Learning Health State Transition Probabilities via Wireless Body Area Networks

Tal Geller, Yair Bar David, 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 (WBAN) for remote health monitoring applications. A partially observable Markov decision process is used to describe the information flow and behavior of the WBAN. We then discuss a sensor activation policy, used for optimizing the trade-off between power consumption and probability of patient health state misclassification. In order to determine the underlying health state transition probabilities, by which a patient's health state evolves, we develop a learning algorithm which uses the data collected from a group of patients, each being monitored by a WBAN. Finally, a numerical examination demonstrates the applicability of such a system, which applies the learning process and sensor activation policy simultaneously.**

Keywords **— Wireless body area networks, controlled sensing, partially observable Markov decision processes (POMDP), transition probability learning.**

I. INTRODUCTION

In this paper, we explore the implementation of learning techniques in wireless body area networks (WBAN). WBANs usually consist of an array of sensors which may be placed on or near an individual, to continuously measure different physiological parameters. In addition, WBANs include a controlling unit, or a controller, which is responsible for managing the sensors, along with analyzing the data or transferring the data to an additional computing unit. One of the main applications of these systems is to estimate the physical and health condition of an individual [1], [2], [6].

Due to recent advances in sensor and wireless communications technologies, the use of WBANs has been on the rise [2], [6]. Extensive research has been conducted in order to define the characteristics needed for successful implementation of WBANs in modern real-world applications [3], [6]. Specifically, the inherent trade-off between energy consumption and accuracy encapsulates a number of design and implementation decisions which should be taken into account when developing WBAN systems [1]. Along with other limitations, energy consumption is still one of the main factors that hinder the advancement of these systems. Most research done in the field concerns optimizing hardware and communication components, such as the sensors and the wireless communication protocols between the sensors and the controlling unit, in order to reduce energy consumption [7], [8]. Some research aimed to increase energy efficiency by optimizing the controlling algorithms, which usually concern sensor selection given the information gained throughout the system's activity [2], [9].

In order to develop the controlling policies, partially observable Markov decision processes (POMDP) have been used to model the behavior and transition of information in WBANs [4], [6], [10]. A POMDP is defined by a set of states, actions, conditional transitions between the states, a cost function and a set of observations [5]. In most studies, the POMDP parameters are assumed to be known or somehow approximated, such as the transition probabilities between the states, or the states themselves [6]. In real world applications, the POMDP parameters are rarely known. Thus, setting assumptions concerning the POMDP parameters is needed in order to solve the POMDP model. Such assumptions may lead to noisy estimations and increased uncertainty in the system. Therefore, learning techniques may be applied in order to collect information concerning these parameters, thus unveiling the underlying information of the POMDP. In this paper, we develop a learning algorithm, which approximates the underlying transition probability matrix of a POMDP, using data collected from a group of patients, which share similar transition probabilities. The described algorithm serves as a "heuristic caricature" of the problem. We present an empirical verification of the ideas on an interesting exemplary scenario, with the goal of introducing the problem and spotlighting its design elements and application potential.

II. PREVIOUS WORK

The learning model developed in this paper is an extension to a simpler WBAN model which assumes full knowledge of all POMDP parameters [1]. The central contribution of this paper concerns applying learning techniques to estimate the transition matrix, which is unknown to the controller.

III. HEALTH SENSING MODEL

A. Patient Health States

We first discuss the health state transitions and information flow in the WBAN system. We assume a finite set of health states a patient can occupy, denoted by $\mathcal{H} = \{h_1, h_2, \dots, h_J\}$, such that h_J is considered the healthiest state, and h_1 the least healthy state. Contrary to [1], we don't require the existence of a finite health state where the sensing is no longer relevant. This allows continuous learning based on the feedback from the sensors.

