Simmons, Complexity analysis of real-time reinforcement learning, in: Proceedings of the Eleventh National Conference on Artificial Intelligence, AAAI'93, AAAI Press, 1993, pp. 99—-105.
[36] M. Roy, C. Chowdhury, S. Bhattacharyya, N. Aslam, Finding optimal transmission strategy for intra-wban communications, Electronics Letters 56 (23) (2020) 1283–1286.
[37] J. Y. Khan, M. R. Yuce, G. Bulger, B. Harding, Wireless body area network (wban) design techniques and performance evaluation, Journal of medical systems 36 (3) (2012) 1441–1457.