RIS-aided Smart Manufacturing: Information Transmission and Machine Health Monitoring

Tiep M. Hoang, Son Dinh-Van, Balbir Barn, Ramona Trestian and Huan X. Nguyen

***Abstract*—This paper proposes a novel industrial Internet-of-Things framework to monitor the machine health conditions (MHCs) in a smart factory. The framework utilises reconfigurable intelligent surface (RIS) to address propagation blockages while employing a novel power mapping scheme and an autoencoder to facilitate the transmission and classification of the MHCs. Analytical and numerical analyses are then performed to study the ergodic capacity (primary information) and the MHC accuracy (secondary information) in terms of the RIS size (K) and the transmit power (P). We observe that the accuracy of detecting MHCs does not change significantly with K and P, implying that the MHC alerts can be efficiently conveyed in parallel with the primary information. By contrast, a careful choice of different power mapping levels is necessary in order to achieve the two main goals: i) reasonably high data rate for primary transmission and ii) high accuracy for secondary MHC information.**

***Index terms*—**Industrial IoT, Industry 4.0, Digital Twin, Machine Health, Autoencoder, RIS, 5G/6G.

I. Introduction

It is expected that the manufacturing process will become more and more automated and productive due to the application of advances in machine learning and/or 5G-and-beyond techniques [1]–[4]. Towards Industry 4.0, a smart factory will need to be capable of data analysis and data-driven decision-making. Accordingly, the information of machine health conditions (MHCs) can be discovered and the predictive maintenance can be performed in timely fashion, which helps reduce the downtime and related cost [1], [5]. It is noted that Internet-of-Things (IoT) will be one pillar not only in Industry 4.0 (see [1], [6]) but also in 5G/6G networks (see [7], [8]) and the emerging Digital Twin technology (see [9]), because IoT devices can be easily connected to exchange information at large scale. According to [6], the use of IoT devices for the predictive maintenance in Industry 4.0-oriented factories remains an open issue.

T. M. Hoang is with the Department of Electrical Engineering, the University of Colorado Denver, Denver, CO 80204, United States (e-mail: minhtiep.hoang@ucdenver.edu).

S. Dinh-Van is with the School of Engineering, the University of Warwick, Coventry CV4 7AL, UK (e-mail: Son.V.Dinh@warwick.ac.uk).

B. Barn, R. Trestian and H. X. Nguyen are with the London Digital Twin Research Centre, Middlesex University, The Burroughs, London, NW4 4BT, UK (e-mails: {B.Barn, R.Trestian, H.Nguyen}@mdx.ac.uk).

H. X. Nguyen is also with School of Engineering and Technology, Van Lang University, Ho Chi Minh City, Vietnam (email: hx.nguyen@vlu.edu.vn).

This work was supported by a UK-India Education and Research Initiative (UKIERI) grant, ID DST 2018-19-11. The grant is funded by the UK Department for Business, Energy and Industrial Strategy and Department for Education and delivered by the British Council.

The work of Huan X. Nguyen is also supported by collaborative fund from Van Lang University.

Correspondence author: Huan X. Nguyen.

On the other hand, IoT devices are normally associated with the deployment of IoT-enabled 5G/6G networks, which benefits Industry 4.0 settings with increased data rate [10]–[12]. In fact, the rapid growth of IoT devices can boost the capacity of a wireless communication system [7], [8]. Additionally, 5G/6G networks are not limited to the use of IoT devices only, but they will require the integration of a diverse range of network entities to achieve the desired data rate. Among the recent advances in the field of wireless communications, reconfigurable intelligent surfaces (RISs) are emerging as new network entities that help connect remote devices, especially in an indoor environment with many blockages (e.g., walls, equipment and tables) [13]. An RIS contains multiple reconfigurable reflecting elements that can control the diffusion of an incident wave and create artificial communication channels between a pair of transceivers [14], [15]. Due to the novelty of RISs, their role in supporting Industry 4.0 remains an open question.

A. Related Works

Recently, machine learning has been used for fault detection and predictive maintenance [16]–[20]. The authors of [16] present an approach to machine monitoring and fault diagnosis/detection. Meanwhile, the authors of [17] use support vector machines and k-nearest neighbors for building multi-class classification models, which will then differ faulty situations from non-faulty ones in the manufacturing process. In [18], the so-called sparse filtering technique is used for learning features from motor/locomotive bearing datasets. While [16]–[18] do not discuss the role of IoT devices in building predictive maintenance models, [19] presents an application programming interface (API) oriented method to perform predictive maintenance in IoT-enabled smart factories. However, the API-oriented method in [19] does not give any insight into the physical-layer communication protocols of IoT devices. By contrast, we design an IoT communication protocol at the physical layer to adjust the transmit power of wireless signals, thereby being able to convey the secondary information about MHCs in parallel with the primary information. More noticeably, none of the works in [16]–[18] considers alert transmission in the industrial IoT environment. To address the alert reporting, [20] proposes software engines for IoT-based health monitoring, albeit for human, not machines. Moreover, the work in [20] does not analyze the propagation mechanism of wireless signals and might be dedicated to the application layer. Apparently, [16]–[20] have not yet investigated the alert reporting in smart factories, concerning the landscape of both

IoT-enabled Industry 4.0 and IoT-enabled 5G/6G networks, where IoT devices can be wirelessly connected for information exchange.

From a communication perspective, 5G/6G systems can employ RISs to connect IoT devices instead of using traditional relays [21]–[25]. Due to different types of blockages in the factory environment, the line-of-sight path is likely blocked and there may exist the non-line-of-sight (NLoS) paths only [24], [25]. Thus, it is promising to integrate RISs into the 5G/6G wireless systems of smart factories so that remote IoT devices can communicate more easily and efficiently. In [7], [8], the IoT-enabled 5G-and-beyond networks are surveyed. Meanwhile, [21] employs RISs as intermediate relays for improving the propagation environment with NLoS paths, thereby enhancing the data rate of IoT-enabled networks. In [22], the authors propose an RIS-aided IoT system with the ability to harvest energy from radio frequency signals. In [23], a blockage-aware model is built with the purpose of reducing the handover overhead in RIS-aided networks; however, the deployment of IoT devices is not discussed. Applying the theory of stochastic geometry, the work in [24] models the distribution of RISs, while the work in [25] models the distribution of blockages. Recall that the research topic of [21]–[25] is solely dedicated to the wireless communication aspect but does not relate to the diagnosis of machine health or the predictive maintenance.

Based on what has been discussed above, we can see that there are two major distinct-but-related topics that have not yet been fully investigated. These are the following:

- Topic 1: Health machine monitoring/diagnosis in IoT-enabled Industry 4.0.
- Topic 2: Alert transmission in RIS-aided IoT-enabled 5G/6G networks.

We jointly consider the two topics in this paper. Moreover, we exploit the power of machine learning for achieving the goal in Topic 1. In fact, different from [16]–[19] that only focus on machine health monitoring, we additionally consider the aspect of alert transmission via wirelessly connected IoT devices. Also, different from [21]–[25] that only focus on wireless propagation, we additionally investigate how machine health can be remotely monitored through controlling the transmission mechanism of IoT devices at the physical layer. In other words, we do not only perform the diagnosis of machine health through deep learning, but also study the wireless transmission of alert notifications through IoT devices. As mentioned earlier, the joint topic of Topic 1 and Topic 2 has not been investigated to the best of our knowledge. Thus, we are motivated to carry out this work to *bridge the gap* between these two distinct topics.

B. Main Contributions

Our main contributions can be summarized as follows:

- In parallel with the primary information that is normally transmitted from IoT devices to a data center, we utilize the power domain to transmit another type of information, i.e. the secondary information about the health condition of each and every machine in a production line. Through different transmit power levels that are controlled by power coefficients, we are able to monitor a wide range of MHC-related events. In order to classify these events, we train a deep autoencoder (AE), which is a special type of deep neural network, for learning the most significant characteristics of time series data. Based on the classification of different MHCs, a production supervisor will be able to appropriately respond to the current condition of a certain machine, such as an instant maintenance decision at the right time.
- In parallel with the secondary information about the MHC, we utilize RISs for supporting the primary information transmission between IoT devices and a data center, thereby overcoming the disadvantages of the indoor propagation environment, e.g., the lack of line-of-sight path due to undesired blockages. Moreover, we propose a simple-but-effective algorithm, which adjusts the RIS parameters, so that the capacity can approach its upper bound. Based on applying this RIS-control algorithm, we show that the increase in the RIS size can improve the capacity.
- From a trade-off perspective, we observe that when a machine is in bad condition, the transmit power shift at the associated IoT devices will allow us to perform an alert transmission about the MHC; however, the power reduction will also cause the primary information to be conveyed at a lower data rate. To evaluate the overall impact of the transmit power reduction, we quantify the channel capacity w.r.t the primary information and the accuracy w.r.t the MHC monitoring. It is shown that a careful choice of the power coefficients will make both the capacity and the accuracy reasonably high. On the other hand, the increase of the RIS size can only improve the capacity, but does not affect the classification accuracy of the AEs significantly. Thus, we can raise the transmit power and RIS size to attain a desired channel capacity regarding the primary information transmission, without affecting the classification accuracy of MHCs.

In short, this work implies the feasibility of achieving two goals at the same time at a smart factory setting: i) transmitting primary information at a high data rate and ii) conveying the MHC at a high accuracy.

Notations: $\mathbb{R}^{m\times n}$ denotes the real field that includes all real-valued matrices of size $m \times n$; $\mathbb{C}^{m\times n}$ denotes the complex field that includes all complex-valued matrices of size $m \times n$; The operation $\mathsf{diag}\left([z_1, \ldots, z_K]\right)$ diagonalizes a row vector $[z_1, \ldots, z_K]$ into a diagonal matrix; Bold lowercase letters and bold uppercase letters denote vectors and matrices, respectively; $\mathbf{I}_n$ denotes the identity matrix of size $n \times n$; The upperscripts $(\cdot)^\top$, $(\cdot)^*$, and $(\cdot)^\dagger$ represent the transpose, conjugate, and Hermitian operators, respectively; $\|\cdot\|^2$ denotes the Euclidean norm; $\mathbf{z} \sim \mathcal{CN}(\mathbf{0}, \mathbf{\Sigma})$ denotes a complex-valued Gaussian random vector with the zero-mean and the covariance matrix $\mathbf{\Sigma}$.

The rest of this paper is organized as follows: Section II introduces a manufacturing process of multiple stages. Section III presents an RIS-aided IoT-enabled network and explains the mechanism of transmitting two different types of information.

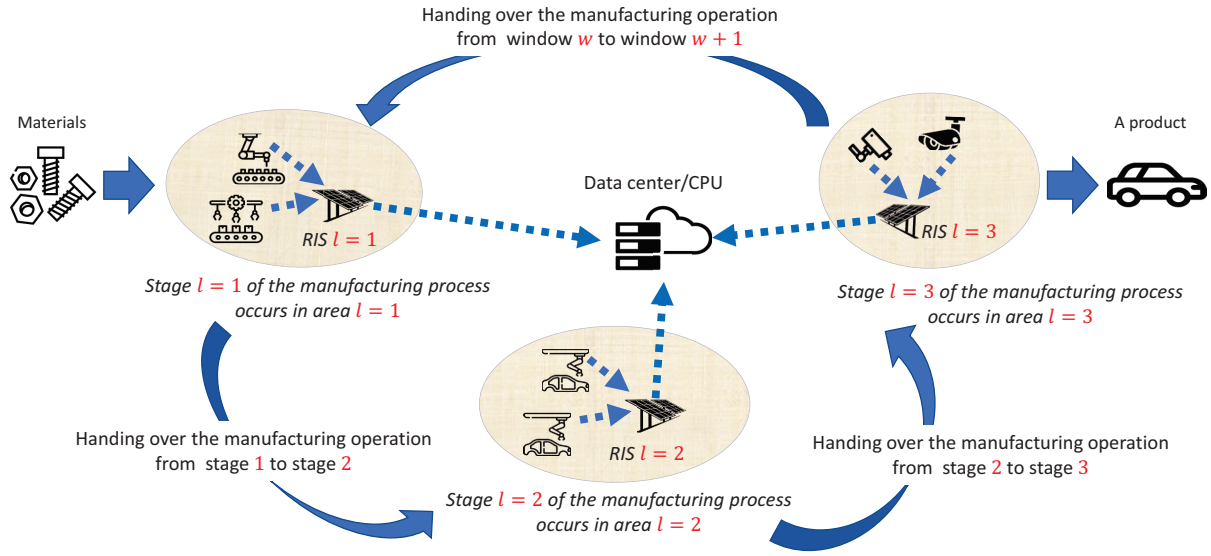


Fig. 1: An IoT network where IoT devices from different in-factory areas connect wirelessly to a CPU via RISs. The factory is divided into L areas, each performing a specific stage of the *cyclic*-manufacturing process. In each area, there are M_ℓ IoT devices and an RIS. The role of the ℓ-th RIS is to support M_ℓ IoT devices inside the ℓ-th area to connect to the CPU. To obtain a finished product, the factory will perform a process of L stages. After finishing a product, the manufacturing process will repeatedly carry out a process of L stages to make a new product.