The transition probability between any two states during two consecutive epochs $h^t, h^{t+1} \in \mathcal{H}$ is given by a transition matrix $\boldsymbol{T}$:

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

The most significant alteration to the model described in [1] is that the transition probabilities between the health states are unknown to the controller. An additional extension is that transitions are possible between all pairs of the health states.

B. Sensors

Since the current actual health state of the patient is unknown to the controller, we use the information collected by a network of $\boldsymbol{N}$ sensors to produce a probability distribution over the health states. This distribution is defined as a belief state. At each decision epoch t, the controller may activate any subset of the sensors. The activated sensors at epoch t are denoted 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, where $n = 1, \dots, N$. Note that for ease of notation, the time index may be omitted when discussing general time-independent properties.

We define $\boldsymbol{l^t}$ as the output vector of all sensors at epoch t and $L(\boldsymbol{s})$ as the set of all possible output vectors for a sensor activation vector $\boldsymbol{s}$. We assume that given a patient's health state, the probability for a certain sensor output is known. For example, given a diabetic patient's health state, the probability that a blood sugar level sensor will return a certain value may be obtained from a known distribution. For simplicity, in this paper we assume the sensors provide binary output of either "1" or "0", i.e. $l_n^t \in \{0,1,\emptyset\}$, where $\emptyset$ denotes a deactivated sensor. The probability to receive an output of "1" from sensor n given that the individual is currently in health state j is denoted by p_{nj}:

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

Accordingly, the complementary probability to receive a signal "0" is $1 - p_{nj}$. This definition may be interpreted as the sensors' accuracies.

C. Belief States

The controller estimates an individual's health state using a belief distribution over the health states, denoted by $\boldsymbol{b} = \left(b_1, b_2, \dots, b_J\right)$. During the sensing period, the belief states evolve given the information collected by the activated sensors $\boldsymbol{s^t}$, the sensors' outcomes $\boldsymbol{l^t}$. Generally, the transition matrix $\boldsymbol{T}$ may also be taken into account while calculating the belief states. However, since the controller in this use case has no knowledge of the actual belief states, he must use an estimation of $\boldsymbol{T}$. We denote the transition matrix estimated by the controller at epoch t by $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}}$. In section V, we discuss the estimation process of the transition matrix. Given an estimated transition matrix, the transition function between belief states is denoted as follows:

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

where each component b_j^{t+1} represents the conditional probability that the patient is in health state h_j, i.e. $b_j^{t+1} = \Pr(h^{t+1} = h_j | \boldsymbol{b^t}, \boldsymbol{s^t}, \boldsymbol{l^t}, \widetilde{\boldsymbol{T}}_{\boldsymbol{t}})$. Since the controller has no prior knowledge concerning the initial health state or the corresponding sensor outputs, the initial belief state $\boldsymbol{b^0}$ is randomized.

In order to determine a patient's belief state at epoch t, where $t > 0$, we first calculate the probability to receive a positive output signal from sensor n, given the previous belief state, namely:

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

We then apply Bayes' theorem to calculate the belief that the patient is in health state h_j based on the output of sensor n:

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

By calculating the above probability for each of the sensors' outputs l_n, where $n = 1, \dots, N$ and combining the outcomes, we obtain an expression for the controller's belief that a patient is in health state h_j:

$$\Pr\left(h_j \middle| \boldsymbol{l}, \boldsymbol{b}\right) = \frac{\Pr\left(\boldsymbol{l} \middle| h_j\right) \cdot b_j}{\Pr(\boldsymbol{l}|\boldsymbol{b})} \quad \forall j = 1, \dots, J \tag{6}$$

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

Finally, the new belief state $\boldsymbol{b^{t+1}}(\boldsymbol{b^t}, \boldsymbol{l^t})$ is calculated by accounting for the estimated transition probabilities of the patient's health state within the epoch, i.e.,

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

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

$$\Pr(\boldsymbol{l}|\boldsymbol{b}) = \sum\nolimits_{j=1}^{J} b_j \cdot \Pr\left(\boldsymbol{l} \middle| h_j\right) \tag{8}$$

D. Power and Misclassification Costs

We define two types of cost components. The first type accounts for the energy consumed by activating the sensors. The cost of activating an array of sensors $\boldsymbol{s}$ is denoted by $C(\boldsymbol{s})$. The second type is the misclassification costs, which accounts for the probability of error in the classification of a patient's true health state. This cost is separated into two different types of misclassifications: false positive and false negative.

