Optimal Health Monitoring via Wireless Body Area Networks

Yair Bar David, Tal Geller, Evgeni Khmelnitsky and Irad Ben-Gal

Department of Industrial Engineering
Tel Aviv University, Ramat-Aviv, Tel-Aviv, 69978, Israel
bengal@tau.ac.il

Andrew Ward, Daniel Miller and Nicholas Bambos

Department of Electrical Engineering
Stanford University, Stanford, CA 94305, USA
bambos@stanford.edu

***Abstract* - We consider the use of a wireless body area network for remote patient health monitoring applications. Our proposed network consists of a controller and multiple sensors, whose signals provide information on the health state of a patient. We model this patient-sensor network as a partially observable Markov decision process. The sensor outputs are used by the controller to update the patient's health-state belief probabilities and select a subset of sensors to be activated at the next decision epoch. We propose two operational algorithms that allow accurate monitoring of a patient's health state while minimizing operational and misclassification costs: i) a greedy algorithm, which applies a one-step look-ahead approach, and ii) a dynamic programming-based algorithm which yields the optimal policy. We provide a numerical example which demonstrates the applicability of the suggested methods and provides insights.**

***Keywords* — Wireless body area networks, optimal control, controlled sensing, partially observable Markov decision processes (POMDP), dynamic sensor selection, dynamic programming**

I. INTRODUCTION

In this paper, we explore the task of real-time patient health status monitoring using a wireless body area network (WBAN). Due to the nature of such wireless networks, the system must be energy-efficient in order to achieve long term user compliance and real-world applicability. Additionally, the system's prediction concerning the patient's health state must be sufficiently accurate.

We present a control scheme, which aims at minimizing the total cost of the system's energy consumption and the penalty associated with misclassification of the patient's health state (we assume the true patient's health state is unknown to the controller). We first model the sensor system and controller, as well as the underlying patient health state, as a partially observable Markov decision process (POMDP). We then discuss two possible sensor selection policies by which the controller selects the activated sensors throughout the systems operation. We assume a WBAN system with N heterogeneous sensors. At each decision epoch, the controller selects a subset of sensors to be activated, observes the activated sensors' outcomes, and updates an internal belief state. This belief state, modeled as a probability distribution, represents the controller's confidence that a patient is in a certain health state. When updating the belief state, the controller considers the sensor outcomes from the current epoch, the previous belief state, and the known true health state dynamics. With this updated belief state, the controller activates a new subset of sensors, enabling the continuation of the closed-loop patient monitoring system.

Recent technological advances have enabled the incorporation of WBANs in real-life health monitoring systems. WBANs typically consist of an array of sensors designed for real-time monitoring of different physiological metrics [1], [2]. Extensive research has been conducted concerning the applicability and requirements for successful implementation of such systems [3]. Amongst other limitations, energy consumption is a main restraining factor of WBAN systems [1], [4], specifically when considering smartphones as a primary communication channel. Research in the field of WBANs has primarily focused on the reduction of energy consumption via innovations in sensor design, communication protocols, signal processing, and other technologies [5]. However, to this day, energy consumption remains one of the main limitations of WBAN systems. We approach this challenge by proposing an efficient patient health monitoring scheme. The approach presented here does not rely on conventional WBAN mathematical modeling techniques, which are usually based on simplifications allowing only homogeneous or identical sensors and "perfect sensing information" [1], [6], [7], [8].

When modeling WBANs, POMDPs have been used to describe the transitions of the system's information states [9]. A POMDP model is defined by a set of states, actions, conditional transitions between the states, a cost function and a set of observations [10]. In this paper, we adopt the POMDP approach and extend the model by allowing non-perfect sensing information obtained from multiple activated sensors at each time step. In addition, misclassification and sensor activation costs are taken into account in order to balance the associated uncertainties and costs in the system. We then discuss and compare two control algorithms: a computationally efficient look-ahead greedy algorithm, which returns a policy that minimizes the cost in the next time step, and a dynamic programming-based algorithm, which results in the optimal policy. The optimal policy is determined by applying value iteration on a discretization of the health state probability space. A numerical examination is presented to assess the policies' performance.

The model described in this paper is an extension to a previous simpler model concerning optimization of WBAN systems based on a single sensor [11]. This study used a POMDP similar to the one used in this paper to describe the system dynamics. The central contribution of this study is the use of multiple sensors, where each activated sensor returns an independent output signal.