Section IV analyzes the achievable capacity of the proposed network. In Section V, we first present the formulation of faulty and non-faulty situations associated with the MHC, and then apply autoencoders for autonomously classifying machine health statuses. In Section VI, the achievable capacity and the detection accuracy are validated, respectively. Finally, Section VII concludes the paper.

II. Production Line Model

We consider an IoT network that consists of a central processing unit (CPU) connected to L in-factory areas via L reconfigurable intelligent surfaces (RISs). Each area is supposed to have only one RIS, i.e., the ℓ-th RIS is placed inside the ℓ-th area. We also assume that inside the ℓ-th area, there are M_ℓ IoT devices. These IoT devices wirelessly connect to the CPU through the ℓ-th RIS to transmit primary information. Moreover, the ℓ-th RIS in the ℓ-th area will not connect with IoT devices in another area $\ell' \neq \ell$.

Let us divide the transmission into W equal (time) *windows*. Within each time window, there are L equal (time) *frames*, and the ℓ-th frame is composed of T_ℓ equal (time) *slots*. Thus, the length of a window is equal to $\sum_{\ell=1}^{L} T_\ell \triangleq T_{\text{tot}}$ (slots), which is a constant. Herein, we assume that a production line makes a product within a time window, and the production line consists of L consecutive stages. As such, the division of a window into L frames implies that each stage of the production line is carried out within a time frame.

Additionally, given that there are L in-factory areas, we can arrange the factory so that the ℓ-th stage of the manufacturing process occurs in the ℓ-th area of the factory. Thus, we will simply use "the ℓ-th frame" as the shorthand for "the ℓ-th stage of the manufacturing process" or " the manufacturing process in the ℓ-th area". For simplicity, let us assume that there are 3 stages (and or 3 in-factory areas), i.e. $L = 3$. For example, the roles of these stages can be summarized as follows:

- Stage $\ell = 1$ is to prepare materials.
- Stage $\ell = 2$ is to process the materials with technologies.
- Stage $\ell = 3$ is to inspect product quality & pack products.

Furthermore, we suppose that during the ℓ-th time frame, with $\ell \in \mathcal{L} \triangleq \{1, 2, 3\}$, only the ℓ-th area is permitted to transmit its signals to the data center, where the CPU is put in place. Thus, during the ℓ-th frame, any signal received at the data center is supposed to come from the ℓ-th area. Finally, local IoT devices are not capable of transmitting their signals beyond the area to which they belong. Figure 1 depicts our manufacturing process performed through 3 stages, each happening in a different area.

Denote $\mathcal{I} \triangleq \{1, 2, \ldots, T_{\text{tot}}\}$ as the set of time *indices* of the w-th window. A certain element $i \in \mathcal{I}$ will imply the i-th index of the w-th window.[1] Depending on the value of ℓ, we will have different ranges for the index i as described below:

- $\ell = 1$ (i.e., Stage 1) for $i \leq T_1$;
- $\ell = 2$ (i.e., Stage 2) for $T_1 < i \leq T_1 + T_2$;
- $\ell = 3$ (i.e., Stage 3) for $T_1 + T_2 < i \leq T_1 + T_2 + T_3$.

As such, given an index i, we can specify the value of ℓ and t based on the following relationship:

$$\text{index } i = \underbrace{T_1 + T_2 + \ldots + T_{\ell-1}}_{\text{total duration of previous frames}} + \text{slot } t \text{ of frame } \ell, \quad (1)$$

where $t \in \{1, \ldots, T_\ell\}$ indicates the position of a slot in the considered frame ℓ. Please see the illustration of the time

[1]For example, if we have $T_1 = 10, T_2 = 20$ and $T_3 = 30$, then the 15-th element in $\mathcal{I}$, i.e. the index $i = 15$, implies the 5-th slot of the 2-th frame of the w-th window, because $T_1 < t = 15 < T_1 + T_2$. For notational simplicity, we will mention the 15-th index of the w-th window without needing to mention frames.

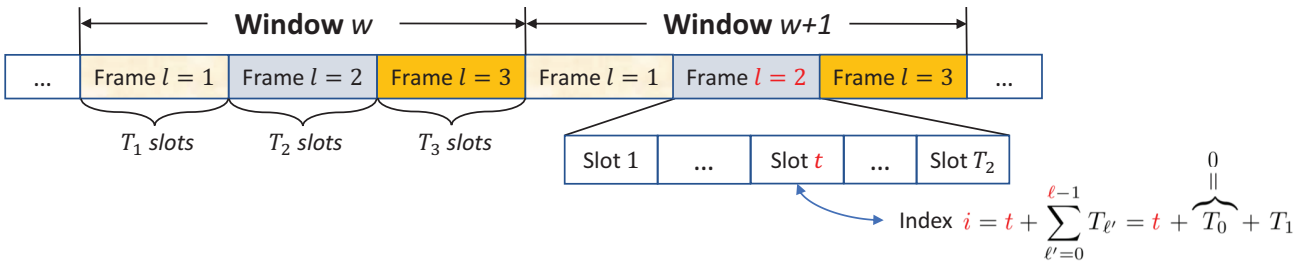


Fig. 2: A manufacturing cycle is the time needed to convert materials into products. A manufacturing cycle corresponds to a window, and each window involves $L = 3$ stages: i) prepare materials, ii) process the materials, and iii) inspect the product quality and pack products. The ℓ-th stage corresponds to the ℓ-th frame. Moreover, the ℓ-th frame has T_ℓ slots. Thus each manufacturing cycle has $T_{\text{tot}} = T_1 + \ldots + T_L$ slots. The relationship (1) between slot t and index i is also illustrated.

division in Figure 2. Moreover, for notational clarity, let us define the set of time indices

$$\mathcal{I}_\ell = \left\{ 1 + \sum_{\ell'=0}^{\ell-1} T_{\ell'}, 2 + \sum_{\ell'=0}^{\ell-1} T_{\ell'}, \ldots, T_\ell + \sum_{\ell'=0}^{\ell-1} T_{\ell'} \right\},$$

where $T_{\ell'=0} = 0$. We observe that the ℓ-th frame has T_ℓ slots, and the t-th slot of the ℓ-th frame is equivalent to the index $i = t + \sum_{\ell'=0}^{\ell-1} T_{\ell'}$. Hence, from now on, the ℓ-th frame also implies all time indices in the set $\mathcal{I}_\ell$, i.e.

$$\text{the } \ell\text{-th frame} \Leftrightarrow \forall i \in \mathcal{I}_\ell.$$

Note that $\mathcal{I}_1, \mathcal{I}_2 \ldots, \mathcal{I}_L$ are disjoint sets, yet they form the universal set $\mathcal{I} = \bigcup_{\ell=1}^{L} \mathcal{I}_\ell$.

In the rest of this paper, the upper-script $\{\cdot\}^{[i,w]}$ will be used to stand for "*the i-th index of the w-th window*". As just aforementioned, the index i enables us to indicate the t-th slot of the ℓ-th frame. Additionally, the lower-script $\{\cdot\}_{m\to\ell}$ implies the link between the m-th IoT device of the ℓ-th area and the associated RIS (i.e., the ℓ-th RIS). Likewise, the lower-script $\{\cdot\}_{\ell\to C}$ implies the link between the ℓ-th RIS and the CPU.

III. Information Transmission Model

In this section, we first present how wireless signals are transmitted in a conventional manner in Sub-section III-A, and then present how to employ the power domain for additionally performing an alert transmission in Sub-section III-B. Noticeably, that the conventional information transmission model in Sub-section III-A relates to the case of perfect production line, while our proposed model of transmitting two different types of information relates to a more practical case of imperfect production line.

A. Perfect Production Line: No Alert Transmission

The main purpose of the IoT devices is to send primary information to the data center, e.g., reporting the manufacturing process to the data center regularly so that the center can capture the information about the product quality. The reported information might enable us to know whether the manufactured goods satisfy a pre-determined requirement or not. Under the ideal assumption that the production line works perfectly, we do not need to monitor the health condition of machines in the production line. In this case, the transmitted signals will be designed so that they contain *only one type of information*, e.g. the primary information about the product quality.

Denote $s_m^{[i,w]}$ as the signal transmitted by the m-th IoT device.[2] Denote $r_{\text{IDEAL}}^{[i,w]}$ as the aggregated signal received at the CPU. We have

$$r_{\text{IDEAL}}^{[i,w]} = r_1^{[i,w]} + r_2^{[i,w]} + \ldots + r_M^{[i,w]} + n^{[i,w]} \tag{2}$$

where $r_m^{[i,w]}$ is contributed by the transmitted signal $s_m^{[i,w]}$, and $n^{[i,w]} \in \mathbb{C}^{1\times 1}$ is the additional white Gaussian noise (AWGN). As for $s_m^{[i,w]}$, we assume $\mathbb{E}\left\{s_m^{[i,w]}\right\} = 0$ and $\mathbb{E}\left\{|s_m^{[i,w]}|^2\right\} = \mathbb{E}\left\{|s_{m|\text{ideal}}^{[i,w]}|^2\right\} = P$. As for the AWGN, we assume $\mathbb{E}\left\{n^{[i,w]}\right\} = 0$ and $\mathbb{E}\left\{|n^{[i,w]}|^2\right\} = N_0$ is the noise variance. Concerning an element $r_m^{[i,w]}$ in (2), we have

$$r_m^{[i,w]} = \sqrt{10^{-\frac{\varrho_{m\ell}}{10}}}\ \mathbf{g}_{\ell\to C}^{[i,w]}\ \mathbf{\Phi}_\ell\ \mathbf{h}_{m\to\ell}^{[i,w]}\ s_m^{[i,w]},\ \text{for } \forall i \in \mathcal{I}, \tag{3}$$

where $\varrho_{m\ell}$ is the effective path loss between the m-th IoT device inside the ℓ-th area and the CPU; $\mathbf{h}_{m\to\ell}^{[i,w]} \in \mathbb{C}^{K_\ell\times 1}$ is the channel from the m-th IoT device inside the ℓ-th area to the ℓ-th RIS; $\mathbf{\Phi}_\ell \in \mathbb{C}^{K_\ell\times K_\ell}$ is the phase shift matrix of the ℓ-th RIS; and $\mathbf{g}_{\ell\to C}^{[i,w]} \in \mathbb{C}^{1\times K_\ell}$ is the channel from the ℓ-th RIS to the CPU. A manufacturing cycle is associated with a time window. Thus, each manufacturing cycle starts with the 1-st stage and ends with the L-th stage.[3] Please see Figure 1 for illustration.

Substituting (3) into (2), we can rewrite (2) as follows:

$$r_{\text{IDEAL}}^{[i,w]} = \sum_{m=1}^{M} \chi_{m\ell}^{[i,w]}\ s_m^{[i,w]} + n^{[i,w]}, \text{for } i \in \mathcal{I}, \tag{4}$$

where

$$\chi_{m\ell}^{[i,w]} \triangleq \sqrt{10^{-\frac{\varrho_{m\ell}}{10}}}\ \mathbf{g}_{\ell\to C}^{[i,w]} \mathbf{\Phi}_\ell \mathbf{h}_{m\to\ell}^{[i,w]} \tag{5}$$

is a random variable. Note that $\mathbf{\Phi}_\ell = \text{diag}\left(\boldsymbol{\phi}_\ell\right)$ is the diagonal matrix of the ℓ-th RIS, where $\boldsymbol{\phi}_\ell = [\theta_{\ell,1}, \theta_{\ell,2}, \ldots, \theta_{\ell,K_\ell}]$ is a row vector that contains K_ℓ complex-valued coefficients and satisfies $\|\boldsymbol{\phi}_\ell\|^2 = \sum_{k=1}^{K} |\theta_{\ell,k}|^2 = K$.