The false positive error refers to the case where a patient's true health state is healthier than estimated by the controller. Correspondingly, the false negative error refers to the case where the patient's health state is less healthy than estimated by the controller. The definitions of the false positive and false negative errors correspond to the purpose of a health monitoring system, where we would like to alert the patient, or medical staff, of a patient's deteriorating health state. 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.

We note that additional methods for defining the cost structure may be incorporated into this model, such as dynamic costs over time or different costs per health state. More sophisticated cost methods may allow more flexible behavior of the controller according to the system states.

E. Risk / misclassification factor

Due to the importance of accuracy in health monitoring systems, minimizing the misclassification factor is a major consideration when defining the sensor activation policy.

In order to express the misclassification factor, we define the misclassification cost per health state h_j, denoted by $\rho_j(\boldsymbol{b^t}, \boldsymbol{l^t})$, as follows:

$$\rho_j(\boldsymbol{b^t}, \boldsymbol{l^t}) = C_{FP} \sum_{j'=1}^{j-1} b_{j'}^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) + C_{FN} \sum_{j'=j+1}^{J} b_{j'}^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \quad (9)$$

thus taking into account both the false positive and false negative errors, described in the previous section. For a certain health state h_j, we multiply the false positive cost C_{FP} by the sum of belief probabilities that the patient is in a worse state than h_j. Similarly, we multiple the false negative cost C_{FN} by the sum of belief probabilities that the patient is in a better state than h_j. At each decision epoch, the controller calculates the total estimated misclassification cost, over the entire belief state, as follows:

$$\rho(\boldsymbol{b^t}, \boldsymbol{l^t}) = \sum_{j=1}^{J} b_j^{t+1}(\boldsymbol{b^t}, \boldsymbol{l^t}) \cdot \rho_j(\boldsymbol{b^t}, \boldsymbol{l^t}) \quad (10)$$

We note that since the belief state $\boldsymbol{b}$ is a probability distribution, the system may incur both false positive and false negative costs at the same time.

IV. Sensor Activation Control

In order to balance the energy consumption and possible health state misclassification in the system, the controller must use a sensor activation policy which considers both factors and dictates which sensors are activated at each epoch. In this paper, we assume a greedy policy, which applies a simple one-step look ahead approach. At each epoch, the controller tries to minimize the immediate cost incurred by activating a certain subset of sensors. The function used to minimize these costs is given by:

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

We note that additional sensor activation policies may be used, such as a greedy *k*-step look-ahead, or policies based on dynamic programming methods (which may provide an optimal solution) [1].

V. Transition matrix learning

A. Learning Process

In this section, we describe the process of estimating the transition matrix $\boldsymbol{T}$, which is assumed to be shared by a group of individuals, denoted by P, with similar physical and health condition properties, such as age, sex, overall health, etc. This could be used in order to characterize the advancement of a disease within a certain population. An example for a relevant use case would be identifying and characterizing the different stages and the transition probabilities between the stages in Alzheimer's disease in men aged 50 or higher [11].

One of the main benefits of collecting sensor information from a group of individuals is that we may commence the learning process once the sensing period begins, as opposed to performing off-line learning for collecting data. This is due to the fact that we can collect a substantial amount of data used as input for the learning process, during the sensing period. In addition, this allows us to apply the sensor activation policy once the sensing begins, thus potentially optimizing the performance of each individuals WBAN system (and saving energy costs).