II. Health Sensing Model

A. Patient Health States

First, we discuss the model used to represent the underlying health state of a patient. We consider a finite set of discrete health states a patient can occupy, denoted as $\mathcal{H} = \{h_0, h_1, \dots, h_J\}$. The state h_0 denotes a terminal state in which the monitoring of the individual is no longer relevant (e.g. the patient is taken to a hospital, patient mortality, etc.). Thus, if an individual reaches the terminal state, he or she remains in that state until some intervention occurs. The transitions between any two states are given by a (stationary) transition matrix $\boldsymbol{T}$ that describes the transition probability from state $h^t \in \mathcal{H}$ at epoch t to state $h^{t+1} \in \mathcal{H}$ at epoch $t+1$:

$$\mathbf{T} = \mathbf{T}_{ji} = Pr(h^{t+1} = h_j | h^t = h_i) \quad (1)$$

Let us start by defining a basic scenario in which the health states are ordered from the least healthy state to the healthiest state. Similar to a birth/death Markov chain, transitions are defined between the pairs of adjacent states. Figure 1 presents a linear series of health states, where μ and λ denote the transition probabilities to a less healthy state and a healthier state, respectively. As noted above, h_0 is the terminal state to which the transition is allowed only from h_1. Accordingly, the transition to the healthiest state, h_J, is allowed only from h_{J-1}. Assuming the monitoring frequency is high enough, such a transition scheme seems to be appropriate.

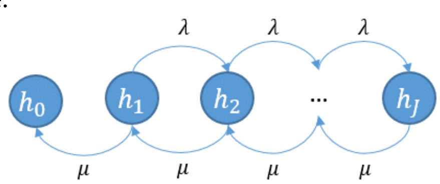

Fig. 1: Birth-death Markov chain describing transitions between health states.

B. Sensors

The health states described in the previous section are unknown to the controller, yet a set of sensors is used to produce a belief distribution over the different health states. We assume the network contains a set of N available sensors. At each decision epoch, a subset of sensors is activated to carry out measurements. We denote the binary activation sensor vector by $\boldsymbol{s^t} = (s_1^t, s_2^t, \dots s_N^t)$, where $s_n^t = 1$ refers to an activated sensor, and $s_n^t = 0$ refers to a deactivated sensor, at time t, where $n = 1, \dots, N$. Note that the time index is omitted for ease of notation when addressing general, time-independent properties henceforward.

For simplicity, we assume binary sensors, i.e. the output from an activated sensor is either 0 or 1. We define $L(\boldsymbol{s}) = \{l_1, \dots, l_N\}$ as the set of all possible output combinations of the sensors given a sensor activation vector $\boldsymbol{s}$, i.e. $l_n^t \in \{0,1,\emptyset\}$. For example, assume there are three sensors, s_1, s_2, s_3, and only s_1 and s_2 have been activated, implying $\boldsymbol{s} = (1,1,0)$. Then $L(\boldsymbol{s}) = \{(1,1,\emptyset),(1,0,\emptyset),(0,1,\emptyset),(0,0,\emptyset)\}$, where $\emptyset$ denotes a deactivated sensor.

We further assume that, given the patient health state h_j, the probability to obtain a positive output "1" (and thus a complementary negative output "0") from sensor n is known and based on the sensor design and quality. This probability is denoted by p_{nj}:

$$p_{nj} = \Pr(l_n = 1 | h_j) \quad \forall n = 1, \dots, N \,; \forall j = 0, \dots, J \quad (2)$$

The probability of the negative output "0" is $1 - p_{nj}$. These probabilities can be interpreted as sensor accuracies, and can be combined into a sensor accuracy matrix, $\boldsymbol{P}$.

C. Belief States

Since the actual health state at each epoch is unknown to the network's controller, the controller makes decisions using a belief distribution over the health states, denoted as a *belief state* $\boldsymbol{b} = (b_0, b_1, \dots, b_J)$. The belief state is estimated with conditional probabilities given the following information: i) the previous belief state; ii) the subset of activated sensors; iii) the sensors' most recent outcome signal (as a sufficient statistic); and iv) the true health state transition matrix. The belief state We thus represent the following transition function between belief states:

$$\boldsymbol{b^{t+1}} = \tau(\boldsymbol{b^t}, \boldsymbol{s^t}, \boldsymbol{l^t}, \boldsymbol{T}) \quad (3)$$

where b_j^{t+1} is the conditional probability that the patient is in health state h_j, i.e. $b_j^{t+1} = \Pr(h^t = h_j | \boldsymbol{b^t}, \boldsymbol{s^t}, \boldsymbol{l^t}, \boldsymbol{T})$.

