Songklanakarin J. Sci. Technol.
45 (3), 379–385, May – Jun. 2023

http://www.sjst.psu.ac.th

Original Article

Production flow modeling based on BLE-based RSSI data with non-detectable areas

Karishma Agrawal* and Supachai Vorapojpisut

Department of Electrical and Computer, Faculty of Engineering,
Thammasat University, Rangsit Campus, Khlong Luang, Pathum Thani, 12120 Thailand

Received: 1 September 2022; Revised: 26 January 2023; Accepted: 21 April 2023

Abstract

This study presents a method for modelling manufacturing processes to predict key performance indicators (KPIs) such as cycle time using Bluetooth Low Energy (BLE) data. We consider BLE applications to be similar to Radio-Frequency IDentification (RFID) scenarios, with a single BLE scanner indicating a single working area. This work considers the case when Received Signal Strength Indicator (RSSI) data are unavailable in some areas, such as, when products are in temporary storage areas away from the production areas. We solve this problem with a Duration and Interval Hidden Markov Model (DI-HMM), in which time spent in production areas is represented as duration and those with absence data as intervals. To parameterize the DI-HMM model, we propose a two-stage machine-learning problem based on a classification tree and a Hidden Semi Markov Model (HSMM). To investigate the proposed model, the RSSI observation sequences are generated using MATLAB Bluetooth Toolbox and real-world experimentation. The runtime scenario compares estimated and original states, and the average accuracy of 100 test sequences is around 95%. In the offline forecast scenario, an estimated DI-HMM parameter is used to forecast 200 sequences, then compared with sequences with a vector distance with a similarity score of 0.4717.

Keywords: manufacturing process, bluetooth low energy, duration and interval hidden markov model, classification tree, hidden semi Markov model

1. Introduction

A factory is a combination of various buildings that host several manufacturing processes to produce final products to be stored and sent to customers. Generally, the factory area has two sections based on their operations: production areas and warehouse areas. The production areas are where a sequence of work is performed to manufacture final products, and the warehouse areas store these products until customers demand them. There are usually small storage areas within the production area to temporarily keep intermediate parts among the work areas. Product tracking data plays a crucial role in providing visibility of productivity in manufacturing areas, such as the amount of work-in-process, storage capacity, and machine status.

Production flow refers to how raw materials move in sequential order from one work area to another to get final products. Each work area will complete its specific task and move the finished parts to the next work area. If the next work area is occupied with some unfinished work, then the parts will be shifted to a temporary storage area. Several manufacturing KPIs are defined based on production flow information, including cycle time, machine utilization, and production throughput. The behaviour of production flow can be understood by tracking changes from raw material to parts and then products in production areas using barcode and RFID systems.

A Radio-Frequency IDentification (RFID) device is the most widely used device for tracking objects in malls, hospitals, factories, and other buildings (Zhu, Mukhopadhyay, & Kurata, 2012). An RFID system consists of two parts: an RFID reader and tags. RFID tags are attached to moving objects, while RFID readers are placed in specific locations to detect tag entries. The tags communicate via radio signal with readers as they pass by. The main drawback of RFID is that it

*Corresponding author
Email address: karishma.agra@dome.tu.ac.th

can only determine entry time, whereas knowing when products enter and leave a production flow is essential for predicting their temporal behaviour. Bluetooth Low-Energy (BLE) devices are widely used in indoor detection applications (Subedi & Pyun, 2020). In comparison to other solutions, BLE technology has several characteristics that satisfy the requirements of production tracking, such as a suitable range (10 m), a reasonable price tag (<$30), and a long battery life (>1 year). BLE object detection consists of two types of devices: BLE tags that broadcast their identities and BLE scanners installed at particular locations that look for nearby BLE tags. Proximity detection uses Received Signal Strength Indicator (RSSI) signals to identify nearby BLE devices and collect data such as RSSI, product ID, and timestamps for analyzing entry and exit times (Narzt *et al.*, 2016).