[2]Recall that the upper-script $\{\cdot\}^{[i,w]}$ has been already defined, thus $s_m^{[i,w]}$ can be implicitly interpreted as the signal transmitted by the m-th IoT device *inside* area ℓ during *the t-th slot of the ℓ-th frame of the w-th window*. Herein, ℓ and t can be easily inferred from (1).

[3]A manufacturing cycle is referred to as the time interval needed to convert materials into finished products. In our case, the length of a manufacturing cycle is equal to the number of slots, i.e. $\sum_{\ell=1}^{L} T_\ell$.

B. Imperfect Production Line: With Alert Transmission

Different from Subsection III-A, we will consider a more practical scenario in which the components of a machine are not always in good condition and will lead to some potential risks in the future. Thus, it is also important to report the MHC information (i.e., the secondary information).

If each stage of the manufacturing process is carried out by a distinct machine, then there will be L distinct machines. Each machine is placed inside an area to run a stage of the manufacturing process. This means that the ℓ-th machine inside the ℓ-th area is used for processing the ℓ-th stage of the manufacturing process. Now, considering the ℓ-th machine's health condition at the i-th index of the w-th window, we denote $(G_\ell^{[i,w]})$ as the event that the machine is in good condition. Similarly, denote $(B_\ell^{[i,w]})$ as the event that the machine is in bad condition. Denote $(B_{\ell|\text{AMBER}}^{[i,w]})$ as the *conditional* event that the machine is still able to work properly given that MHC is in bad condition. And finally, denote $(B_{\ell|\text{RED}}^{[i,w]})$ as the *conditional* event that the machine requires immediate repair/maintenance given that the MHC is in bad condition. Mathematically speaking, we have

$$(B_\ell^{[i,w]}) = (B_{\ell|\text{AMBER}}^{[i,w]}) \cup (B_{\ell|\text{RED}}^{[i,w]}) \tag{6}$$

Obviously, both $(B_{\ell|\text{AMBER}}^{[i,w]})$ and $(B_{\ell|\text{RED}}^{[i,w]})$ reflect the degradation status of the ℓ-th machine. However, we differ $(B_{\ell|\text{AMBER}}^{[i,w]})$ from $(B_{\ell|\text{RED}}^{[i,w]})$ based on the severity. In the case of $(B_{\ell|\text{AMBER}}^{[i,w]})$, the ℓ-th machine is assumed to become overloaded after running a while, but it does not affect the product quality significantly and thus it is not necessarily replaced to save equipment cost. In the case of $(B_{\ell|\text{RED}}^{[i,w]})$, that machine will need to be repaired to guarantee a seamless manufacturing operation.

In order to transmit two different types of information, we will employ the power domain for additionally transmitting the information about the MHC. Our transmission goals are as follows:

- i) The transmitted $s_m^{[i,w]}$ signal itself conveys the messages about the primary information.
- ii) In the power domain, the transmit power $\mathbb{E}\left\{|s_m^{[i,w]}|^2\right\} = \mathbb{E}\left\{|s_{m|\text{drop}}^{[i,w]}|^2\right\}$ conveys the messages about the MHC. Different levels of the transmit power will enable us to know if a machine is in good or bad condition. Furthermore, in the bad condition, we can realize how bad the machine is.

To be more specific, the average power of $s_m^{[i,w]}$ in Subsection III-A has only one power level, i.e. $\mathbb{E}\left\{|s_{m|\text{ideal}}^{[i,w]}|^2\right\} = P$. By contrast, the average power of $s_m^{[i,w]}$ in this subsection will have the following three levels:

$$\mathbb{E}\left\{|s_{m|\text{drop}}^{[i,w]}|^2\right\} = \begin{cases} P, & \text{if } (G_\ell^{[i,w]}) \text{ occurs;} \\ \alpha_1 P, & \text{if } (B_{\ell|\text{AMBER}}^{[i,w]}) \text{ occurs;} \\ \alpha_2 P, & \text{if } (B_{\ell|\text{RED}}^{[i,w]}) \text{ occurs,} \end{cases} \tag{7}$$

where α_1 and α_2 are scaling factors satisfying the condition:

$$\text{either } 0 < \alpha_2 < \alpha_1 < 1, \tag{8a}$$

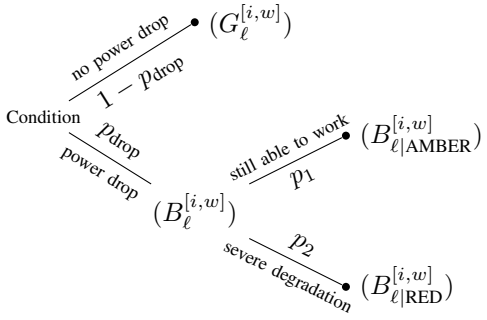
$$\text{or } 1 < \alpha_1 < \alpha_2. \tag{8b}$$

For simplicity, we will only consider one of the two cases and select the first case, corresponding to (8a), in this paper. As seen from (7), a transmission with three levels of power will enable the data center to know about the MHC and perform condition-based maintenance.

The occurrence of transitioning the transmit power from P to $\alpha_1 P$ (or $\alpha_2 P$) will be called a *power drop* event, i.e.

$$\text{power drop} \Leftrightarrow (B_\ell^{[i,w]}).$$

Furthermore, provided that a power drop event occurs at certain time index i, then $(B_{\ell|\text{AMBER}}^{[i,w]})$ occurs with the *conditional* probability of p_1, while $(B_{\ell|\text{RED}}^{[i,w]})$ occurs with the *conditional* probability of $p_2 = 1 - p_1$. The following probability tree-diagram summarizes all the related events:



Note that (7) presents the transmit power of an IoT device. Taking into account all M_ℓ IoT devices inside the ℓ-th area, we set up the following rules:

- If the event $(G_\ell^{[i,w]})$ occurs, then all IoT devices of the ℓ-th area will transmit with the average power of P.
- If the event $(B_{\ell|\text{AMBER}}^{[i,w]})$ occurs, all IoT devices of the ℓ-th area will transmit with the average power of $\alpha_1 P$.
- If the event $(B_{\ell|\text{RED}}^{[i,w]})$ occurs, all IoT devices of the ℓ-th area will transmit with the average power of $\alpha_2 P$.

By contrast, if the ℓ-th machine only works properly during the first T_ℓ^{G} slots and degrades during the last $(T_\ell - T_\ell^{\text{G}})$ slots, then the M_ℓ-th IoT devices will transmit with the average power of $\alpha_1 P$ or $\alpha_2 P$, depending on the severity of the machine degradation. Note that $1 \leq T_\ell^{\text{G}} \leq T_\ell$. Apparently, we do not know when a power drop event occurs, thus we will consider T_ℓ^{G} as a uniformly distributed random variable. Moreover, once the power drop event has occurred at the T_ℓ^{G}-th slot, we will need to divide the ℓ-th frame into 2 smaller durations, each corresponding to a different transmit power. For simplicity, we will consider 2 subsets of $\mathcal{I}_\ell$, they are the following:

$$\mathcal{I}_\ell^{\text{G}} = \left\{1 + \sum_{\ell'=0}^{\ell-1} T_{\ell'}, \ldots, T_\ell^{\text{G}} + \sum_{\ell'=0}^{\ell-1} T_{\ell'}\right\}, \tag{9}$$

$$\mathcal{I}_\ell^{\text{B}} = \mathcal{I}_\ell \setminus \mathcal{I}_\ell^{\text{G}} = \left\{T_\ell^{\text{G}} + 1 + \sum_{\ell'=0}^{\ell-1} T_{\ell'}, \ldots, T_\ell + \sum_{\ell'=0}^{\ell-1} T_{\ell'}\right\}. \tag{10}$$

Example 1. *To illustrate how to employ the power domain for transmitting the information about the MHC, we provide Figure 3 with specific parameters. Therein, we assume that the number of slots in each frame is the same, i.e. $T_\ell = 8$ for $\forall \ell \geq$*

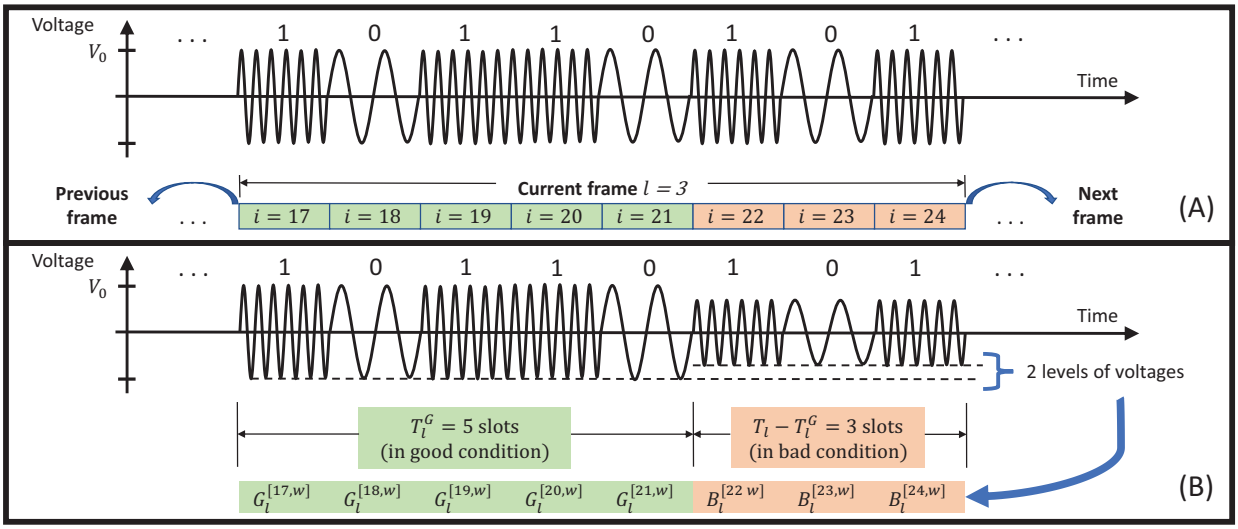


Fig. 3: An illustration of two scenarios. The detailed explanation of this figure is presented in Example 1.