At the initial epoch, the belief state $\boldsymbol{b^0}$ is assumed to be known. Then, the belief state evolves as a result of the decisions made with regard to the subset of sensors that are activated at each epoch, $\boldsymbol{s^t}$, the obtained outputs, $\boldsymbol{l^t}$, and the transitions between the health states, $\boldsymbol{T}$. In order to determine the evolution of the belief state, one has to estimate the probability of a positive output from sensor n given the belief state $\boldsymbol{b}$, namely:

$$\Pr(l_n = 1 | \boldsymbol{b}) = \sum_{j=0}^{J} p_{nj} b_j \quad \forall n = 1, \dots, N \quad (4)$$

By applying Bayes' theorem, we calculate the belief that the patient is in health state h_j based on the output of sensor n:

$$\Pr(h_j | l_n = 1, \boldsymbol{b}) = \frac{p_{nj} b_j}{\Pr(l_n = 1 | \boldsymbol{b})} \quad \forall n = 1, \dots, N \,; \ \forall j = 0, \dots, J \quad (5)$$

By combining all of the sensor outputs in a similar fashion, we get an expression for the controller's belief that a patient is in health state h_j:

$$\Pr\left(h_j \middle| \boldsymbol{l}, \boldsymbol{b}\right) = \frac{\Pr\left(\boldsymbol{l} \middle| h_j\right) \cdot b_j}{\Pr(\boldsymbol{l}|\boldsymbol{b})} \quad \forall j = 0, \dots, J \tag{6}$$

where $\Pr\left(\boldsymbol{l} \middle| h_j\right)$, is the probability of receiving a certain output vector $\boldsymbol{l}$ from an array of activated sensors ($\boldsymbol{l} \in L(\boldsymbol{s})$) given the patient is in health state h_j.

Finally, given the previous belief state and the output vector of the activated sensors, the new belief state $\boldsymbol{b^{t+1}}(\boldsymbol{b^t}, \boldsymbol{l^t})$ is obtained by accounting for a possible transition of the individual's health state within the epoch, i.e.,

$$\begin{aligned} b_j^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) &= [\tau(\boldsymbol{b^t}, \boldsymbol{s^t}, \boldsymbol{l^t})]_j \\ &= \sum_{j'=1}^{J} \Pr\left(h_{j'} \middle| \boldsymbol{l^t}, \boldsymbol{b^t}\right) \cdot \boldsymbol{T}_{j'j} \\ &= \frac{1}{\Pr(\boldsymbol{l^t}|\boldsymbol{b^t})} \sum_{j'=1}^{J} b_{j'}^{t} \cdot \Pr\left(\boldsymbol{l^t} \middle| h_{j'}\right) \cdot \boldsymbol{T}_{j'j} \end{aligned} \tag{7}$$

In general, given any belief state $\boldsymbol{b}$, the probability of obtaining a specific outcome $\boldsymbol{l} \in L(\boldsymbol{s})$ is:

$$\Pr(\boldsymbol{l}|\boldsymbol{b}) = \sum_{j=0}^{J} b_j \cdot \Pr\left(\boldsymbol{l} \middle| h_j\right) \tag{8}$$

D. Power and Misclassification Costs

We define two different types of cost components. The first type is the power cost of the sensors. This cost accounts for the energy consumed by activating and running the sensors. The power cost of sensor vector $\boldsymbol{s}$ is denoted by $C(\boldsymbol{s})$.

The second type of cost is the misclassification cost, which is used to discourage the system from misclassifying the patient's true health state. There are two types of misclassifications: false positives and false negatives. In this model, we are monitoring patient health states, and would ideally like to alert the patient or a doctor when the patient health state deteriorates. Because of this, a false positive in this model refers to the scenario in which the controller believes the patient is less healthy than he or she truly is. Correspondingly, a false negative refers to the case in which the controller believes the patient is healthier than the ground truth. Mathematically, for a true patient health state h_j, a false positive error refers to any $b_k > 0, \forall k = 0, \dots, \mathrm{j} - 1$. A false negative error refers to any $b_k > 0, \forall k = \mathrm{j} + 1, \dots, \mathrm{J}$. We define two constant cost parameters: i). C_{FP}, the cost of a false positive error, and ii). C_{FN}, the cost of a false negative error.