Numerous studies have proposed modelling of the temporal behaviours of production flow. Queuing theory is the most popular and traditional approach for estimating the temporal behaviours of production flow, such as cycle time and throughput time. The basic definition of queueing is that the production task arrives, waits for service, and then exits after finishing the process. However, these works (Gao *et al.*, 2019) usually consider the timing for each station separately. An RFID-enabled graphical deduction model (rfid-GDM) (Ding, Jiang, Sun, & Wang, 2017) was proposed to capture the time-sensitive and other aspects of RFID-tagged products in the production flow. This approach allows the decomposition of events from fixed and moveable RFID readers into a sequence of states. But the rfid-GDM concept does not aggregate RFID data from multiple production sequences into statistical parameters. This limits its application in the manufacturing process with respective operations. The unsupervised measurement (Nakai, Maekawa, & Namioka, 2016) studied wearable sensor data arranged as segments or **"**motifs**"** and figures out cycle time based on the motif's repetition intervals. However, the unsupervised measurement considers only the cycle time of one station, while neglecting the temporal characteristics of the whole production line. The use of BLE and HMM (Subedi & Pyun, 2020) was applied to define trajectories, and another work that combined BLE and HMM (Arslan, Cruz, & Ginhac, 2019) explained semantic trajectories to better understand worker mobility and improve safety. But these works ignore scenarios with the absence of BLE data due to the limited range of BLE communication. To overcome limitations related to the BLE detection range, we consider the HMM mathematical framework since the model can represent both the statistical behaviour of RSSI values as observations and the sequence of production flow as state transitions.

The main objective of this paper is to develop a mathematical model for predicting temporal metrics of the production flow using the HMM concept. We consider BLE applications similar to RFID scenarios where a single BLE scanner is installed in a single work area. Three core ideas make our study different from others:

1) Our scenario is motivated by the installation of barcode/RFID in production lines, so the BLE signal may be detected by one scanner of the occupied area, many (close-by areas), or none (no close area).
2) The conversion of RSSI values into the proximity of areas is to be aggregated as a stochastic model of sequence along the production line.
3) We use a two-stage learning problem to parameterize the model.

The paper is structured as follows. Section 2 reviews the concepts and technologies used. Section 3 defines the tracking problem as the DI-HMM model and the algorithm used to solve it. Section 4 contains the numerical results. The conclusion of the paper is discussed in Section 5.

2. Concepts and Technologies

2.1 BLE based radio frequency identification (RFID) technology

RFID has two components: reader and tag. There are two types of RFID tags: passive RFID and active RFID (Zhu, Mukhopadhyay, & Kurata, 2012). Mostly, passive RFID is used in industries. RFID tags obtain power from the reader, therefore the reader can receive a signal from that tag only when that tag reaches the reader. The RFID reader tends to be bulky in size and covers a short distance. BLE is a related technology that uses a similar concept to RFID but with a battery-powered tag. We use an RFID-like scenario where a single BLE scanner identifies a particular work area.

BLE is a low-power and short-range wireless technology that can be utilized for indoor object detection. RSSI reflects the distance between tags and scanning devices in dBm (Subedi & Pyun, 2020). Theoretically, distance and RSSI have an inverse relationship, but the RSSI fluctuates even when the location is the same. In our paper, we used the Wemos D1 R32 which is an ESP32 board with a PCB antenna. The initial study employed two boards, one as a BLE tag (broadcasting mode) and the other as a BLE scanner. By positioning the BLE tag at various distances and gathering RSSI data. Figure 1 shows the relationship between distance and RSSI.

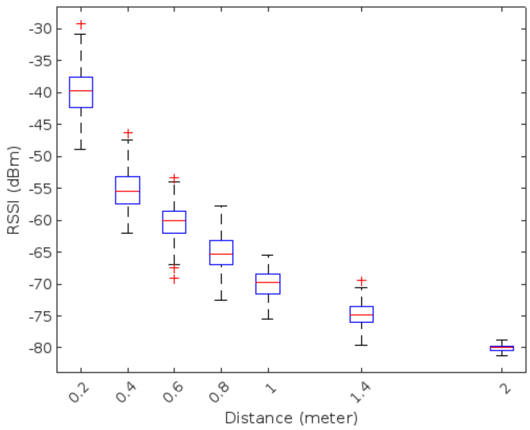

Figure 1. RSSI vs distance of BLE devices

Figure 1 depicts the RSSI variation by distance. Three fundamental issues with RSSI-based systems are signal strength attenuation, signal interference, and multipath propagation (Subedi & Pyun, 2020). Trilateration and fingerprinting (Jondhale *et al.*, 2022) are popular approaches that use RSSI values to detect the position of objects. Both approaches assume there are three or more BLE scanners that read RSSI values from the same BLE tag to find the object in 2D coordinates. However, BLE localization methods are ineffective in our study because they require at least two or more BLE scanners to find the coordinates and match the closest suitable area. Additionally, this will not work when no BLE data is available. That is why we propose a HMM-based method to compensate for such issues with the information of states in the production sequence.