In order to identify a generalized transition matrix for the patients in P, we use the information collected from the sensors, which perform sensing for all the individuals in the group separately. Matrix $\boldsymbol{T}$ is the real transition matrix by which the patients' true health states evolve. It is important to note that although $\boldsymbol{T}$ is shared by all patients in P, each patient's health trajectory is independent from other patients and therefore may evolve differently. Since $\boldsymbol{T}$ is unknown to the controller, the controller must learn and continuously estimate the transition matrix. The estimated transition matrix is used to calculate the patients' belief states and derive a sensor activation policy.

At each epoch t, the controller produces a belief state for each patient $p \in P$ given the information collected from the patient's sensors, the patient's previous belief state and the most recently estimated transition matrix $\boldsymbol{\widetilde{T}_{t-1}}$. We denote the belief state produced by the controller for patient p at epoch t, as $\boldsymbol{b_p^t}$. The controller then updates $\boldsymbol{\widetilde{T}_{t-1}}$, as will be explained in the next section. This process is repeated throughout the sensing period. At the beginning of the sensing period, the transition matrix used by the controller, $\boldsymbol{\widetilde{T}_0}$, may be guessed according to previous knowledge, domain expertise or initialized randomly.

Generally, the learning process described above may continue indefinitely. In some cases, the process of learning the transition matrix may become irrelevant, or impossible, after a long period. One example for such a case would be the existence of a terminal health state – a state which when arrived to, no further transition to other states occur throughout the sensing period. Practically, this state could represent a situation where urgent medical attention is required. Given a terminal state exists, it can be assumed that all patients will eventually arrive to the terminal health state. When this happens, learning the transition matrix would no longer be relevant. Furthermore, the controller would probably deactivate all sensors for all patients, thus making the learning process obsolete, since no additional data is received from the sensors. In order to identify cases where the learning may become obsolete, we define stopping criteria, as shown in the following section.

B. Updating the transition matrix

At each epoch t, the controller first updates the patients' belief states. Then, it updates the estimated transition matrix, considering the evolution of the patients' belief states during the latest epoch. In order to do so, we rely on the premise that the belief states fairly estimate the patients' true health states. This allows us to assume that the transitions between the belief states may be used to estimate the transition probabilities between the real health states.

The transition probabilities between the belief states, denoted $\boldsymbol{T}^*$, is calculated by minimizing the total differences between the belief states produced at epoch t, $\boldsymbol{b_P^t}$, and the expected belief states given the transition probabilities $\widetilde{\boldsymbol{T}}_{\boldsymbol{t-1}}$ and belief states at epoch $t-1$, $\boldsymbol{b_p^{t-1}}$. At each epoch, we define two matrices: $\boldsymbol{M}$ and $\boldsymbol{N}$. Each matrix contains the belief states for all patients at epochs $t-1$ and t accordingly. The matrices have dimensions $P \times J$, such that each row contains the belief state vector for a single patient, i.e.:

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{b_1^{t-1}} \\ \boldsymbol{b_2^{t-1}} \\ \vdots \\ \boldsymbol{b_P^{t-1}} \end{bmatrix}, \boldsymbol{N} = \begin{bmatrix} \boldsymbol{b_1^{t}} \\ \boldsymbol{b_2^{t}} \\ \vdots \\ \boldsymbol{b_P^{t}} \end{bmatrix}$$

We define a distance function for minimizing the discrepancies between the belief states $\boldsymbol{N}$, and the expected belief states, in matrix form:

$$D = dist(\boldsymbol{M} \cdot \boldsymbol{T}^*, \boldsymbol{N}) \qquad (12)$$

There are numerous distance definitions which can be used to calculate (12). In this paper, we examine a form similar to the Euclidean distance, as follows:

$$D = \|\boldsymbol{N} - \boldsymbol{M} \cdot \boldsymbol{T}^*\|_F^2 \qquad (13)$$