E. Risk / misclassification factor

A major consideration that should be taken into account when defining the activation policy (i.e. selecting which sensors should be activated) is the misclassification factor. This expresses the probability that the system will wrongly estimate the patient's health state in the next time step.

To express the misclassification factor in the system's control, we define the misclassification cost $\rho_j(\boldsymbol{b^t}, \boldsymbol{l^t})$ for each health state h_j as follows:

$$\begin{aligned} \rho_j(\boldsymbol{b^t}, \boldsymbol{l^t}) = C_{FP} &\sum_{j'=0}^{j-1} b_{j'}^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \\ &+ C_{FN} \sum_{j'=j+1}^{J} b_{j'}^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \end{aligned} \tag{9}$$

The misclassification cost is composed of the two types of errors discussed in the previous section (false positive and false negative). For a certain state h_j, one can calculate the probability that a patient is considered to be in a worse state than the true h_j (false positive error) and a better state than the true h_j (false negative error). These probabilities are multiplied by the cost parameters discussed above (C_{FP}, C_{FN}). It is important to note that, since the belief state $\boldsymbol{b}$ is a probability distribution, it is possible for our system to incur both false positive and false negative costs at the same time. Furthermore, the cost function in (9) can be refined so that different belief states have different misclassification errors. In particular, the cases where different health states incur different misclassification costs and different sensors incur different misclassification costs can be considered by the proposed formulation.

At every decision epoch, the controller calculates the expected misclassification cost over the entire belief state, namely:

$$\begin{aligned} &\rho(\boldsymbol{b^t}, \boldsymbol{l^t}) \\ &= \sum_{j=0}^{J} b_j^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \left(C_{FP} \sum_{j'=0}^{j-1} b_{j'}^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \right. \\ &\left. + C_{FN} \sum_{j'=j+1}^{J} b_{j'}^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \right) \end{aligned} \tag{10}$$

III. SENSOR ACTIVATION CONTROL

In the previous section, we established a model, specified how the controller updates its internal belief state (7), and defined two cost functions ($C(\boldsymbol{s})$ and $\rho(\boldsymbol{b}, \boldsymbol{l})$). In this section, we explain how the controller decides which sensors to activate based on its belief state and these cost functions.

In this model, we consider the terminal state h_0 to be an absorbing state. In practice, this may represent a situation where the patient arrives at an emergency room and requires medical intervention, or any other scenario where the health sensing is no longer relevant. As a result, the path that starts from a given initial belief state, $\boldsymbol{b^0}$, terminates in the absorption state, $\boldsymbol{b} = (1, 0, \dots, 0)$ after a finite number of decision epochs. We denote the total minimum cost of the path that starts at $\boldsymbol{b^t}$ by the value function, $V(\boldsymbol{b^t})$.

A. One-Step Look-Ahead Greedy Policy

To validate the problem definition and provide a lower bound solution, we implement a greedy algorithm. In this algorithm, the controller applies a one-step look-ahead function at each epoch and makes the decision (of what sensors to activate) that minimizes the immediate cost incurred. The value function in this greedy regime is given by:

$$V^G(\boldsymbol{b^t}) = \min_{\boldsymbol{s^t}\in S}\left\{(1-\alpha)\cdot C(\boldsymbol{s^t}) + \alpha \cdot \sum_{\boldsymbol{l}\in L(\boldsymbol{s^t})} Pr(\boldsymbol{l^t}|\boldsymbol{b^t})\cdot \rho(\boldsymbol{b^t},\boldsymbol{l^t})\right\} \quad (11)$$

where α is a sensitivity analysis parameter, which enables us in later stages to analyze the dynamics of the model given different weights for the activation and misclassification costs. The default value of α is 0.5.

One simple extension to the greedy approach is a k-step greedy approach, in which the controller calculates the best decision based on the next k decision epochs.

B. Dynamic Programming Policy

The optimal policy is determined by means of dynamic programming as follows:

$$V^O(\boldsymbol{b^t}) = \min_{\boldsymbol{s^t}\in S}\left\{(1-\alpha)\cdot C(\boldsymbol{s^t}) + \alpha \cdot \sum_{\boldsymbol{l}\in L(\boldsymbol{s^t})} Pr(\boldsymbol{l^t}|\boldsymbol{b^t}) \cdot \left(V^O\left(\boldsymbol{b^{t+1}}(\boldsymbol{b^t},\boldsymbol{l^t})\right) + \rho(\boldsymbol{b^t},\boldsymbol{l^t})\right)\right\} \quad (12)$$

From this value function, we can extract an optimal policy by selecting the $\boldsymbol{s^t}$ which minimizes the value function for the belief state, i.e. $\underset{\boldsymbol{s^t}\in S}{\operatorname{argmin}}\, V^O(\boldsymbol{b^t})$.