2.2. Hidden Markov model (HMM)

HMM is a double-layer stochastic process with some specific hidden states based on the Markov process that relies upon observation states (Rabiner, 1989). Based on the HMM concept, the production sequence and transition in the manufacturing area can be determined from an observation sequence. HMM consists of 5-elements: State (S), Initial State probability (π), Transition Matrix (A), Observation State (O), and Emission Matrix (B). The following three conditions are required for the realization of HMM:

1) The Markov assumption: The HMM's next state depends just on the current state.
2) Independence assumption: The emission probability of the current observation depends only on the current state, consequently it is independent from other states and other observations.
3) The Stationarity assumption: State transition probabilities do not consider an actual time at which the transformation of the state occurs

Manufacturing is a complex process where products go through different work areas and operations, including picking, assembly, and forwarding. Cycle time is a critical KPI in the manufacturing industry because it describes temporal characteristics of work areas that can be used in the evaluation and forecasting of production efficiency. In the production process, production time is a fundamental aspect of the transition. To resolve this problem, the Hidden Semi Markov Model (HSMM) introduces the variable duration (D) that relies on each state and controls the transition matrix (A) (Sun, Wang, Moran & Rowe, 2020). Model accuracy increases if the system depends on time, in comparison with the plain HMM. HSMM has two main properties that make it different from HMM. First, the self-transition probability is assumed zero, i.e., no return to the same state is allowed, and a single state depends upon the sojourn duration (Sun, Wang, Moran, & Rowe, 2020).

3. Problem Statement

This study used a BLE-based system to track and collect product movement information to estimate the total cycle time of products. A BLE-based system uses BLE tags attached to the products, and BLE scanners that have been strategically placed in work areas to detect BLE tags. However, the coverage area of the BLE scanner in the work area is limited. As a result, when the product is between two work areas, no BLE scanner may be able to receive precise RSSI data, and the product location becomes unknown. Collecting product ID and RSSI data is the first step in understanding the production flow and its temporal metrics. Then, using the RSSI data sequence, the product's proximity location and production flow are estimated using the DI-HMM forward-backward algorithm. The other parameters, such as the overall cycle time and the cycle time for each area, are predicted using the learning algorithm.

Our previous study (Vorapojpisut & Agrawal, 2022) formulated the product tracking problem as an HSMM model $\Delta_{HSMM} = \{A, B, \pi, p\}$ that defines its state transition probability to depend on the sojourn time, which is the time for workers or machines to complete their tasks. HSMM Δ_{HSMM} consist of four mains of parameters (Sun, Wang, Moran & Rowe, 2020).

$$\Delta_{HSMM} = \{A, B, \pi, p\} \quad (1)$$

where A represents the transition matrix with duration probability p for each state, B signifies the emission matrix, and π is the initial state probability.

The HSMM model can represent duration time in its context, but cannot capture the transition period between work areas. The Duration and Interval Hidden Markov Model (DI-HMM) $\Delta_{DI-HMM} = \{A, B, \pi, p, L\}$ defines the periods without observations as state intervals (Narimatsu & Kasai, 2015). In HSMM, the next state S_j starts immediately after ending the present state S_i, but, the next state S_j in DI-HMM starts after a state interval that occurs after ending its previous state S_i. A state interval $I_{i,j}$ between two consecutive states where $i, j \in \{1,2,\ldots,N\}$ is expressed by interval length probability $P(L_{i,j})$ assumed to be the Gaussian distributed. A time length of observation of the DI-HMM is $\sum(d_i + I_{i,j})$, where $d_i \in D$ and $I_{i,i+1}$stands for the time difference between ending-time of S_i and starting-time of S_j.

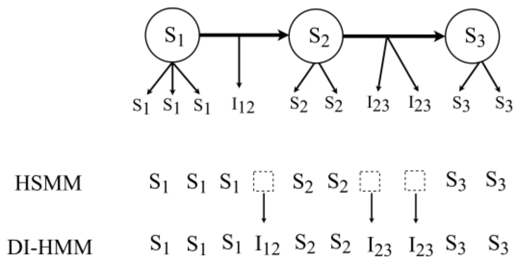

Figure 2. Sequence of DI-HMM

The following assumptions are made to reflect manufacturing process characteristics.