1. *Herein, we set* $\ell = 3$. *The set of indices associated with the* 3*-rd frame is* $\mathcal{I}_3 = \left\{1 + \sum_{\ell'=0}^{3-1} T_{\ell'}, \ldots, T_3 + \sum_{\ell'=0}^{3-1} T_{\ell'}\right\} = \{17, 18, \ldots, 24\}$. *The sub-figure (A) illustrates a bitstream of* 10110101 *to be conveyed during* $T_\ell = 8$ *slots of a certain frame* ℓ. *Noticeably, in the sub-figure (A), the bitstream is the only type of information to be conveyed. By contrast, the sub-figure (B) illustrates that two types of information can be conveyed at the same time by changing the signal voltage to notify us of the health condition of the* 3*-rd machine. During the first* T_ℓ^G *slots, the* 3*-rd machine works well. However, during the last* $(T_\ell - T_\ell^G)$ *remaining slots, this machine's health condition changes and it then alerts the data center to the new situation. To notify us of the MHC,* M_3 *IoT devices inside the* 3*-rd area will lower their voltage and the new voltage level depends on the severity. Consequently, apart from the bitstream of* 10110101*, the* 3*-rd machine's health condition is also informed. In the sub-figure (B), we set* $T_\ell^G = 5$ *and have the sequence of the events* $(G_\ell^{[17,w]})$, $(G_\ell^{[18,w]})$, $(G_\ell^{[19,w]})$, $(G_\ell^{[20,w]})$, $(G_\ell^{[21,w]})$, $(B_\ell^{[22,w]})$, $(B_\ell^{[23,w]})$, $(B_\ell^{[24,w]})$ *transmitted. Herein,* $(B_\ell^{[\cdot,w]})$ *can be either* $(B_{\ell|AMBER}^{[\cdot,w]})$ *or* $(B_{\ell|RED}^{[\cdot,w]})$, *depending on the severity of the machine degradation.*

Based on what has been discussed, we are now able to formulate the received signal at any time index i of the w-th time window as follows:

$$r_{\text{practical}}^{[i,w]} = \begin{cases} r_{\text{IDEAL}}^{[i,w]}, & i \in \mathcal{I}^{\text{G}}; \\ r_{\text{AMBER}}^{[i,w]} \text{ or } r_{\text{RED}}^{[i,w]}, & i \in \mathcal{I}^{\text{B}}, \end{cases} \tag{11}$$

where

$$r_{\text{AMBER}}^{[i,w]} = \sum_{m=1}^{M} \chi_{m\ell}^{[i,w]}\ s_m^{[i,w]} + n^{[i,w]}, \tag{12}$$

$$\text{with } \mathbb{E}\left\{|s_{m|\text{drop}}^{[i,w]}|^2\right\} = \alpha_1 P;$$

$$r_{\text{RED}}^{[i,w]} = \sum_{m=1}^{M} \chi_{m\ell}^{[i,w]}\ s_m^{[i,w]} + n^{[i,w]}, \tag{13}$$

$$\text{with } \mathbb{E}\left\{|s_{m|\text{drop}}^{[i,w]}|^2\right\} = \alpha_2 P,$$

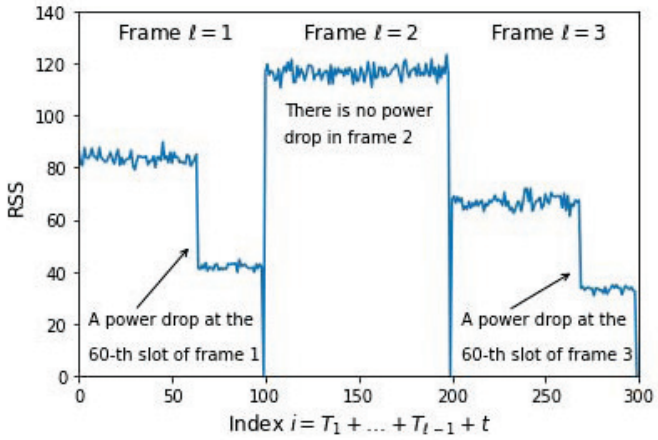


Fig. 4: Received signal strength (RSS), i.e. $\left|r_{\text{practical}}^{[i,w]}\right|^2$, versus the time index i. A manufacturing cycle (i.e., a time window) consists of 3 time frames. Each frame has 100 time slots. The 1-st machine, which is located in the 1-st area, changes to alerting mode from the T_1^{G}-th slot onwards. The 2-nd machine still works properly. Finally, the 3-rd machine, which is located in the 3-rd area, enters alerting mode from the T_3^{G}-th slot onwards. Herein, we set $T_1^{\text{G}} = T_3^{\text{G}} = 60$.

and

$$\mathcal{I}^{\text{G}} = \mathcal{I}_1^{\text{G}} \cup \mathcal{I}_2^{\text{G}} \cup \ldots \cup \mathcal{I}_L^{\text{G}}, \tag{14}$$

$$\mathcal{I}^{\text{B}} = \mathcal{I} \setminus \mathcal{I}^{\text{G}} = \mathcal{I}_1^{\text{B}} \cup \mathcal{I}_2^{\text{B}} \cup \ldots \cup \mathcal{I}_L^{\text{B}} \tag{15}$$

Furthermore, our transmission scheme will be designed so that if a power drop event occurs at the $\left(T_\ell^{\text{G}} + 1\right)$-th slot with the probability of p_{drop}, then the new level of power will remain unchanged until the last slot of the ℓ-th frame. For intuitive understanding, please refer to Figure 4.

IV. Achievable Capacity

Given that a power drop event occurs (with the probability of p_{drop}) at the slot $\left(T_\ell^{\text{G}} + 1\right)$-th slot of the ℓ-th frame, we can

TABLE I: A summary of critical symbols

Symbols	Meanings
$(G_\ell^{[i,w]})$	the ℓ-th machine is in *good* condition at the i-th index of the w-th window
$(B_\ell^{[i,w]})$	the ℓ-th machine is in *bad* condition at the i-th index of the w-th window
$(B_{\ell\|\text{AMBER}}^{[i,w]})$	$(B_\ell^{[i,w]})$ occurs and the ℓ-th machine is still able to work
$(B_{\ell\|\text{RED}}^{[i,w]})$	$(B_\ell^{[i,w]})$ occurs but the ℓ-th machine cannot work properly
$\mathcal{I}_\ell^{\text{G}}$	the set of time indices at which the ℓ-th machine is still in good condition
$\mathcal{I}_\ell^{\text{B}}$	the set of time indices at which the ℓ-th machine is in bad condition
K	the number of RIS elements
L	the number of stages in the manufacturing process
T_ℓ	the number of time slots dedicated to the ℓ-th stage
M_ℓ	the number of IoT devices inside the ℓ-th in-factory area
P	when the event $(G_\ell^{[i,w]})$ occurs, the average transmit power of each IoT device is P
α_1	when the event $(B_{\ell\|\text{AMBER}}^{[i,w]})$ occurs, the average transmit power of each IoT device is $\alpha_1 P$
α_2	when the event $(B_{\ell\|\text{RED}}^{[i,w]})$ occurs, the average transmit power of each IoT device is $\alpha_2 P$

rely on (11)–(13) to calculate the SNR at the t-th slot of the ℓ-th frame as follows:

$$\text{SNR}_\ell^{[i,w]} = \frac{\sum_{m=1}^{M} |\chi_{m\ell}^{[i,w]}|^2 \ \mathbb{E}\left\{|s_{m|\text{drop}}^{[i,w]}|^2\right\}}{\mathbb{E}\left\{|n^{[i,w]}|^2\right\}} \tag{16}$$

Depending on the value of the index i, we will obtain a specific value of $\text{SNR}_{\ell|\text{drop}}^{[i,w]}$. To be more specific,

- For $i \in \mathcal{I}_\ell^{\text{G}}$, we have

$$\text{SNR}_\ell^{[i,w]} = (P/N_0)\gamma_\ell^{[i,w]}, \text{with the prob. of } 1, \tag{17}$$

  where $\gamma_\ell^{[i,w]} = \sum_{m=1}^{M_\ell} |\chi_{m\ell}^{[i,w]}|^2$.
- For $i \in \mathcal{I}_\ell^{\text{B}}$, we have

$$\text{SNR}_\ell^{[i,w]} = \begin{cases} \frac{P}{N_0}\gamma_\ell^{[i,w]}, \text{with the prob. of } (1-p_{\text{drop}}); \\ \frac{\alpha_1 P}{N_0}\gamma_\ell^{[i,w]}, \text{with the prob. of } (p_{\text{drop}}p_1); \\ \frac{\alpha_2 P}{N_0}\gamma_\ell^{[i,w]}, \text{with the prob. of } (p_{\text{drop}}p_2). \end{cases} \tag{18}$$

As a result, given the realizations of channels, $i \in \mathcal{I}_\ell$, the instantaneous capacity of the link from the ℓ-th area to the CPU will be formulated as $C_\ell^{[i,w]} = \log_2\left(1 + \text{SNR}_\ell^{[i,w]}\right)$. Obviously, the specific value of $C_\ell^{[i,w]}$ depends on the value of the index i. As a result, we have

- For $i \in \mathcal{I}_\ell^{\text{G}}$:

$$C_\ell^{[i,w]} = \log_2\left(1 + (P/N_0)\gamma_\ell^{[i,w]}\right). \tag{19}$$

- For $i \in \mathcal{I}_\ell^{\text{B}}$:

$$C_\ell^{[i,w]} = \begin{cases} \log_2\left(1 + (P/N_0)\gamma_\ell^{[i,w]}\right), \\ \quad \text{with the prob. of } (1-p_{\text{drop}}); \\ \log_2\left(1 + (\alpha_1 P/N_0)\gamma_\ell^{[i,w]}\right), \\ \quad \text{with the prob. of } (p_{\text{drop}}p_1); \\ \log_2\left(1 + (\alpha_2 P/N_0)\gamma_\ell^{[i,w]}\right), \\ \quad \text{with the prob. of } (p_{\text{drop}}p_2). \end{cases} \tag{20}$$

The ergodic capacity of the link from the ℓ-th area to the CPU can be calculated as follows:

- For $i \in \mathcal{I}_\ell^{\text{G}}$:

$$C_\ell^{[i]} = \mathbb{E}\left\{\log_2\left(1 + (P/N_0)\gamma_\ell^{[i,w]}\right)\right\}. \tag{21}$$

- For $i \in \mathcal{I}_\ell^{\text{B}}$:

$$\begin{aligned} C_\ell^{[i]} = &(1-p_{\text{drop}})\,\mathbb{E}\left\{\log_2\left(1 + (P/N_0)\gamma_\ell^{[i,w]}\right)\right\} \\ &+ (p_{\text{drop}}p_1)\,\mathbb{E}\left\{\log_2\left(1 + (\alpha_1 P/N_0)\gamma_\ell^{[i,w]}\right)\right\} \\ &+ (p_{\text{drop}}p_2)\,\mathbb{E}\left\{\log_2\left(1 + (\alpha_2 P/N_0)\gamma_\ell^{[i,w]}\right)\right\}. \end{aligned} \tag{22}$$

Proposition 1. *Given that $\chi_{m\ell}^{[i,w]}$ is defined as in (5) and $\gamma_\ell^{[i,w]}$ is defined as $\gamma_\ell^{[i,w]} = \sum_{m=1}^{M_\ell} |\chi_{m\ell}^{[i,w]}|^2$, the ergodic quantity $\mathbb{E}\left\{\log_2\left(1 + a\gamma_\ell^{[i,w]}\right)\right\}$ is upper-bounded by $\log_2\left(1 + aK^2\sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}}\right)$, where a is a constant.*

Proof. See Appendix A. □

Denote $C_{\ell|\text{upper}}^{[i]}$ as the upper bound for $C_\ell^{[i]}$ at the index $i \in \mathcal{I}_\ell$. Note that $C_\ell^{[i]} \le C_{\ell|\text{upper}}^{[i]}$. Using Proposition 1, we can formulate $C_{\ell|\text{upper}}^{[i]}$ as follows:

- For $i \in \mathcal{I}_\ell^{\text{G}}$, we have

$$C_{\ell|\text{upper}}^{[i]} = \log_2(1 + \Omega_\ell), \tag{23}$$

  where $\Omega_\ell = (P/N_0)K^2\sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}}$.
- For $i \in \mathcal{I}_\ell^{\text{B}}$, we have

$$\begin{aligned} C_{\ell|\text{upper}}^{[i]} = &(1-p_{\text{drop}})\log_2(1+\Omega_\ell) \\ &+ p_{\text{drop}}\log_2\left[(1+\alpha_1\Omega_\ell)^{p_1}(1+\alpha_2\Omega_\ell)^{p_2}\right]. \end{aligned} \tag{24}$$

Adding up $C_\ell^{[i]}$ over all $|\mathcal{I}| = \sum_{\ell=1}^{L}|\mathcal{I}_\ell|$ indices, we obtain the total ergodic capacity of a window as follows:

$$\begin{aligned} C^{\text{tot}} = \sum_{\ell=1}^{L}\Bigg[&\sum_{i\in\mathcal{I}_\ell^{\text{G}}} \mathbb{E}\left\{\log_2\left(1 + \frac{P}{N_0}\gamma_\ell^{[i,w]}\right)\right\} \\ &+ (1-p_{\text{drop}})\sum_{i\in\mathcal{I}_\ell^{\text{B}}} \mathbb{E}\left\{\log_2\left(1 + \frac{P}{N_0}\gamma_\ell^{[i,w]}\right)\right\} \end{aligned}$$

$$+ p_{\text{drop}} p_1 \sum_{i \in \mathcal{I}_\ell^{\text{B}}} \mathbb{E}\left\{\log_2\left(1 + \frac{\alpha_1 P}{N_0}\gamma_\ell^{[i,w]}\right)\right\} + p_{\text{drop}} p_2 \sum_{i \in \mathcal{I}_\ell^{\text{B}}} \mathbb{E}\left\{\log_2\left(1 + \frac{\alpha_2 P}{N_0}\gamma_\ell^{[i,w]}\right)\right\}\Bigg]. \tag{25}$$

Denote $C_{\text{upper}}^{\text{tot}}$ as the upper bound for C^{tot}. Using Proposition 1, we can formulate $C_{\text{upper}}^{\text{tot}}$ as

$$C_{\text{upper}}^{\text{tot}} = \sum_{\ell=1}^{L}\Bigg[\sum_{i \in \mathcal{I}_\ell^{\text{G}}} \log_2\left(1 + \Omega_\ell\right) + (1 - p_{\text{drop}}) \sum_{i \in \mathcal{I}_\ell^{\text{B}}} \log_2\left(1 + \Omega_\ell\right) + p_{\text{drop}} \sum_{i \in \mathcal{I}_\ell^{\text{B}}} \log_2\left[\left(1 + \alpha_1\Omega_\ell\right)^{p_1}\left(1 + \alpha_2\Omega_\ell\right)^{p_2}\right]\Bigg]. \tag{26}$$

For the sake of readability, a summary of critical symbols is presented in Table I.