C. Belief State Discretization

The belief state was defined in the previous section as a belief vector of health state probabilities over a continuous domain. However, the dynamic programming formulation in (12) explores all possible future states, and with a continuous belief state, the state space is infinite. In order to develop a numerical procedure to approximate the value function V^O, we discretize the belief state space. That is, given a belief state vector from an infinite set of probability vectors, we produce a new belief state vector from a finite set of probability vectors. To this end, we make use of the following naïve discretization method:

1. Create a set B which contains all valid probability vectors for the given level of discretization
2. Given a non-valid vector $\boldsymbol{b}$, calculate the distance between $\boldsymbol{b}$ and each valid vector $\overline{\boldsymbol{b}} \in B$, according to the Manhattan metric: $\sum_j |\overline{b}_j - b_j|$.
3. Return the vector $\overline{\boldsymbol{b}}^*$closest to $\boldsymbol{b}$.

For example, if the level of discretization is 0.25 and $\boldsymbol{b} = [0.26\ 0.24\ 0.45\ 0.05]$, then the algorithm returns $\overline{\boldsymbol{b}}^* = [0.25\ 0.25\ 0.5\ 0.0]$ as the closest valid vector.
After discretizing the state space as described above, we calculate $V^O(\overline{\boldsymbol{b}})$ for each valid state $\overline{\boldsymbol{b}} \in B$. The algorithm below implements the known value iteration method [12]:

1. Run the discretization algorithm and create the set of discretized belief states B.
2. Set the initial value of the cost function at zero, $V^{current}(\overline{\boldsymbol{b}}) = 0, \forall \overline{\boldsymbol{b}} \in B$.
3. While the stopping criterion is not satisfied (described after the algorithm):
   - 3.1. Save the latest values and policies for each belief state as V^{old},
   - 3.2. For each belief state $\overline{\boldsymbol{b}}$:
     - 3.2.1. Calculate the value function $V^{current}(\overline{\boldsymbol{b}}^t)$ using (12), where the term $V^{current}\left(\boldsymbol{b^{t+1}}(\overline{\boldsymbol{b}}^t,\boldsymbol{l^t})\right)$ is calculated as follows:
       - 3.2.1.1. Perform discretization for $\boldsymbol{b^{t+1}}(\overline{\boldsymbol{b}}^t,\boldsymbol{l^t})$
       - 3.2.1.2. Extract the value function value for $\overline{\boldsymbol{b}}^{t+1}$ using $V^{old}(\overline{\boldsymbol{b}}^{t+1})$
4. Return $V^{current}$, from which the optimal policy can be extracted for each discretized belief state $\overline{\boldsymbol{b}}$.

The stopping criteria for the algorithm is met when the difference between two subsequent iterations is smaller than δ, which is defined as:

$$\delta = Max_{\overline{\boldsymbol{b}}\in B}\left|V^{current}(\overline{\boldsymbol{b}}) - V^{old}(\overline{\boldsymbol{b}})\right|$$

IV. Numerical Examination

A. Simulation Parameters

In this section, we provide a small numerical example of the model dynamics. We define the model parameters as follows:

- $J = 4$ denotes the number of health states,
- $N = 5$ denotes the number of sensors,
- $\boldsymbol{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ .25 & .7 & .05 & 0 \\ 0 & .2 & .75 & .05 \\ 0 & 0 & .2 & .8 \end{bmatrix}$ denotes the stochastic transition matrix of the true patient health state,
- $\boldsymbol{P} = \begin{bmatrix} .99 & .02 & .5 & .55 \\ .6 & .9 & .05 & .5 \\ .5 & .45 & .95 & .01 \\ .99 & .01 & .5 & .5 \\ .6 & .4 & .5 & .5 \end{bmatrix}$ denotes the sensor accuracy matrix,
- $C_s = [10 \quad 20 \quad 15 \quad 5 \quad 2]$ represents the sensors' activation costs,
- $C_m = [150 \quad 750]$ represents the misclassification costs (FP, FN).