1) Each work area with a BLE scanner represents a state S_i, where $i = 1,2,3,..N$ and N denotes the number of work areas. In this case, M observed (RSSI) signals are generated based on the states, therefore $M = N$.
2) The transition matrix A is fixed to reflect that the manufacturing process of similar products is sequential and identical. Since all product manufacturing begins in the S_1 work area, the initial state probability for the S_1 working area is 1 and 0 for the others.
3) The emission probabilities B are assumed to be fixed, but may be updated periodically based on the collected RSSI data (Sun, Wang, Moran &

Rowe, 2020), as follows:

$$b_i(O_{t:t+d-1}) = \prod_{k=t}^{t+d-1} b_i(O_k) \tag{2}$$

The sequential likelihood of the HSMM denoted as $P(O|\Delta_{HSMM})$ can be determined through a forward-backward procedure. The forward-backward has two sub-divided variables: forward variable and backward variable. The HSMM forward variable (Sun, Wang, Moran & Rowe, 2020) for estimating likelihood is defined as follows:

$$\alpha_t(i) = P(\,O_{1:t},\, S_i\ ends\ at\ t \mid \Delta_{HSMM}) = \sum_{d=1}^{min(t,D)} \alpha_{t-d}^*(i)\ p_i(d) b_i(O_{t-d+1:t}) \tag{3}$$

$$\alpha_t^*(i) = P(O_{1:t}\,, S_i\ begins\ at\ t+1\,|\Delta_{HSMM}) = \sum_{i=1}^{N} \alpha_t(i)\ a_{j,i} \qquad t = 1,\ldots,T \quad i = 1,\ldots,N \tag{4}$$

where, $\alpha_t(i)$ is the joint probability of obtaining the partial observation sequence up to time *t* and ending in the state i at the time t, given the model Δ_{HSMM}; and $\alpha_t^*(i)$ is the joint probability of obtaining the observation sequence up to time *t;* and entry to state i begins at the next time $t+1$, given the model Δ_{HSMM}. The proximity likelihood of a location is calculated using the formula $\widehat{S_t} = arg \max_{1<i<N} \alpha_t(i)$. The HSMM backward variable (Sun, Wang, Moran & Rowe, 2020) is defined as follows:

$$\beta_t(i) = P(O_{t:T}|\, S_i\ begins\ at\ t\,, \Delta_{HSMM}) = \sum_{d=1}^{(T-t,D)} \beta_{t+d}^*(i)\ p_i(d) b_i(O_{t:t+d-1}) \tag{5}$$

$$\beta_t^*(i) = P(O_{t:T}\ |S_i\ ends\ at\ t-1, \Delta_{HSMM}) = \sum_{j=1}^{N} a_{i,j}\ \beta_t(j) \qquad t = T,\ldots,1 \quad i = 1,\ldots,N \tag{6}$$

where $\beta_t(i)$ is the probability of the partial observation sequence from epoch *t* to the end, given that the systems start in state i at the time t and with a model Δ_{HSMM}, and $\beta_t^*(i)$ is the probability of the partial observation sequence from time t to the end, given that the system leaves the state i at the previous time $t-1$ and with a model Δ_{HSMM}. Then, the duration probability $\hat{p}_i(d)$ matrix is re-estimated based on the forward and backward variables as (Sun, Wang, Moran & Rowe, 2020):

$$\hat{p}_i(d) = \frac{\sum_{t=1}^{T-d+1} \alpha_{t-1}^*(i)\, \mathrm{p}_i(d)\, b_i(O_{t:t+d-1}) \beta_{t+d}^*(i)}{\sum_{d=1}^{D} \sum_{t=1}^{T-d+1} \alpha_{t-1}^*(i)\, \mathrm{p}_i(d)\, b_i(O_{t:t+d-1}) \beta_{t+d}^*(i)} \tag{7}$$

The work area contains BLE scanners that collect RSSI data and determine the location of BLE tags based on strong RSSI values, while the movement and storage of products outside work areas will lead to weak and noisy RSSI values such that product location becomes unknown. To solve the problem of missing RSSI data for some tags, we propose a classification tree (Kang & Zadorozhny, 2016) to identify inside/outside scenarios based on BLE data, as shown in Figure 3.