V. Machine Health Monitoring

A. Time-Series Data and Labels

Based on the received signal $r^{[i,w]}$, we will develop a classification strategy using machine learning techniques. We consider $r^{[i,w]}$ as the raw data that will be extracted and transformed into a certain useful attribute for the training/learning purposes.

Regarding the i-th index of the w-th window, let us define $a^{[i,w]} \triangleq |r^{[i,w]}|/M$. After a product is made, i.e. after a time window passes by, we obtain the following *time series*:

$$\mathbf{a}_{\text{raw}}^{[w]} = \left[a^{[1,w]}, a^{[2,w]}, \ldots, a^{[T_{\text{tot}},w]}\right]. \tag{27}$$

Herein, the time series $\mathbf{a}_{\text{raw}}^{[w]}$ is presented in the form of a row vector. Based on $\mathbf{a}_{\text{raw}}^{[w]}$, we can decode the secondary information about machine health during all L phases. The process of decoding the secondary information is actually the classification into one of multiple classes, each indicating the overall health status of all L machines. The classification process will be performed by an AE later in this section. In order to facilitate this process, we normalize $\mathbf{a}_{\text{raw}}^{[w]}$ to the range $(0, 1)$ by simply multiplying a scaling factor $1/\max\left(a^{[1,w]}, \ldots, a^{[T_{\text{tot}},w]}\right)$. Mathematically, we have $\mathbf{a}^{[w]} = \frac{1}{\max\left(a^{[1,w]},\ldots,a^{[T_{\text{tot}},w]}\right)}\mathbf{a}_{\text{raw}}^{[w]}$. Using the normalized time series $\mathbf{a}^{[w]}$ as the data, we can train the AE so that it can classify $\mathbf{a}^{[w]}$ into one of the predetermined classes, thereby showing the health status of the ℓ-th machine. For example, if the class of $\mathbf{a}^{[w]}$ shows that the ℓ-th machine's health condition is in good condition, then the shape of the time series $\mathbf{a}^{[w]}$ indicates that there is no "drop" during $i \in \mathcal{I}_\ell$. By contrast, if the class of $\mathbf{a}^{[w]}$ shows that the ℓ-th machine's health condition is in bad condition, then there is a drop at a certain index $i \in \mathcal{I}_\ell$. Recall that the occurrence of a drop at the i-th time index is associated with either $(G_\ell^{[i,w]})$ or $(B_\ell^{[i,w]})$, depending on the severity of the machine damage.

Since $(G_\ell^{[i,w]})$ and $(B_\ell^{[i,w]})$ are the events that are associated with the i-th index of the w-th window, thus they can only enable us to know about the MHC at the index i. Due to the fact that the ℓ-th frame has T_ℓ slots, corresponding to the index i from $(T_1 + \ldots + T_{\ell-1}) + 1$ to $(T_1 + \ldots + T_{\ell-1}) + T_\ell$, it is necessary to consider an event capturing the MHC over all T_ℓ slots of the ℓ-th frame. We thus define

$$\mathbf{0}_\ell^{[w]} = \bigcap_{i=T_1+\ldots+T_{\ell-1}+1}^{T_1+\ldots+T_{\ell-1}+T_\ell} (G_\ell^{[i,w]}) = \bigcap_{i \in \mathcal{I}_\ell} (G_\ell^{[i,w]}) \tag{28}$$

as the event that the ℓ-th machine is in good condition throughout the ℓ-th frame. On the other hand, if there is a power drop at the $(T_\ell^{\text{G}} + 1)$-st slot of the ℓ-th frame, then we will denote

$$\mathbf{1}_\ell^{[w]} = \left(\bigcap_{i \in \mathcal{I}_\ell^{\text{G}}} (G_\ell^{[i,w]})\right) \bigcap \left(\bigcap_{i \in \mathcal{I}_\ell^{\text{B}}} (B_{\ell|\text{AMBER}}^{[i,w]})\right) \tag{29}$$

$$\text{and} \quad \mathbf{2}_\ell^{[w]} = \left(\bigcap_{i \in \mathcal{I}_\ell^{\text{G}}} (G_\ell^{[i,w]})\right) \bigcap \left(\bigcap_{i \in \mathcal{I}_\ell^{\text{B}}} (B_{\ell|\text{RED}}^{[i,w]})\right) \tag{30}$$

as the event that the ℓ-th machine is in good condition during the first T_ℓ^{G} slots, i.e. $i \in \mathcal{I}_\ell^{\text{G}}$, and in bad condition during the remaining $(T_\ell - T_\ell^{\text{G}})$ slots, i.e. $i \in \mathcal{I}_\ell^{\text{B}}$. Note that the occurrence $\mathbf{1}_\ell^{[w]}$ or $\mathbf{2}_\ell^{[w]}$ depends on the severity of the ℓ-th machine. For notational simplicity, we will drop the upper script $(\cdot)^{[w]}$ and write $\mathbf{0}_\ell$, $\mathbf{1}_\ell$ and $\mathbf{2}_\ell$ hereafter.

Since there are L stages in a manufacturing cycle, we will need to consider the combination of L events occurring in L time frames. Denote $\mathbf{z}_\ell \in \{\mathbf{0}_\ell, \mathbf{1}_\ell, \mathbf{2}_\ell\}$. Then, we can define the event

$$\langle \mathbf{z}_1, \ldots, \mathbf{z}_L \rangle = \mathbf{z}_1 \cap \mathbf{z}_2 \cap \ldots \cap \mathbf{z}_L \tag{31}$$

and rely on it to assess the health condition of all L machines. For example, if we have $\langle \mathbf{0}_1, \mathbf{2}_2, \mathbf{1}_3 \rangle$, then we can conclude that the 1-st machine works well, the 2-nd machine needs to be repaired, and the 3-rd machine is not in good condition but an urgent maintenance is not necessary.

From (31), we can see that there are multiple possibilities related to the health condition of L machines. Each possibility can be referred to as a combination of L events $\mathbf{z}_1^{[w]}, \ldots, \mathbf{z}_L^{[w]}$. Table II illustrates 27 different possibilities of $\langle \mathbf{z}_1, \ldots, \mathbf{z}_L \rangle$ with $L = 3$. Generally, we can easily calculate the number of possibilities of $\langle \mathbf{z}_1, \ldots, \mathbf{z}_L \rangle$ as follows:

$$V = \binom{L}{0}2^0 + \binom{L}{1}2^1 + \ldots + \binom{L}{L}2^L = 3^L, \tag{32}$$

where the second equality follows the binomial formula. With $V = 3^L$ possibilities of $\langle \mathbf{z}_1, \ldots, \mathbf{z}_L \rangle$, we will also have 3^L different labels/classes. By training an AE, we will be able to automatically classify time series into 3^L different classes.

B. Machine Health Monitoring based on Autoencoder

When a time series $\mathbf{a}^{[w]}$ is fed to the input of the AE, we will obtain the following output:

$$\widehat{\mathbf{a}}^{[w]} = \left[\widehat{a}^{[1,w]}, \widehat{a}^{[2,w]}, \ldots, \widehat{a}^{[T_{\text{tot}},w]}\right].$$

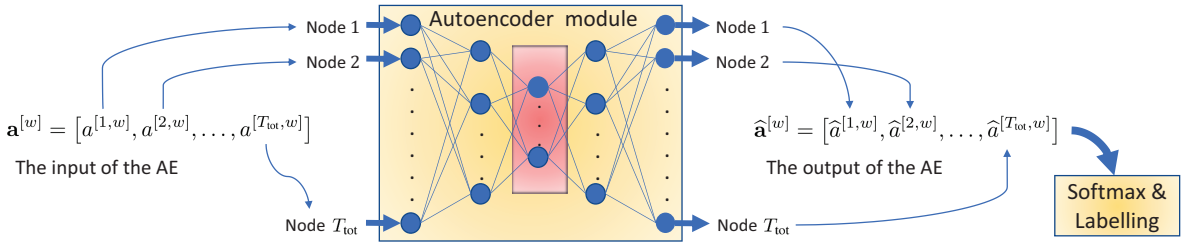


Fig. 5: The classification method that combines the AE architecture and softmax.

TABLE II: The description

The number of ways to choose k out of $L = 3$ positions so that these positions contain $\widehat{\mathbf{z}}_\ell \in \{\mathbf{1}_\ell, \mathbf{2}_\ell\}$.	After choosing k out of L positions, $\langle \mathbf{z}_1, \ldots, \mathbf{z}_L \rangle$ becomes the following:	The number of cases when $\widehat{\mathbf{z}}_\ell$ is replaced by either $\mathbf{1}_\ell$ or $\mathbf{2}_\ell$.	Possibilities	Classes
With $k = 0$, we have $\binom{L}{0}$ ways	$\langle \mathbf{0}_1, \mathbf{0}_2, \mathbf{0}_3 \rangle$	2^0 case	$\langle \mathbf{0}_1, \mathbf{0}_2, \mathbf{0}_3 \rangle$	1
With $k = 1$, we have $\binom{L}{1}$ ways	$\langle \mathbf{0}_1, \mathbf{0}_2, \widehat{\mathbf{z}}_3 \rangle$	2^1 cases	$\langle \mathbf{0}_1, \mathbf{0}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{0}_1, \mathbf{0}_2, \mathbf{2}_3 \rangle$	2, 3
	$\langle \mathbf{0}_1, \widehat{\mathbf{z}}_2, \mathbf{0}_3 \rangle$	2^1 cases	$\langle \mathbf{0}_1, \mathbf{1}_2, \mathbf{0}_3 \rangle$, $\langle \mathbf{0}_1, \mathbf{2}_2, \mathbf{0}_3 \rangle$	4, 5
	$\langle \widehat{\mathbf{z}}_1, \mathbf{0}_2, \mathbf{0}_3 \rangle$	2^1 cases	$\langle \mathbf{1}_1, \mathbf{0}_2, \mathbf{0}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{0}_2, \mathbf{0}_3 \rangle$	6, 7
With $k = 2$, we have $\binom{L}{2}$ ways	$\langle \mathbf{0}_1, \widehat{\mathbf{z}}_2, \widehat{\mathbf{z}}_3 \rangle$	2^2 cases	$\langle \mathbf{0}_1, \mathbf{1}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{0}_1, \mathbf{1}_2, \mathbf{2}_3 \rangle$, $\langle \mathbf{0}_1, \mathbf{2}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{0}_1, \mathbf{2}_2, \mathbf{2}_3 \rangle$,	8, 9, 10, 11
	$\langle \widehat{\mathbf{z}}_1, \widehat{\mathbf{z}}_2, \mathbf{0}_3 \rangle$	2^2 cases	$\langle \mathbf{1}_1, \mathbf{1}_2, \mathbf{0}_3 \rangle$, $\langle \mathbf{1}_1, \mathbf{2}_2, \mathbf{0}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{1}_2, \mathbf{0}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{2}_2, \mathbf{0}_3 \rangle$	12, 13, 14, 15
	$\langle \widehat{\mathbf{z}}_1, \mathbf{0}_2, \widehat{\mathbf{z}}_3 \rangle$	2^2 cases	$\langle \mathbf{1}_1, \mathbf{0}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{1}_1, \mathbf{0}_2, \mathbf{2}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{0}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{0}_2, \mathbf{2}_3 \rangle$	16, 17, 18, 19
With $k = 3$, we have $\binom{L}{3}$ ways	$\langle \widehat{\mathbf{z}}_1, \widehat{\mathbf{z}}_2, \widehat{\mathbf{z}}_3 \rangle$	2^3 cases	$\langle \mathbf{1}_1, \mathbf{1}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{1}_1, \mathbf{1}_2, \mathbf{2}_3 \rangle$, $\langle \mathbf{1}_1, \mathbf{2}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{1}_1, \mathbf{2}_2, \mathbf{2}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{1}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{1}_2, \mathbf{2}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{2}_2, \mathbf{1}_3 \rangle$, $\langle \mathbf{2}_1, \mathbf{2}_2, \mathbf{2}_3 \rangle$	20, 21, 22, 23, 24, 25, 26, 27