We later use $\boldsymbol{T}^*$ to update the transition matrix estimation used by the controller. $\|\boldsymbol{X}\|_F$ signifies the Frobenius norm of matrix $\boldsymbol{X}$. By differentiating D w.r.t $\boldsymbol{T}^*$, we receive the following expression:

$$\frac{\partial D}{\partial \boldsymbol{T}^*} = -2\boldsymbol{M^T N} + 2\boldsymbol{M^T M T}^* \qquad (14)$$

By equating the derivative to zero, we receive an expression for $\boldsymbol{T}^*$:

$$\boldsymbol{T}^* = (\boldsymbol{M^T M})^{-1}\boldsymbol{M^T N} \qquad (15)$$

We now update the estimated transition matrix using exponential smoothing with parameter $\alpha \in [0,1]$, which acts as a learning rate parameter:

$$\widetilde{\boldsymbol{T}}_{\boldsymbol{t}} = \alpha \cdot \boldsymbol{T}^* + (1-\alpha) \cdot \widetilde{\boldsymbol{T}}_{\boldsymbol{t-1}} \qquad (16)$$

We note that exponential smoothing allows control over the learning rate, compared to other averaging methods, thus allowing better process stability. We analyze the effect of α in the numerical examination provided in section VI.

We note that the rows of $\boldsymbol{T}^*$ are all summed up to 1, but it is possible for $\boldsymbol{T}^*$ to contain negative values. If, as a result, $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}}$ contains a negative value, the corresponding row must be normalized. The most straightforward method for normalization consists of equating the negative values in $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}}$ to zero, and dividing each value in the row by the sum of values in the row. This ensures that each row of $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}}$ contains only non-negative values and the sum of each row is 1. Thus, $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}}$ remains a valid transition matrix.

As discussed in the previous section, we now define a stop criterion. At every epoch, we calculate the determinant of $\boldsymbol{M^T M}$. In case there are multiple identical rows in $\boldsymbol{M}$ (e.g. many patients have arrived to the terminal state, in which case their belief states are similar), the determinant will be close to zero. The determinant value decreases as the number of identical rows in $\boldsymbol{M}$ increases. Once the determinant value decreases beyond a certain threshold, the learning process stops. The last transition matrix learned by the controller is used for continuous health monitoring for patients where the monitoring is still relevant.

To summarize, the algorithm below implements the proposed learning process at t:

1. Set matrix $\boldsymbol{M}$ using all patients' belief states during previous epoch $t-1$
2. Set matrix $\boldsymbol{N}$ using belief states at current epoch t
3. Calculate matrix $\boldsymbol{T}^*$ according to (15)
4. Update the latest estimated transition matrix used by the controller, $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}} = \boldsymbol{\alpha} \cdot \boldsymbol{T}^* + (\boldsymbol{1}-\boldsymbol{\alpha}) \cdot \widetilde{\boldsymbol{T}}_{\boldsymbol{t-1}}$
5. Normalize $\widetilde{\boldsymbol{T}}_{\boldsymbol{t}}$ if needed:
   5.1. Replace negative values with 0
   5.2. Divide all values by sum of row
6. Continue while stop criterion isn't met

VI. NUMERICAL EXAMINATION

A. Simulation Parameters

In this section, we provide a numerical example to demonstrate the dynamics of the WBAN behaviour and the transition matrix learning process. We simulate the health trajectories of a group of individuals. In addition, we apply the sensor activation policy, by which the controller estimates the individuals' belief states. In this example, we assume the probability of a patient staying in his current health state is generally higher than the probability to transit to a different health state. In addition, we define a terminal health state, thus encapsulating a real world assumption. We define the model parameters as follows: $P = 100$ (the number of patients), $J = 4$ (the number of health states), $N = 3$ (the number of sensors). Additional parameters:

- $\boldsymbol{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ .08 & .9 & .02 & 0 \\ 0 & .08 & .9 & .02 \\ 0 & 0 & .05 & .95 \end{bmatrix}$ denotes the real transition matrix of the true patient health state,
- $\boldsymbol{A} = \begin{bmatrix} .99 & .02 & .5 & .55 \\ .6 & .9 & .05 & .5 \\ .5 & .45 & .95 & .01 \end{bmatrix}$ denotes the sensor accuracy matrix,
- $C_s = [10 \quad 20 \quad 15]$ denotes the sensors' activation costs,
- $C_m = [750 \quad 3750]$ denotes the misclassification costs (FP, FN),
- $\widetilde{\boldsymbol{T}}_{\boldsymbol{0}} = \begin{bmatrix} .5 & .3 & .1 & .1 \\ .2 & .5 & .2 & .1 \\ .1 & .2 & .5 & .2 \\ .1 & .1 & .3 & .5 \end{bmatrix}$ denotes the initial transition matrix used by the controller.

We note that the algorithm is robust with respect to the initial transition matrix, which has little impact on the learning process. In addition, the cost parameters were selected in order to reflect a clear trade-off between the misclassification costs and the sensors' activation cost. This allows simpler analysis and insights concerning the model. In practice, these values can be estimated based on real use-cases or domain knowledge. For example, the sensor activation costs can be based on the cost of recharging the sensors and the misclassification costs can be estimated based on the cost of medical care needed due to health state misclassification.

For simplicity of the numerical examination, we assume conditional independence between the sensors' outputs given the true health state, i.e.

$$\Pr\left(l_n = 1, l_{n'} = 1 \middle| h_j\right) = p_{nj} \cdot p_{n'j} \tag{17}$$

Thus, the probability of receiving a certain combination of sensor outputs $\boldsymbol{l} \in L(\boldsymbol{s})$ can be calculated by multiplying the probabilities of receiving each of the sensors outputs given the true health state, i.e.:

$$\Pr\left(\boldsymbol{l} \middle| h_j\right) = \prod\nolimits_{n|l_n=1} p_{nj} \cdot \prod\nolimits_{n|l_n=0} (1 - p_{nj}) \tag{18}$$

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

$$\Pr\left(h_j \middle| \boldsymbol{l}, \boldsymbol{b}\right) = \frac{\Pr\left(\boldsymbol{l} \middle| h_j\right) \cdot b_j}{\Pr(\boldsymbol{l}|\boldsymbol{b})} = \frac{b_j \cdot \prod_{n|l_n=1} p_{nj} \cdot \prod_{n|l_n=0} (1 - p_{nj})}{\left(\prod_{n|l_n=1} \sum_j p_{nj} b_j\right) \cdot \left(\prod_{n|l_n=0} \sum_j (1 - p_{nj}) b_j\right)} \quad \forall j = 1, \dots, J \tag{19}$$

thus allowing 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*

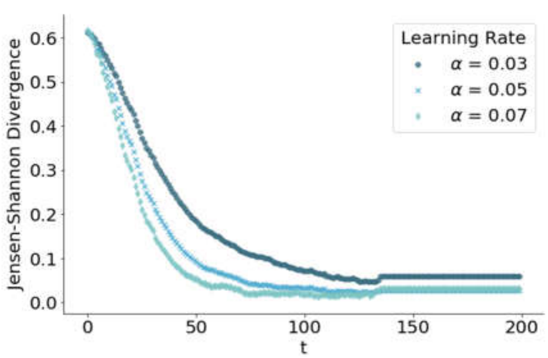

Fig. 1: Jensen-Shannon divergence between the real transition $\boldsymbol{T}$ and the transition matrix estimated by the controller $\widetilde{\boldsymbol{T}_t}$, for different learning rates