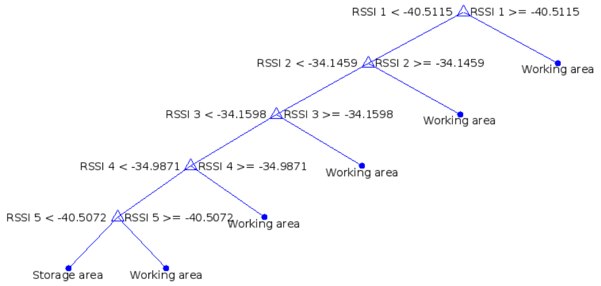

Figure 3. Classification tree to detect inside/outside scenarios

Input: $O_{1:T}^z = \{o_1^z, o_2^z, \ldots o_T^z\}$ where Z is number of training sequences

$ctree$ Classification tree model

Training phase:

$p_{sum} = 0$

$I_{sum} = 0$

for $z = 1$ to Z

Assign the HSMM parameters $\{\pi, A, B\}$

$z == 1$ assign p as uniformly distributed probability

for $t = 1$ to T

$label = predict(ctree\,,\, o_t)$

if $label == \{working\ area\}$

Calculate $\alpha_t\,, \alpha_t^*, \beta_t$, and β_t^* using (3), (4), (5), and (6)

else

Calculate interval I

end

end

Update parameter $\hat{p}$ using (7)

Calculate L using (8)

$p_{sum} = p_{sum} + \hat{p}$

$p_{avg} = {p_{sum}}/{Z}$

$p = p_{avg}$

end

Output: $\hat{p}$ and L

Algorithm 1. Re-estimate the duration and interval probabilities using DI-HMM

Algorithm 1 is to re-estimate the duration and interval probabilities using the classification tree and HSMM. When RSSI data are classified as a positive case (work area), the training algorithm of the HSMM model $\Delta_{HSMM} = \{A, B, \pi, p\}$ will proceed normally. When RSSI data are classified as a negative case (outside the work area), the training algorithm of HSMM model $\Delta_{HSMM} = \{A, B, \pi, p\}$ is paused, and the counting of the timing steps will be used for the modeling of interval probability $L_{i,j}$.

$$L_{i,j} = p(I_{i,j}) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(I_{i,j}-\mu)^2}{2\sigma^2}} \tag{8}$$

where $i = 1,2,.N-1$ and $j = 1,2,.N$.

4. Numerical Results

To evaluate our proposed algorithm, we conducted three numerical studies using data from simulated and real-world scenarios. A simulated experiment generates the RSSI sequence using the left-to-right HMM concept and the MATLAB Bluetooth toolbox. Real-world data are collected in a physical setup using four BLE scanners to represent the work areas and one storage area without a BLE scanner.

4.1 Simulated experiment

This study simulated end-to-end BLE transmission scenarios in the presence of a path loss model, RF impairments, and additive white Gaussian noise (AWGN). The MATLAB Bluetooth toolbox version R2022b was used to generate simulated data by providing the 2-D coordinates of the BLE scanners and moving positions for BLE tags. Algorithm 2 illustrates how to estimate the RSSI based on the scanner position with moving tags.

Input: Environment = Industrial, $P_t = 0\,dB$ Tx power

$Scanner_{pos} = [x_1\ y_1; x_2\ y_3; \ldots; x_N\ y_N]$,
where N is number of scanners

$Tag_{Motion} = [x_1\ y_1; x_2\ y_3; \ldots; x_L\ y_L]$,
where L is total path steps

$Tag_{at\ scanner} = [pos_1; pos_2; \ldots; pos_N]$,

$Interval = [I_{12min}\ I_{12max};\ I_{23min}\ I_{23max}; \ldots; I_{(N-1)\ N_{min}}\ I_{(N-1)\ N_{max}}]$

$Duration = [d_{1_{min}}\ d_{1_{max}}; d_{2_{min}}\ d_{2_{max}}; \ldots; d_{N_{min}}\ d_{N_{max}}]$

Sequence generation:

1. **for** $trajectory =$ 1 to L
2. Tag position = $Tag_{Motion}(trajectory, :)$
3. Nearest scanner = $find\ (Tag_{at\ scanner} == trajectory)$
4. **if** isempty *(*Nearest scanner*)* == 1
5. t = randomly select $Interval$ based on the two Nearest scanners
6. $S_{selected}$ = 'Storage area'
7. **else**
8. t = Select a random $Duration$ based on the Nearest scanner
9. $S_{selected}$ = Nearest scanner
10. **end**
11. S_{time} (1: t) = $S_{selected}$
12. $S = [S\ ,\ S_{time}]$
13. **for** $scan$ = 1 to N
14. Scanner position = $Scanner_{pos}(scan, :)$
15. Distance real = distance between the Tag position and Scanner position
16. **for** k = 1 to t
17. Distance = Distance real ± randomly generated minor error
18. PL_{dB} = Calculate path loss with End-to-End Bluetooth BR/EDR Simulation Procedure.
19. $RSSI = P_t - PL_{dB}$
20. $o(scan, k) = RSSI$
21. **end**
22. $O = [O\ ,\ o]$
23. **end**
24. **end**

Output: Hidden state S and RSSI observations O

Algorithm 2 Generates RSSI using MATLAB Bluetooth toolbox

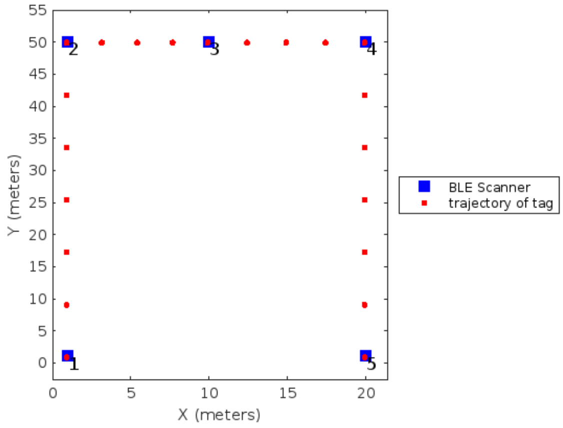

Figure 4. BLE scanner position and BLE tag trajectory.

500 different sequences were generated based on the setup of the BLE scanners and BLE tag trajectory shown in Figure 4 $Duration = [3{,}8;\ 5{,}6;\ 4{,}7;\ 2{,}4;\ 1{,}1]$ and $Interval = [3{,}15;\ 5{,}10;\ 5{,}8;\ 10{,}15]$ were used for algorithm 2. Each simulated sequence contains two types of information**:** observed (RSSI) signals and state sequences, as illustrated in Figure 5.

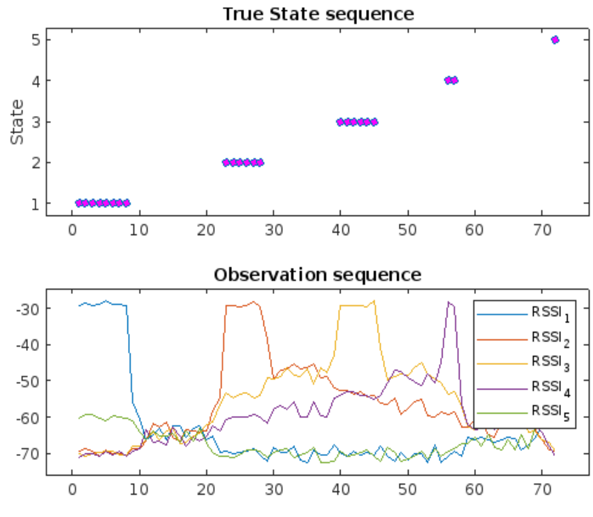

Figure 5. Example of simulated sequences (states and RSSI)

We conducted the runtime estimation and the offline forecast studies. The runtime estimation compares the state sequences and estimated DI-HMM sequence. In the offline forecast scenario, the re-estimated DI-HMM is used to forecast sequences, then average the forecast and state sequences using the sample count.

4.1.1 Runtime estimation scenario

In runtime estimation, 100 observed sequences were randomly selected to re-estimate the duration and interval probabilities using Algorithm 1. Then, the DI-HMM state sequences were estimated from RSSI values using the re-estimated duration and interval probabilities. Figure 6 shows the simulated state sequence as ground truth and the estimated DI-HMM state sequence.