Applying the softmax function to the output of the AE, we can classify the output time series $\widehat{\mathbf{a}}^{[w]}$ into one of the $V = 3^L = 3^3 = 27$ classes. To indicate that the class of the input time series $\mathbf{a}^{[w]}$ and the class of the output time series $\widehat{\mathbf{a}}^{[w]}$ belong to a certain class c, we introduce the lower script $\{\cdot\}_c$ to the notations $\mathbf{a}^{[w]}$ and $\widehat{\mathbf{a}}^{[w]}$, respectively. Thus, $\mathbf{a}_c^{[w]}$ implies that the input time series $\mathbf{a}^{[w]}$ belongs to the c-th class, $c \in \{1, \ldots, 27\}$. Also, $\widehat{\mathbf{a}}_c^{[w]}$ implies that the output time series $\widehat{\mathbf{a}}^{[w]}$ belongs to the c-th class.

When it comes to the training process of the AE, the loss function to be minimized can be defined as follows:

$$f_{\text{loss}} = (1/WV) \sum_{w=1}^{W} \sum_{c=1}^{V} \left\| \widehat{\mathbf{a}}_c^{[w]} - \mathbf{a}_c^{[w]} \right\|^2. \tag{33}$$

The execution of the AE aims to minimize the loss function f_{loss} so that the AE is capable of learning the representations of the data. This goal is attained through training and the weights of the AE will be optimized. After training, the middle layer of the AE, at which the data is compressed into a low-dimensional latent space, can represent the most significant features of the data. For notational simplicity, we denote ω as the number of neurons at the middle layer of the AE. In general, the higher the value of ω, the higher the complexity of the AE. Since the AE extracts the most characteristics of data, ω should be chosen so that it is less than T_{tot}. Thus, when the input time series $\mathbf{a}^{[w]}$ traverses from the first layer to the middle layer, we obtain the dimensionality reduction of $(T_{\text{tot}} - \omega)$. The impact of ω will be experimentally evaluated in the numerical section. Figure 5 depicts our classification algorithm that is made up of the AE and the softmax layer.

TABLE III: The Confusion Matrix

		Predicted classes			
		Class 1	Class 2	...	Class 27
Actual classes	Class 1	*True*	*False*	...	*False*
	Class 2	*False*	*True*	...	*False*
	⋮	⋮	⋮	⋱	⋮
	Class 27	*False*	*False*	...	*True*

Meanwhile, Table III illustrates the confusion matrix on which the accuracy calculation is based. To be more specific, the accuracy is the ratio of the sum of *True* events along the diagonal to the total *True* and *False* events in the confusion matrix.

VI. Numerical Results

In this section, we evaluate the ergodic capacity associated with the primary information and the detection accuracy of machine health associated with the secondary information. Unless otherwise specified, the system parameters are: $L = 3$, $M_1 = M_2 = M_3 = 5$, $K_1 = K_2 = K_3 = 20$, $T_0 = 60$, $\alpha_1 = 0.7$ and $\alpha_2 = 0.4$. By setting $T_1 = T_2 = T_3 = 100$, the number of neurons in the first/last layer is equal to $T_{\text{tot}} = T_1 + T_2 + T_3 = 300$. The number of training samples is equal to $3^L \times 10^3 = 27 \times 10^3$.

A. Channel Capacity

Figure 6 shows the ergodic capacity $C_\ell^{[i]}$ at different values of the index $i \in \mathcal{I}$. When an incident occurs at the $(T_\ell^{\text{G}} + 1)$-st

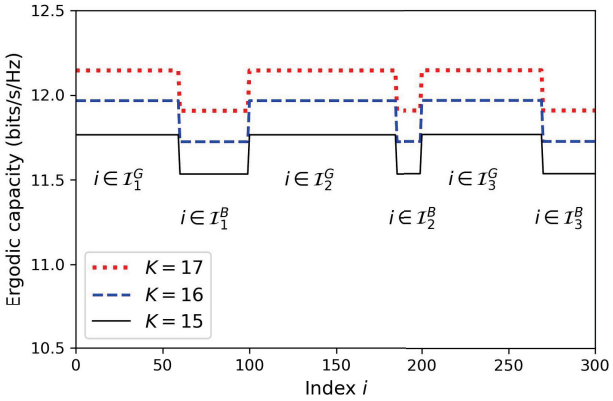


Fig. 6: The ergodic capacity $C_\ell^{[i]}$ versus i is depicted for three different values of K.

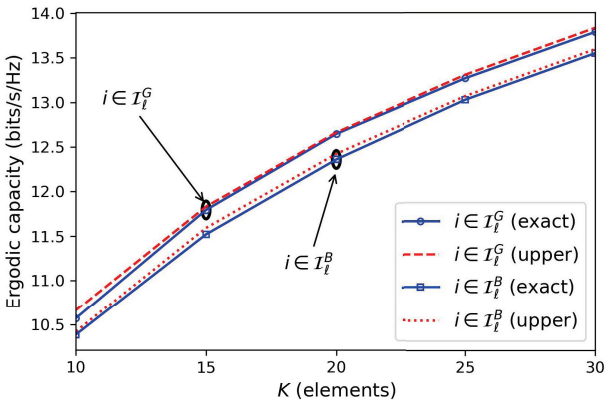


Fig. 7: $C_\ell^{[i]}$ and $C_{\ell|\text{upper}}^{[i]}$ are depicted with respect to K.

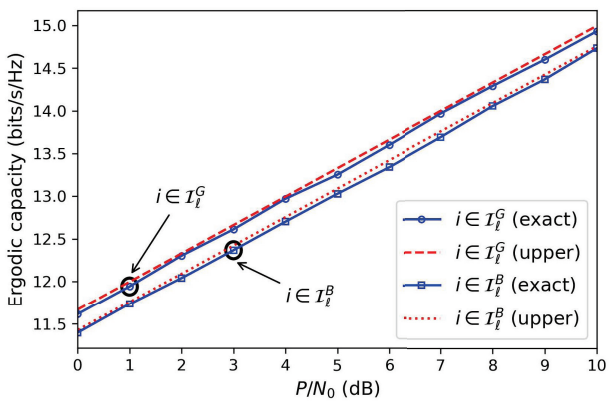


Fig. 8: $C_\ell^{[i]}$ and $C_{\ell|\text{upper}}^{[i]}$ are depicted with respect to P/N_0.

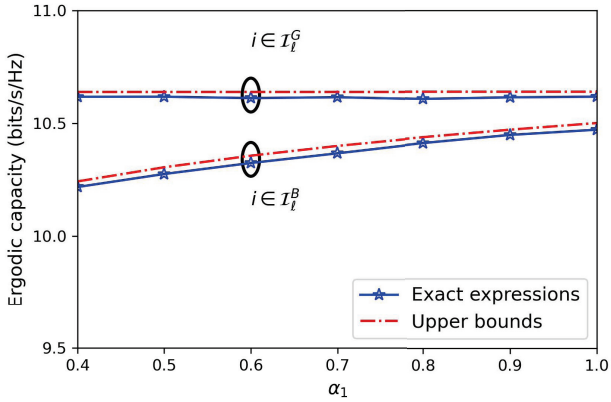


Fig. 9: $C_\ell^{[i]}$ and $C_{\ell|\text{upper}}^{[i]}$ are depicted with respect to α_1.

slot of the ℓ-th frame, M_ℓ IoT devices reduce their transmit power from P to a lower power level (i.e., $\alpha_1 P$ or $\alpha_2 P$) and keep the new power level until the last slot of frame ℓ, thus it can be seen that $C_\ell^{[i]}$ at $i \in \mathcal{I}^{\text{B}} = \mathcal{I}_1^{\text{B}} \cup \mathcal{I}_2^{\text{B}} \cup \mathcal{I}_3^{\text{B}}$ is lower than $C_\ell^{[i]}$ at $i \in \mathcal{I}^{\text{G}}$. As illustrated in Fig. 6, there are $L = 3$ incidents in 3 areas/stages/frames:

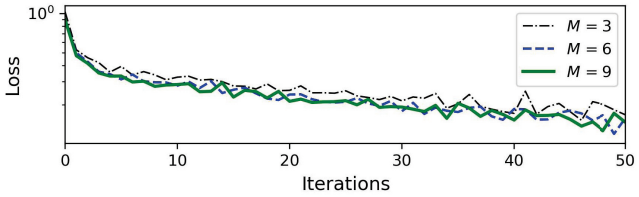
- A power drop occurs at the 61-st slot of the 1-st frame, corresponding to $i = 61$.
- A power drop occurs at the 81-st slot of the 2-nd frame, corresponding to $i = 181$.
- A power drop occurs at the 71-st slot of the 3-rd frame, corresponding to $i = 271$.

We also observe that the ergodic capacity is, in general, improved with the increase of the RIS size K.

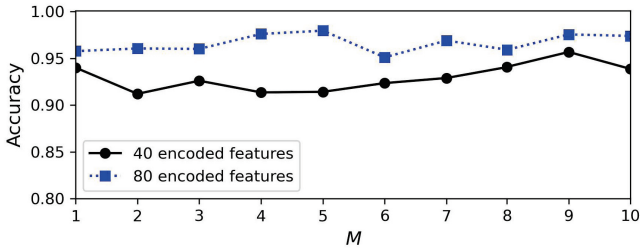
Figure 7 shows the exact ergodic capacity $C_\ell^{[i]}$ and its upper bound $C_{\ell|\text{upper}}^{[i]}$ versus the RIS size K. There are two scenarios depicted in Fig. 7, i.e. $i \in \mathcal{I}_\ell^{\text{G}}$ and $i \in \mathcal{I}_\ell^{\text{B}}$. Naturally, in the case of $i \in \mathcal{I}_\ell^{\text{G}}$, the capacity is higher than that of $i \in \mathcal{I}_\ell^{\text{G}}$. At a specific value of K, the ergodic capacity at $i \in \mathcal{I}_\ell^{\text{G}}$ is higher than the ergodic capacity at $i \in \mathcal{I}_\ell^{\text{B}}$, because there is no sacrifice in transmit power. Obviously, the IoT devices of the ℓ-th area have to reduce their transmit power in the case of $i \in \mathcal{I}_\ell^{\text{B}}$ in order to convey the message of the ℓ-th machine' health, while they do not have to do so when $i \in \mathcal{I}_\ell^{\text{G}}$. Furthermore, either $i \in \mathcal{I}_\ell^{\text{G}}$ or $i \in \mathcal{I}_\ell^{\text{B}}$, the difference between the exact ergodic capacity and its upper bound is relatively small. Obviously, the control of RIS helps to raise the capacity close to its upper limit. More importantly, the ergodic capacity increases with the RIS size K, regardless of the index i, indicating that increasing the number of RIS elements improves the capacity.