The cost parameter values described above were selected in order to allow insightful analysis, provide intuition, and reflect a clear trade-off between the sensor activation and health state misclassification costs. In practice, these values can be estimated more accurately from real use-cases and by domain experts. For example, the misclassification costs may represent the cost of the medical care needed in the case of misclassifying a patient's health state.

To simplify the numerical implementation of the method, we assume that the sensors' outputs are conditionally independent from each other given the true health state, i.e.

$$\Pr(l_n = 1, l_{n'} = 1 | h_j) = p_{nj} \cdot p_{n'j} \quad (13)$$

Thus, the probability to obtain a certain combination of sensor outputs $\boldsymbol{l} \in L(\boldsymbol{s})$ is calculated by multiplying the probabilities of receiving each individual sensor output, i.e.:

$$\Pr(\boldsymbol{l}|h_j) = \prod\nolimits_{n|l_n=1} p_{nj} \cdot \prod\nolimits_{n|l_n=0} (1 - p_{nj}) \qquad (14)$$

Now, one can calculate the health state distribution shown in (7) and (8) as follows:

$$\Pr(h_j|\boldsymbol{l}, \boldsymbol{b}) = \frac{\Pr(\boldsymbol{l}|h_j) \cdot b_j}{\Pr(\boldsymbol{l}|\boldsymbol{b})}$$
$$= \frac{b_j \cdot \prod_{n|l_n=1} p_{nj} \cdot \prod_{n|l_n=0} (1 - p_{nj})}{\left(\prod_{n|l_n=1} \sum_j p_{nj} b_j\right) \cdot \left(\prod_{n|l_n=0} \sum_j (1 - p_{nj}) b_j\right)} \qquad (15)$$
$$\forall j = 0, \ldots, J$$

Such a simplifying assumption allows closed-form calculations for the outcome probabilities. In future research we plan to relax this assumption and consider possible dependencies among the sensors' outputs.

B. *Results*

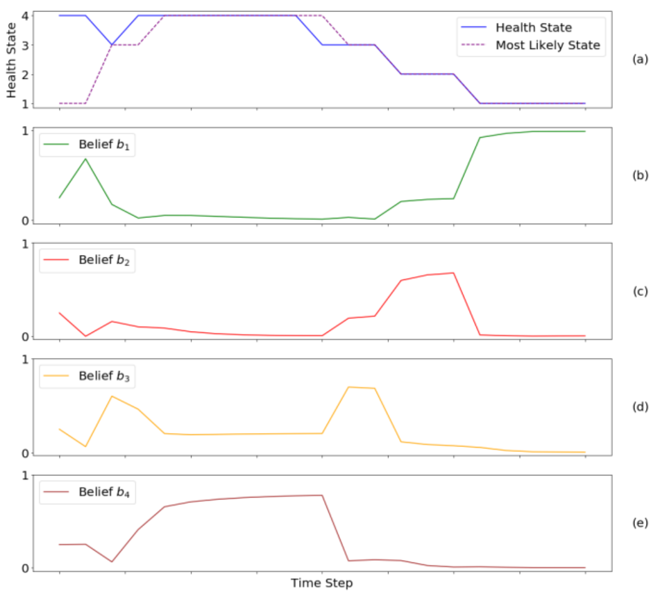

Fig. 2: (a) the (unknown) health state path over 20 decision epochs as a result of the health state dynamics and the most likely state using a greedy policy; (b)-(e) the belief state distribution for each state corresponding to the evolution of the health states.

We first simulate the greedy algorithm described in Section III.A to derive a greedy policy. It is important to note that while the greedy policy is suboptimal, it operates on the continuous belief state space, not the discretized space used by our dynamic programming algorithm. The most likely state shown in Figure 2(a) is generated by $h^{ML} = argmax_h(b)$. One can observe that the greedy policy generally provides accurate predictions of the actual health states.

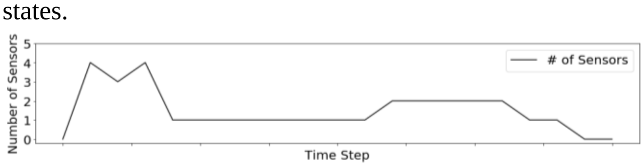

Fig. 3: The number of sensors activated throughout the simulation using the greedy policy

Figure 3 presents the number of sensors activated throughout the simulation. One can observe that the largest number of sensors is activated mostly during health state transitions, i.e. at periods where the patient's health is relatively unstable. At the end of the simulation, when the status evolves to the terminal state, the controller doesn't activate any sensors. In order to better understand the system dynamics, we now demonstrate the trade-off between the misclassification cost and activation cost. By varying parameter α between 0.05 and 0.95, we change the weight of the two cost components in (11).