The distance between one matrix to another at each epoch t, as seen in the graph above, is calculated by the Jensen-Shannon divergence between each pair of rows in both matrices, and summing up the obtained distances. The Jensen-Shannon divergence is an information based measure, typically used for measuring distances between distributions. Therefore, this measure effectively represents the distances between each pair of rows in the transition matrices. We note that even though the distance D is Euclidean based and the distance shown in the graph is information based, the algorithm has reached convergence. One can observe that the transition matrix is estimated rather successfully, as the latest estimated transition matrix learned by the controller, using $\alpha = 0.05$, is:

$$\widetilde{\boldsymbol{T}}_{\boldsymbol{final}} = \begin{bmatrix} .995 & 0 & .004 & .001 \\ .075 & .912 & .013 & 0 \\ .035 & .12 & .834 & .011 \\ .016 & .008 & .037 & .939 \end{bmatrix}$$

We note that the controller manages to decrease the number of activated sensors throughout the sensing period by activating the greedy policy, parallel to the learning process.

We now further demonstrate the effect of the learning parameter α on the convergence of the estimated transition matrix.

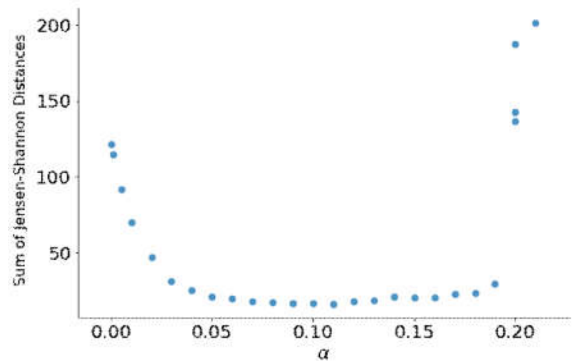

Fig. 2: Sum of Jensen-Shannon divergence throughout the simulations between the real transition matrix and the estimated transition matrix per α

As can be seen, for high enough values of α, the algorithm is unsuccessful at learning the transition matrix. This is due to the substantial changes in the estimated transition matrix at each epoch, which cause the controller to inaccurately estimate the patients' belief states. For lower values of α, the learning process is done rather slowly, causing higher distances.

An important issue when determining the simulation parameters is the number of patients that participate. Generally, the larger the amount of data collected, the more accurate the results. From this perspective, a large number of patients are desired. On the other hand, a large number of patients may be harder to manage, require extensive data storage, cause higher problem complexity, incur higher overhead costs, etc. Therefore, we would like to minimize the number of patients, while selecting enough patients to satisfy the required accuracy.

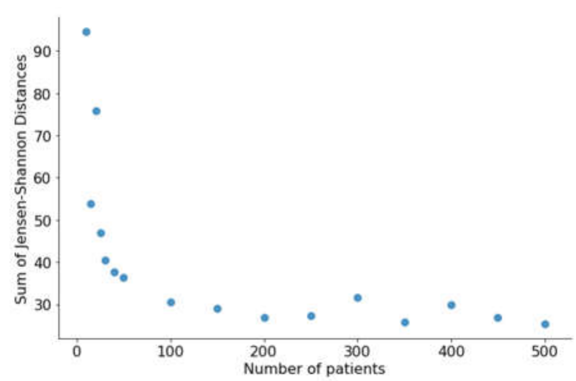

Fig. 3: Sum of Jensen-Shannon divergence throughout the sensing period between the real transition matrix and the estimated transition matrix w.r.t the number of participating patients

One can see that a lower number of patients results in larger distances throughout the learning process, meaning monitoring accuracy is lower. On the other hand, a high number of patients generally provides better accuracy. Interestingly, it seems a number of patients of between 100 and 200 patients, provides similar accuracy to of 300 patients or more. Of course, the optimal number of patients is influenced by the number of sensors, health states, possible sensor outputs and additional parameters which determine the complexity of the problem.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we have addressed the health sensing problem by using a POMDP model of a WBAN health monitoring system. The paper focuses on designing and implementing a learning algorithm, which can identify the underlying transition matrix of the POMDP, which is unknown to the controller. In parallel to the learning process, the controller activates a greedy sensor activation policy to optimize the trade-off between the two different types of costs: i) power cost, which accounts for the energy consumed by activating the sensors, ii) misclassification costs, which accounts for the probability of error when classifying a patient's true health state.