Figure 8 shows the ergodic capacity $C_\ell^{[i]}$ and its upper bound $C_{\ell|\text{upper}}^{[i]}$ versus P/N_0 under the condition of $i \in \mathcal{I}_\ell^{\text{G}}$ and that of $i \in \mathcal{I}_\ell^{\text{B}}$. Again, we observe that when $i \in \mathcal{I}_\ell^{\text{G}}$, we do not have to reduce the transmit power to convey the secondary information about the MHC, thus the capacity in the case of $i \in \mathcal{I}_\ell^{\text{G}}$ is higher than that of $i \in \mathcal{I}_\ell^{\text{B}}$. Regardless of the ℓ-th machine's health condition at some time index $i \in \mathcal{I}_\ell$, the capacity increases with P/N_0. Moreover, $C_\ell^{[i]}$ is tightly upper-bounded by $C_{\ell|\text{upper}}^{[i]}$.

Figure 9 shows the exact ergodic capacity $C_\ell^{[i]}$ and its upper bound $C_{\ell|\text{upper}}^{[i]}$ at different values of α_1. Note that $\alpha_2 = 0.4 \leq \alpha_1 \leq 1$. Depending on either $i \in \mathcal{I}_\ell^{\text{G}}$ or $i \in \mathcal{I}_\ell^{\text{B}}$, two curves of $C_\ell^{[i]}$, as well as two curves of $C_{\ell|\text{upper}}^{[i]}$, are depicted. We observe

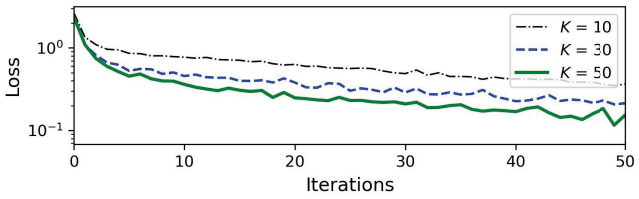


(a) Loss function over iterations, with $\omega = 80$ and different M.

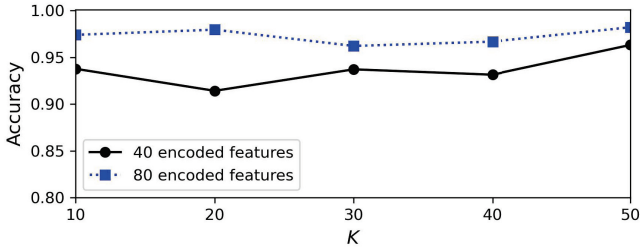


(b) Accuracy vs M at the 50-th iteration.

Fig. 10: The performance vs the number of IoT devices M.



(a) Loss function over iterations, with $\omega = 80$ and different K.



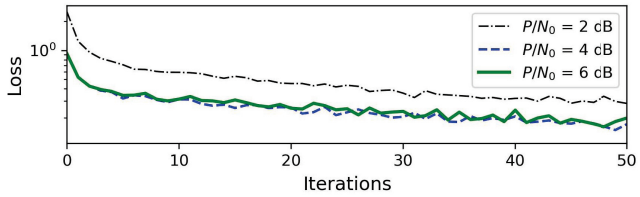
(b) Accuracy vs K at the 50-th iteration.

Fig. 11: The performance vs the number of RIS elements K.

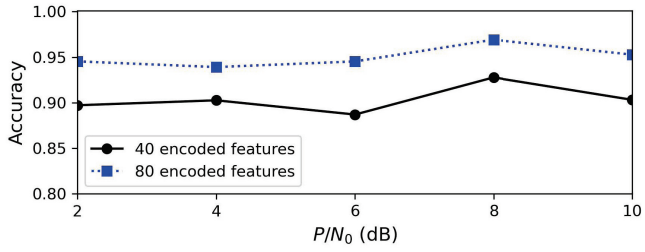
that the capacity is the same in the case of $i \in \mathcal{I}_\ell^{\mathrm{G}}$, because the average transmit power is not changed. By contrast, in the case of $i \in \mathcal{I}_\ell^{\mathrm{B}}$, the average transmit power depends on α_1, thus the capacity increases with α_1, because of the increase in transmit power. Since $\alpha_1 \leq 1$, $C_\ell^{[i]}$ at $i \in \mathcal{I}_\ell^{\mathrm{G}}$ is larger than $C_\ell^{[i]}$ at $i \in \mathcal{I}_\ell^{\mathrm{B}}$ due to the reduced power in the case that the ℓ-th machine is in bad condition. Noticeably, the upper bound $C_{\ell|\text{upper}}^{[i]}$ is quite close to the exact expression $C_\ell^{[i]}$, in both cases of $i \in \mathcal{I}_\ell^{\mathrm{G}}$ and $i \in \mathcal{I}_\ell^{\mathrm{B}}$.

B. Accuracy of MHC Classification

Figure 10 shows the loss function and the accuracy versus the number of IoT devices. From Fig. 10a and Fig. 10b, we can see that the accuracy seems not to be affected by the change



(a) Loss function over iterations, with $\omega = 80$ and different P/N_0.



(b) Accuracy vs P/N_0 at the 50-th iteration.

Fig. 12: The performance vs the ratio P/N_0.

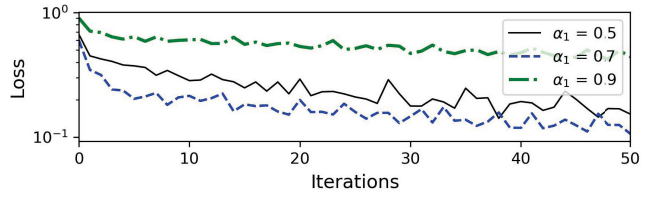
of M. For $1 \leq M \leq 10$, the accuracy exceeds 0.9 when $\omega \in \{40, 80\}$. In the case of $\omega = 80$ neurons, the accuracy is little higher than that of $\omega = 40$ neurons. This may imply that the AE yields a higher accuracy when its architecture becomes more sophisticated (i.e. having more neurons).

Figure 11 shows the loss function and the accuracy versus the number of RIS elements. We can see from Fig. 11a that the loss reduces over iterations. Moreover, the case of $K = 50$ is the best while the case of $K = 10$ is the worst. This means that the increase of the RIS size K helps the AEs to converge faster. However, Fig. 11b shows that a higher value of K does not always guarantee a higher accuracy. In general, the accuracy is not significantly impacted by changing the RIS size. With $\omega = 80$ neurons, we can obtain an accuracy of over 0.95, while the accuracy is below 0.9 with $\omega = 40$ neurons.

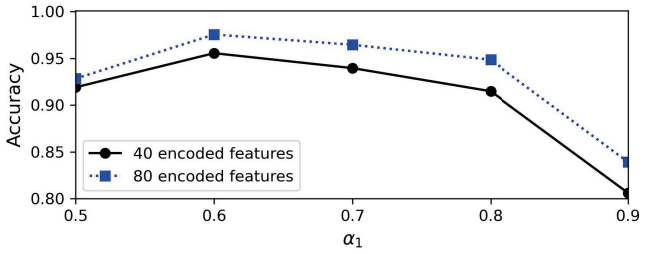
In Fig. 12, the loss function and the accuracy versus P/N_0 are respectively illustrated. In general, Fig. 12a and Fig. 12b show that the accuracy is not much affected by P/N_0. This means that we can perform transmission with low-power IoT devices, while still being able to keep a high value of accuracy. On the other hand, we can reconfirm that the accuracy in the case of $\omega = 80$ is higher that of $\omega = 40$.

Figure 13 depicts the loss function and accuracy versus the power coefficient α_1. As seen from Fig. 13a, the loss has the lowest value in the case of $\alpha_1 = 0.7$. Meanwhile, Fig. 13b shows that with $\omega \in \{40, 80\}$, the accuracy attains its highest value at $\alpha_1 = 0.6$. By contrast, the accuracy is the lowest at $\alpha_1 = 0.9$, regardless of $\omega = 40$ or $\omega = 80$. Thus, we can say that a careful choice of α_1 is necessary to obtain the accuracy as high as possible. Finally, we also confirm again that in the case of $\omega = 80$, the accuracy is higher than that of $\omega = 40$, because the AE can learn the data better.

Remark 1. *The increase of the transmit power P (or the increase of the RIS size K) helps improve the primary information transmission (i.e., the capacity $C_\ell^{[i]}$ is improved),*



(a) Loss function over iterations, with $\omega = 80$ and different α_1.



(b) Accuracy vs α_1 at the 50-th iteration.

Fig. 13: The performance vs the power coefficient α_1.

while the accuracy of machine health assessment is not much affected. Hence, we can perform the transmission of both the primary and secondary information at any desired pair of (P, K), while being able to maintain a high accuracy of machine health assessment.

Remark 2. *The transmission of the primary information is improved when increasing α_1, while the secondary information of machine health is only transmitted with a high accuracy when α_1 is carefully chosen. Hence, there is a trade-off between the data rate associated with the primary information and the accuracy associated with the secondary information about machine health. For example, Fig. 9 shows that $C_\ell^{[i]}$ at $\alpha_1 = 0.9$ is higher than that at $\alpha_1 = 0.7$; however, Fig. 13 shows that the accuracy at $\alpha_1 = 0.9$ approximates to 0.81, which is lower than the accuracy of 0.92 at $\alpha_1 = 0.7$. In this case, we can accept reducing the data rate a little, i.e. reducing α_1 from 0.9 to 0.7, in order to gain a higher accuracy, from 0.81 to 0.92.*

VII. Conclusions

In this paper, we have developed an AE-based framework to facilitate the monitoring of MHCs in a smart factory using industrial IoT and RIS devices. We have found that while the increase of the IoT devices' transmit power P or RIS size K can improve the ergodic capacity, the accuracy of machine health assessment is not really affected. Thus, it is the discretion of the designers to select a reasonable pair of (P, K) to achieve two goals: i) a high data rate and ii) a high accuracy of fault detection. Additionally, we have observed that the choice of coefficient α_1 in our power mapping scheme has a significant influence on both the capacity and the accuracy. Given that the capacity is theoretically derived but the accuracy is experimentally quantified, the optimal value of α_1 can only be found by a heuristic approach. As a result, the designers will need to tune α_1 to achieve a trade-off between the capacity and the accuracy. In general, the proposed framework allows us to assess the MHC with high accuracy by lowering transmit power to a lower level but still being able to keep a reasonable transmission rate with the help of RIS. The proposed AE can attain a high accuracy of over 95% as long as the number of neurons at the middle layer is sufficiently large. The significance of our framework lies in that low-power IoT-aided systems can sacrifice part of the data rate, associated with the primary information, in order to transmit a totally-different type of information about the MHCs (i.e., the secondary information). Nevertheless, the sacrificed data rate can be compensated by integrating RISs into the system. In future works, it may be necessary to extend this work to a more general case, where more power levels are used for delivering more alert levels.

Appendix

A. Proof of Proposition 1

Let us consider a function $f(\gamma_\ell^{[i,w]}) = \log_2(1 + a\gamma_\ell^{[i,w]})$ where $\gamma_\ell^{[i,w]} = \sum_{m=1}^{M_\ell} |\chi_{m\ell}^{[i,w]}|^2$. The expectation of $f(\gamma_\ell^{[i,w]})$ can be calculated as follows:

$$\begin{aligned}
&\mathbb{E}\left\{f(\gamma_\ell^{[i,w]})\right\} \\
&= \mathbb{E}\left\{\log_2\left(1 + a\sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}} |\mathbf{g}_{\ell\to C}^{[i,w]}\, \mathsf{diag}\left(\boldsymbol{\phi}_\ell\right) \mathbf{h}_{m\to\ell}^{[i,w]}|^2\right)\right\} \\
&= \mathbb{E}\left\{\log_2\left(1 + a\sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}} \left|\mathbf{u}_{m,\ell}^{[i,w]} \boldsymbol{\phi}_\ell^\top\right|^2\right)\right\}, \qquad (34)
\end{aligned}$$

where the row vector $\mathbf{u}_{m,\ell}^{[i,w]} = \mathbf{g}_{\ell\to C}^{[i,w]} \odot \left(\mathbf{h}_{m\to\ell}^{[i,w]}\right)^\top$ is an element-wise product.