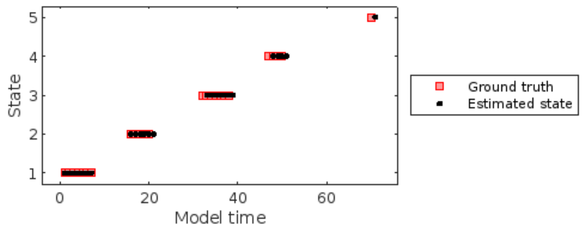

Figure 6. Comparison of the original and estimated states

DI-HMM model accuracy is evaluated using the confusion matrix, shown in Figure 7**.** The confusion matrix was used to evaluate performance, and classification $accuracy = correct\ predictions\ /\ total\ predictions * 100$.

The accuracy of this model was $(521 + 561 + 519 + 273 + 41 + 4569)/6759 = 0.9593$ or approximately 95%.

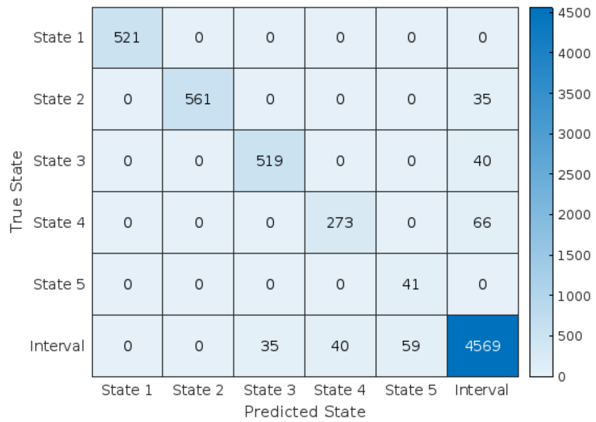

Figure 7. Confusion matrix

4.1.2 Offline forecast scenario

In the offline forecast, different state sequences are sampled to re-estimate the duration and interval probabilities with Algorithm 1. Then, the model is used to forecast another 200 state sequences. To compare the similarity of sequences, we encoded the simulated/forecast state sequence using run-length encoding (RLE). RLE is a simple algorithm for lossless data compression that works when a simulated/forecast state sequence contains the same value repeatedly. For example, if RLE's input is “AAABBCCCC,” its output is “3A2B4C”. After that, the vector distance is used to compare the multidimensional similarity of the forecast and simulated RLE vectors. The relationship between the training sample size and its vector distance is displayed in Figure 8. The average vector distance for the entire system was approximately 0.4717, which is close to zero and demonstrates that the estimated duration and interval are close to the original sequences used for the training of duration and interval parameters.

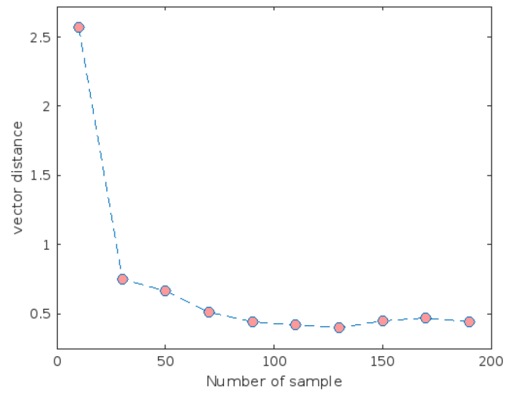

Figure 8. Vector distance for simulated and forecast sequences

4.2 Real-world experiment

In a real-world scenario, different stations carry out various tasks to collect RSSI data from BLE devices. The experiment consists of three components: Wemos R32 boards as BLE tags and scanners, Wi-Fi access points, and the Firebase database. Figure 2 demonstrates the BLE scanner's coverage area and the relationship between distance and RSSI. Four BLE scanners are arranged in a U shape over a 12×5m area, as shown in Figure 9.

Every 5 seconds as one timing unit, each BLE scanner scans, stores, and sends data to Firebase's real-time database. Each BLE tag was moved from one work location to the next in a predetermined order with 20 datasets collected from the area shown in Figure 9.

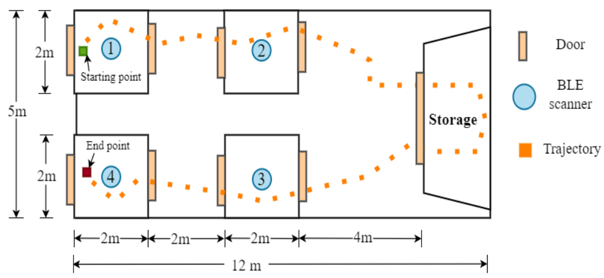

Figure 9. Scanner location in the experiment

Table 1 compares the average state duration and interval values between the data generated by our DI-HMM model and the data collected. This proves that our proposed method can achieve reasonable accuracy with realistic and limited datasets.