The algorithm relies on sensor data collected from a group of individuals, which share the unknown transition matrix. Each patient undergoes a health state trajectory, which is independent from other patients. The real health state trajectories evolve according to the unknown transition matrix, while the belief states evolve according to the transition matrix estimated by the controller. The main benefit for using a group of individuals is that it allows us to learn the transition matrix, without performing a preliminary off-line learning process. In addition, the sensor activation policy may be activated from the beginning of the sensing period, given enough data is collected continuously from the group of patients.

We have presented a basic numerical examination where the controller activated the greedy sensor activation policy in parallel to the transition matrix learning process. The results show that the controller successfully estimates the transition matrix, while activating the sensor activation policy, thus decreasing the power consumed by the sensors. An analysis of the learning rate parameter has shown that low values of α and high values of α limit the controllers ability to estimate the transition matrix. In addition, we have demonstrated how the number of patients may affect the accuracy of the transition matrix estimation. While a large number of patients is needed in order to successfully estimate the transition matrix, it is possible to decrease the number of patients, without necessarily reducing the learning capabilities, compared to higher values.

The model described in this paper serves as a simple proof of concept. Future work must include additional analysis to prove generalizability and robustness of the algorithm. This includes applying the algorithm to more complex problem instances, along with further analysis of the effect of the problem parameters, such as the sensor accuracies.

In addition, a number of extensions may be considered in order to improve the learning capability and real world applicability of the system. For example, applying more sophisticated sensor activation policies may improve the controller's ability to estimate the patients' belief states, without increasing the power consumption. This may improve the controller's ability to learn the transition matrix. In addition, using different distance functions for calculating the cost function J, may improve the estimation of the distance between the estimated transition matrix and the real transition matrix, thus improving the learning accuracy. An additional method to improve the learning capability is to use patient clustering, based on the patients' belief states evolution, in order to determine sub groups within the patient population, which may have differences in their transition matrices. One additional application that we plan to examine is applying the learning process for a single patient. This may be used to customize the monitoring process per patient, thus improving monitoring accuracy and power efficiency.

ACKNOWLEDGMENTS

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

REFERENCES

[1] Y. Bar David, T. Geller, A. Ward, D. Miller, E. Khmelnitsky, N. Bambos and I. Ben-Gal, (in press). "Optimal Health Monitoring via Wireless Body Area Networks," Conference on Decision and Control, 2018.

[2] 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.

[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. 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.

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

[6] Y. O. Mohammed and U.A Baroudi, "Partially observable Markov decision processes (POMDPS) and wireless body area networks (WBAN)", *KSII Transactions on Internet and Information Systems (TIIS)*, Vol. 7, No. 5, pp. 1036-1057, 2013.

[7] E. Jovanov, "A survey of power efficient technologies for Wireless Body Area Networks," *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Vancouver, BC, pp. 3628-3628, 2008.

[8] J. Xing and Y. Zhu, "A Survey on Body Area Network", *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*, Beijing, pp. 1-4, 2009.

[9] D. Tian and N. D. Georganas. "A coverage-preserving node scheduling scheme for large wireless sensor networks", *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 32-41, ACM, 2002.

[10] X. Fei, A. Boukerche and R. Yu, "A POMDP Based K-Coverage Dynamic Scheduling Protocol for Wireless Sensor Networks", *IEEE Global Telecommunications Conference GLOBECOM 2010*, Miami, FL, pp. 1-5, 2010.

[11] D. E. Spackman, S. Kadiyala, P. J. Neumann, D. L. Veenstra, and S. D. Sullivan, "Measuring Alzheimer Disease Progression with Transition Probabilities: Estimates from NACC-UDS," *Current Alzheimer Research*, vol. 9, no. 9, pp. 1050–1058, Jan. 2012.