Using the Cauchy-Schwarz inequality, we have

$$\left|\mathbf{u}_{m,\ell}^{[i,w]} \boldsymbol{\phi}_\ell^\top\right|^2 \le \left\|\mathbf{u}_{m,\ell}^{[i,w]}\right\|^2 \|\boldsymbol{\phi}_\ell\|^2, \qquad (35)$$

where the equality holds for

$$\boldsymbol{\phi}_\ell = \frac{\mathbf{u}_{m,\ell}^{[i,w]} \boldsymbol{\phi}_\ell^\top}{\left\|\mathbf{u}_{m,\ell}^{[i,w]}\right\|^2} \left(\mathbf{u}_{m,\ell}^{[i,w]}\right)^*. \qquad (36)$$

Adjusting the ℓ-th RIS to obey (36) so that $\|\boldsymbol{\phi}_\ell\|^2 = K$ is satisfied, we can arrive at the following:

$$\begin{aligned}
\left|\mathbf{u}_{m,\ell}^{[i,w]} \boldsymbol{\phi}_\ell^\top\right|^2 &= K\left\|\mathbf{u}_{m,\ell}^{[i,w]}\right\|^2 = K\left\|\mathbf{g}_{\ell\to C}^{[i,w]} \odot \left(\mathbf{h}_{m\to\ell}^{[i,w]}\right)^\top\right\|^2 \\
&= K\sum_{k=1}^{K} \left|g_{k|\ell\to C}^{[i,w]}\right|^2 \left|h_{k|m\to\ell}^{[i,w]}\right|^2, \qquad (37)
\end{aligned}$$

where $g_{k|\ell\to C}^{[i,w]}$ is the k-th entry in the vector $\mathbf{g}_{\ell\to C}^{[i,w]}$, and $\mathbf{h}_{k|m\to\ell}^{[i,w]}$ is the k-th entry in the vector $\mathbf{h}_{m\to\ell}^{[i,w]}$. Substituting (37) into (34), we obtain

$$\begin{aligned}
&\mathbb{E}\left\{f(\gamma_\ell^{[i,w]})\right\} \\
&= \mathbb{E}\Bigg\{\log_2\Bigg(1 + a\sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}} K
\end{aligned}$$

$$\times \sum_{k=1}^{K} \left| g_{k|\ell\to C}^{[i,w]} \right|^2 \left| h_{k|m\to\ell}^{[i,w]} \right|^2 \Bigg) \Bigg\}$$

$$\overset{(\star)}{\leq} \log_2 \Bigg(1 + a \sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}} K \times \mathbb{E} \left\{ \sum_{k=1}^{K} \left| g_{k|\ell\to C}^{[i,w]} \right|^2 \left| h_{k|m\to\ell}^{[i,w]} \right|^2 \right\} \Bigg)$$

$$\overset{(\star\star)}{=} \log_2 \Bigg(1 + a \sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}} K \times \sum_{k=1}^{K} \mathbb{E} \left\{ \left| g_{k|\ell\to C}^{[i,w]} \right|^2 \right\} \mathbb{E} \left\{ \left| h_{k|m\to\ell}^{[i,w]} \right|^2 \right\} \Bigg)$$

$$= \log_2 \left(1 + aK^2 \sum_{m=1}^{M_\ell} 10^{-\frac{\varrho_{m\ell}}{10}} \right), \tag{38}$$

where $(\star)$ follows the Jensen's inequality and $(\star\star)$ follows the fact that $g_{k|\ell\to C}^{[i,w]}$ is independent of $h_{k|m\to\ell}^{[i,w]}$.

REFERENCES

[1] H. Ahuett-Garza and T. Kurfess, "A brief discussion on the trends of habilitating technologies for Industry 4.0 and smart manufacturing," *Manufacturing Letters*, vol. 15, pp. 60–63, 2018, industry 4.0 and Smart Manufacturing.

[2] J. Lee, H. Davari, J. Singh, and V. Pandhare, "Industrial artificial intelligence for Industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 18, pp. 20–23, 2018.

[3] J. García-Morales, M. C. Lucas-Estan, and J. Gozalvez, "Latency-sensitive 5G RAN slicing for Industry 4.0," *IEEE Access*, vol. 7, pp. 143 139–143 159, 2019.

[4] G. Berardinelli, N. H. Mahmood, I. Rodriguez, and P. Mogensen, "Beyond 5G wireless IRT for Industry 4.0: Design principles and spectrum aspects," in *IEEE Globecom Workshops (GC Wkshps)*, 2018, pp. 1–6.

[5] J. Yan, Y. Meng, L. Lu, and L. Li, "Industrial big data in an industry 4.0 environment: Challenges, schemes, and applications for predictive maintenance," *IEEE Access*, vol. 5, pp. 23 484–23 491, 2017.

[6] M. Compare, P. Baraldi, and E. Zio, "Challenges to IoT-enabled predictive maintenance for Industry 4.0," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4585–4597, 2020.

[7] N. Saxena, A. Roy, B. J. R. Sahu, and H. Kim, "Efficient IoT gateway over 5G wireless: A new design with prototype and implementation results," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 97–105, 2017.

[8] C. L. Stergiou, K. E. Psannis, and B. B. Gupta, "IoT-based big data secure management in the fog over a 6G wireless network," *IEEE Internet of Things J.*, vol. 8, no. 7, pp. 5164–5171, 2021.

[9] H. X. Nguyen, T. Ramona, D. To, and M. Tatipamula, "Digital twin for 5G and beyond," *IEEE Communications Mag.*, vol. 59, pp. 10–15, 2021.

[10] H. Rahimi, A. Zibaeenejad, and A. A. Safavi, "A novel IoT architecture based on 5G-IoT and next generation technologies," in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2018, pp. 81–88.

[11] K. Shahzad and M. O'Nils, "Condition monitoring in industry 4.0-design challenges and possibilities: A case study," in *2018 Workshop on Metrology for Industry 4.0 and IoT*, 2018, pp. 101–106.

[12] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the Internet of things and industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[13] E. Basar, "Reconfigurable intelligent surface-based index modulation: A new beyond mimo paradigm for 6g," *IEEE Transactions on Communications*, vol. 68, no. 5, pp. 3187–3196, 2020.

[14] D. Dardari and N. Decarli, "Holographic communication using intelligent surfaces," *IEEE Communications Magazine*, vol. 59, no. 6, pp. 35–41, 2021.

[15] E. Bjornson and L. Sanguinetti, "Power scaling laws and near-field behaviors of massive MIMO and intelligent reflecting surfaces," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 1306–1324, 2020.

[16] D. P. Filev, R. B. Chinnam, F. Tseng, and P. Baruah, "An industrial strength novelty detection framework for autonomous equipment monitoring and diagnostics," *IEEE Trans. on Ind. Info.*, vol. 6, no. 4, pp. 767–779, 2010.

[17] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, "Machine learning for predictive maintenance: A multiple classifier approach," *IEEE Trans. on Ind. Info.*, vol. 11, no. 3, pp. 812–820, 2015.

[18] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Trans. on Ind. Electron.*, vol. 63, no. 5, pp. 3137–3147, 2016.

[19] W. Yu, T. Dillon, F. Mostafa, W. Rahayu, and Y. Liu, "A global manufacturing big data ecosystem for fault detection in predictive maintenance," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 183–192, 2020.

[20] R. K. Pathinarupothi, P. Durga, and E. S. Rangan, "IoT-based smart edge for global health: Remote monitoring with severity detection and alerts transmission," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2449–2462, 2019.

[21] P. Mursia, V. Sciancalepore, A. Garcia-Saavedra, L. Cottatellucci, X. C. Pérez, and D. Gesbert, "RISMA: Reconfigurable intelligent surfaces enabling beamforming for IoT massive access," *IEEE J. on Sel. Areas in Commun.*, vol. 39, no. 4, pp. 1072–1085, 2021.

[22] A. Bhowal, S. Aïssa, and R. Singh Kshetrimayum, "RIS-assisted spatial modulation and space shift keying for ambient backscattering communications," in *IEEE Int. Conf. on Commun. (ICC)*, 2021, pp. 1–6.

[23] L. Jiao, P. Wang, A. Alipour-Fanid, H. Zeng, and K. Zeng, "Enabling efficient blockage-aware handover in RIS-assisted mmWave cellular networks," *IEEE Trans. on Wirel. Commun.*, pp. 1–1, 2021.

[24] M. Nemati, J. Park, and J. Choi, "RIS-assisted coverage enhancement in millimeter-wave cellular networks," *IEEE Access*, vol. 8, pp. 188 171–188 185, 2020.

[25] G. Ghatak, S. R. Khosravirad, and A. De Domenico, "Stochastic geometry framework for ultra-reliable cooperative communications with random blockages," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

Tiep M. Hoang received the B.Eng. degree from the HCMC University of Technology, Vietnam, in 2012, the M.Eng. degree from Kyung Hee University, South Korea, in 2014, and the Ph.D. degree from the Queen's University of Belfast, United Kingdom, in 2019. From 2020 to 2022, he was a (postdoctoral) Research Fellow with the School of Electronics and Computer Science, the University of Southampton, United Kingdom. Since May 2022, he has been a Postdoctoral Fellow with the Department of Electrical Engineering, the University of Colorado Denver, United States. His current research interests include 5G/6G wireless communications, wireless security and authentication, reconfigurable intelligent surface (RIS), convex optimization, and machine learning.

Son Dinh-Van received the B.S. degree from Hanoi University of Science and Technology, Vietnam, in 2013, the M.S. degree from Soongsil University, Seoul, South Korea, in 2015, and the Ph.D. degree from Queen's University of Belfast, Belfast, U.K., in 2019, all in electrical engineering. He is currently a Research Fellow with the Intelligent Vehicles Research Group, WMG, University of Warwick, U.K. He was a Data Scientist with Frequenz Gmbh, Germany, and a Visiting Researcher with Middlesex University, U.K, in 2020 and 2021, respectively. His current research interests include industrial Internet of Things, 5G/6G communications, PHY-layer security, millimeter-wave, and future artificial wireless communications systems.

Balbir Barn is Professor of Software Engineering at Middlesex University. Following a PhD in Computer Science from the University of Bath, he embarked on an industrial research career of over 15 years working in the various research labs at Marconi, Texas Instruments and Sterling Software where he was involved in the research and design of leading software products such as the IEF. He joined Middlesex in 2008 as Associate Dean for Business in the School of Engineering and Information Sciences after an earlier academic role as Head of Department and Professor of Computing at the University of West London. He has also provided consultancy to prestigious clients such as JP Morgan Chase and JISC.

Ramona Trestian is a Senior Lecturer with Design Engineering and Mathematics Department, Faculty of Science and Technology, Middlesex University, London, UK. She was previously an IBM-IRCSET Exascale Postdoctoral Researcher with the Performance Engineering Laboratory (PEL) at Dublin City University (DCU), Ireland since December 2011. She was awarded the PhD from Dublin City University in March 2012 and the B.Eng. in Telecommunications from Technical University of Cluj-Napoca, Romania, 2007. She published in prestigious international conferences and journals and has five edited books. Her research interests include mobile and wireless communications, quality of experience, multimedia streaming, and digital twin modelling. She is an Associate Editor of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS.

Huan X. Nguyen (M'06–SM'15) received the B.Sc. degree from the Hanoi University of Science and Technology, Vietnam, in 2000, and the Ph.D. degree from the University of New South Wales, Australia, in 2007. He is currently a Professor of Digital Communication Engineering at Middlesex University London (U.K.), where he is also the Director of the London Digital Twin Research Centre and Head of the 5G/6G & IoT Research Group. He leads research activities in digital twin modelling, 5G/6G enabling technologies, IoT communication, and machine learning within his university with focus on digital transformation, industry 4.0 and critical applications (disasters, intelligent transportation, health). He has been leading 10+ major council/industry funded projects, publishing 120+ peer-reviewed research papers, and serving as chairs for international conferences (ICT'21, ICEM2021, ICT'20, ICT'19, IWNPD'17, PIMRC'20, FoNeS-IoT'20, ATC'15). He is an Adjunct Professor at Van Lang University in Vietnam.