5. Conclusions

The main benefit of our approach is that it can uncover temporal behaviour and extract potential correlations in the DI-HMM that earlier research was unable to do, using

RSSI data from the production flow. With the help of a two-stage learning problem, it is possible to estimate the probability distributions $\hat{p}_i(d)$ and $L_{i,j}$ from RSSI data collected from experiments. After that, the DI-HMM model can be used to compute the crucial KPI metrics, i.e., cycle time and throughput time. The estimated cycle time of the products in the manufacturing area is the sum of the state's durations and interval times, represented as $\sum_{q=1}^{N}(d_q + I_{q-1,q})$. Throughput time is the ratio of time taken to complete production to a unit of the product, so the estimated cycle time of the product is equal to the throughput time. Work-in-Process defines the current state of the product in the manufacturing area. This study will benefit manufacturers by assisting them in estimating productivity on production flow. Moreover, we can forecast future target product behaviour based on the most recent forecasts.

References

Arslan, M., Cruz, C., & Ginhac, D. (2019). Semantic trajectory insights for worker safety in dynamic environments. *Automation in Construction*, *106*, 102854. Retrieved from https://doi.org/10.1016/j.autcon.2019.102854

Ding, K., Jiang, P., Sun, P., & Wang, C. (2017). RFID-enabled physical object tracking in process flow based on an enhanced graphical deduction modeling method. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *47*(11), 3006–3018. Retrieved from https://doi.org/10.1109/tsmc.2016.2558104

Gao, S., Rubrico, J. I. U., Higashi, T., Kobayashi, T., Taneda, K., & Ota, J. (2019). Efficient throughput analysis of production lines based on modular queues. *IEEE Access*, *7*, 95314–95326. Retrieved from https://doi.org/10.1109/access.2019.2928309

Jondhale, S. R., Mohan, V., Sharma, B. B., Lloret, J., & Athawale, S. V. (2022). Support vector regression for mobile target localization in indoor environments. *Sensors*, *22*(1), 358. Retrieved from https://doi.org/10.3390/s22010358

Kang, Y., & Zadorozhny, V. (2016). Process discovery using classification tree hidden semi-Markov model. *Proceedings of the 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*. Retrieved from https://doi.org/10.1109/iri.2016.55

Nakai, D., Maekawa, T., & Namioka, Y. (2016). Towards unsupervised measurement of assembly work cycle time by using wearable sensor. *Proceedings of the 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (Per Com Workshops)*. Retrieved from https://doi.org/10.1109/percomw.2016.7457056

Narimatsu, H., & Kasai, H. (2015). Duration and interval hidden Markov model for sequential data analysis. *Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN)*. Retrieved from https://doi.org/10.1109/ijcnn.2015.7280808

Narzt, W., Mayerhofer, S., Weichselbaum, O., Haselbock, S., & Hofler, N. (2016). Bluetooth Low Energy as enabling technology for Be-In/Be-Out systems. *Proceedings of the 2016 13th IEEE Annual Consumer Communications and Amp; Networking Conference (CCNC)*. Retrieved from https://doi.org/10.1109/ccnc.2016.7444817

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286. Retrieved from https://doi.org/10.1109/5.18626

Subedi, S., & Pyun, J. Y. (2020). A survey of smartphone-based indoor positioning system using RF-based wireless technologies. *Sensors*, *20*(24), 7230. Retrieved from https://doi.org/10.3390/s20247230

Sun, S., Wang, X., Moran, B., & Rowe, W. S. (2020). A hidden semi-Markov model for indoor radio source localization using received signal strength. *Signal Processing*, *166*, 107230. Retrieved from https://doi.org/10.1016/j.sigpro.2019.07.023

Vorapojpisut, S., & Agrawal, K. (2022). Modeling of manufacturing processes using hidden semi-Markov model and RSSI data. *Proceeding of the 2022 17th International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP)*. Retrieved from https://doi.org/10.1109/isai-nlp56921.2022.9960270

Zhu, X., Mukhopadhyay, S. K., & Kurata, H. (2012). A review of RFID technology and its managerial applications in different industries. *Journal of Engineering and Technology Management*, *29*(1), 152–167. Retrieved from https://doi.org/10.1016/j.jengtecman.2011.09.011