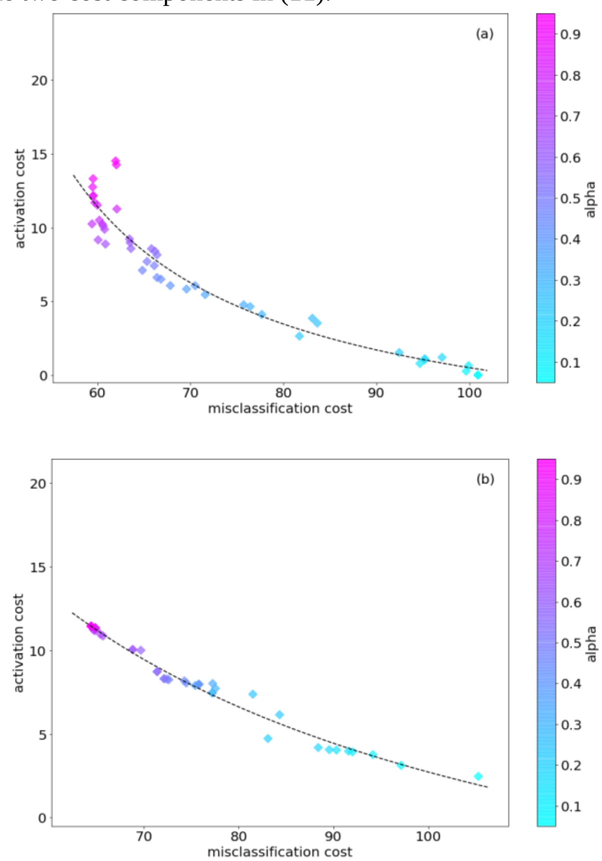

Fig. 4: Average activation costs vs. average misclassification costs per α, over 30 i.i.d Monte Carlo simulations for the greedy solution (a) and dynamic programming solution (b). The graph was generated using 46 values of α between 0.05 and 0.95

In Figure 4, one can observe the trade-off between the activation costs and the misclassification costs. For higher values of α, the misclassification costs have more influence, therefore more sensors are activated to minimize the overall misclassification costs. For lower values of α, the opposite occurs: fewer sensors are activated, and the misclassification costs increase. We now compare the two policies. Note that, generally, dynamic programming using value iteration provides an optimal solution; however, because of the state space discretization, as described in the previous section, the dynamic programming solution in this case might be sub-optimal. To allow a more nuanced comparison of the two algorithms, we modify the sensors' accuracy matrix such that the considered sensors are less accurate, which results in more complex and contrasting policies returned by the algorithms:

$$\boldsymbol{P} = \begin{bmatrix} .8 & .15 & .5 & .55 \\ .6 & .85 & .15 & .5 \\ .5 & .45 & .75 & .3 \\ .8 & .15 & .5 & .5 \\ .6 & .4 & .5 & .5 \end{bmatrix}$$

The comparison of the two policies is shown in Figure 5.

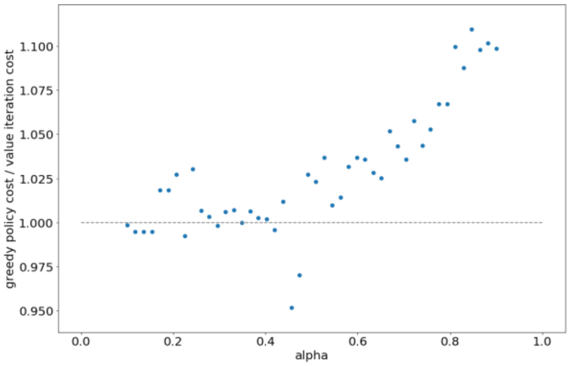

Fig. 5: Ratio between the average total cost of the greedy and value iteration solution, for each α. The dotted line marks the ratio value of 1; points above this line represent values of α for which the greedy solution returned a higher cost than the dynamic programming solution.

As expected, the average total costs incurred using the greedy solution is generally higher than the same costs incurred using the value iteration solution. This is due to the fact that the greedy policy is a heuristic solution, whereas the dynamic programming solution is an optimal solution (as discussed previously, the dynamic programming solution is an approximation of the optimal solution due to the use of discretization, which may explain the outlier values of α for which the greedy algorithm returned a lower cost).

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have addressed the optimal health sensing problem by building a model of a WBAN used for health sensing purposes. This model may be used for optimizing the trade-off between two different types of costs: i) power cost, which accounts for the system's energy consumption when activating a certain subset of sensors, ii) misclassification cost, which accounts for the probability of error when classifying a patient's health state. The model is based on a POMDP, since the actual health states are unknown to the controller. We have proposed two algorithms for setting a sensor activation policy. The "greedy" algorithm applies a one-step look ahead approach. The second algorithm is based on a dynamic programming framework, which was solved using value iteration over a discretized belief state space.

We have presented a numerical examination of the two algorithms. For the selected parameters, the greedy algorithm provided a relatively accurate sensing policy, successfully approximating the patient's actual health state. In addition, a smooth change in the relative weights of the power and misclassification costs results in a relatively smooth change in the number of sensors used by the returned policies. We then compared the two policies, and although the discretization process limits the dynamic programming algorithm's optimality, in general the dynamic programming policy returned a lower cost than the returned cost of the greedy policy. These results indicate that a WBAN patient health monitoring system could be beneficial in certain application areas. However, when discussing the use of remote systems for health monitoring purposes, some domain specific issues should be considered. Primarily, the accuracy of the controlled monitoring activity is crucial, since misclassifying a patient's health state could have grave implications. In addition, there are a large variety of patients and health situations, making it difficult for a single WBAN systems to perform well across this spectrum. Therefore, a promising avenue of future research would be to implement more intelligent techniques that could learn the patient's health parameters (e.g. health state transition matrix) online, thus dynamically boosting the system's accuracy while reducing the overall energy consumption.

ACKNOWLEDGMENTS

This paper was supported by the Digital Living 2030 grant and the Koret foundation grant for Smart Cities and Digital Living.

REFERENCES

[1] D. S. Zois, M. Levorato, and U. Mitra, "Energy-efficient, heterogeneous sensor selection for physical activity detection in wireless body area networks," *IEEE Transactions on signal processing*, vol. 61, no. 7, pp. 1581-1594, 2003.

[2] E. Jovanov, A. Milenkovic, C. Otto, P. De Groen, B. Johnson, S. Warren, & G. Taibi, "A WBAN system for ambulatory monitoring of physical activity and health status: applications and challenges," *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, Shanghai, pp. 3810-3813, 2005.

[3] M. Ghamari, B. Janko, R. Sherratt, W. Harwin, R. Piechockic, and C. Soltanpur, "A Survey on Wireless Body Area Networks for eHealthcare Systems in Residential Environments," *Sensors*, vol. 16, no. 6, pp. 831, 2016.

[4] A. Milenković, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," *Computer Communications*, vol. 29, nos. 13-14, pp. 2521–2533, 2006.

[5] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao, and V. C. M. Leung, "Body Area Networks: A Survey," *Mobile Networks and Applications*, vol. 16, no. 2, pp. 171–193, 2010.

[6] Y. Wang, B. Krishnamachari, Q. Zhao, and M. Annavaram, "Markov-optimal sensing policy for user state estimation in mobile devices," *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks - IPSN 10*, pp. 268–278, 2010.

[7] Y. Wang, B. Krishnamachari, and M. Annavaram, "Semi-Markov state estimation and policy optimization for energy efficient mobile sensing," *2012 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pp.533-541, 2012.

[8] L. K. Au, A. A. T. Bui, M. A. Batalin, and W. J. Kaiser, "Energy-Efficient Context Classification With Dynamic Sensor Control," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 6, no. 2, pp. 167–178, 2012.

[9] A. Panangadan, S.M. Ali and A. Talukder, "Markov decision processes for control of a sensor network-based health monitoring system," *Proceedings of the National Conference on Artificial Intelligence*, Vol. 20, No. 3, pp. 1529. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

[10] V. Krishnamurthy, *Partially Observed Markov Decision Processes*. Cambridge University Press, 2016.

[11] D. Miller, Z. Zhou, N. Bambos and I. Ben-Gal, (in press). "Optimal Sensing for Patient Health Monitoring," *International Communications Conference*, 2018.

[12] D. P. Bertsekas, *Dynamic programming and optimal control*, vols. 1, 2. Belmont: Athena Scientific, 1995.