AFRL-AFOSR-UK-TR-2022-0035

Internet of Complex Things

Marchetti, Nicola
UNIVERSITY OF DUBLIN, TRINITY COLLEGE
COLLEGE GREEN
DUBLIN, , 2
IE

03/25/2022
Final Technical Report

Air Force Research Laboratory
Air Force Office of Scientific Research
European Office of Aerospace Research and Development
Unit 4515 Box 14, APO AE 09421

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED	
20220325	Final	START DATE 20180615	END DATE 20210314

4. TITLE AND SUBTITLE
Internet of Complex Things

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
	FA9550-18-1-0214	61102F
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
Nicola Marchetti

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
UNIVERSITY OF DUBLIN, TRINITY COLLEGE
COLLEGE GREEN
DUBLIN 2
IE

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
EOARD UNIT 4515 APO AE 09421-4515	AFRL/AFOSR IOE	AFRL-AFOSR-UK-TR-2022-0035

12. DISTRIBUTION/AVAILABILITY STATEMENT
A Distribution Unlimited: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This projected started in August 2018 under a standard three-year grant and endured the COVID pandemic. The PI received a COVID-inspired 9 month No Cost Extension. There are three members on this research team: The PI, the Co-IP, and a Post-Doc. They also partnered In total the team generated 3 journal articles: IEEE Communications, IEEE Internet of Things Journal; and IEEE Transactions on Instrumentation Measurement. This work is satisfactory and of great interest to the USAF and DoD.

The researchers are working as part of the Wireless Engineering and Complexity Science (Why-COM) lab in Trinity College Dublin. The research center is aimed at proposing and investigating: (i) new theories of wireless networks based on complex systems science and its application to network analysis and resource management; and (ii) component technologies for 5G and beyond. The research work concluded with the following two research papers:

1. Dey, H. Joshi and N. Marchetti, \Space-Time Spreading aided Distributed MIMO-WSNs", IEEE Communications Letters, vol. 25, no. 4, pp. 1338-1342, April 2021.
2. M. Dzaferagic, N. Marchetti, and I. Macaluso, "Fault detection and classification in Industrial IoT in case of missing sensor data", IEEE Internet of Things Journal (submitted).

If the DoD is to take advantage of future generations of wireless sensors, extending the lifetime of wireless sensors, devices, and networks is essential (i.e., reduce energy consumption). Any gains in energy efficiency must also be accomplished without significantly reducing the wireless network's sensing capabilities.

The PI's first work proposes space-time spreading (STS) of local sensor observations and decisions before reporting them over the local network in order to reduce the chance of interference. STS should enable each sensor in the network to transmit in a specific space-time block when all the other sensors are silent. Thus, this approach significantly reduces the possibility of interference in local communications. This approach sounds straightforward and should be relatively easy to implement at no cost to improve energy efficiency without degrading performance for 5G devices and networks.

The PI's second work is looking at the limits of infrastructure sharing, and specifically using blockchain-enabled smart contracts to regulate agreements between mobile networks and small to medium sized network infrastructure providers. The idea is to explore the concepts of using blockchain-inspired smart contracts to automatically build, enforce, guarantee, and/or assure interactions in a trusted environment by preventing mischievous behavior from any of the contracting parties.

These are both timely issues particular for emerging 5G and future 6G wireless devices and networks, and communications technologies in general. Work done to improve energy efficiency and/or trust of these high bandwidth, low-power networks is very significant for just about every aspect of modern connected life.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**		
U	U	U	SAR	22

19a. NAME OF RESPONSIBLE PERSON	**19b. PHONE NUMBER** *(Include area code)*
LOGAN MAILLOUX	314 235 6163

CONNECT/AFOSR Final Performance Report

INTERNET OF COMPLEX THINGS

Complex IoT

PI: Prof. Nicola Marchetti
Co-PI: Dr. Irene Macaluso

Grant Number: FA9550-18-1-0214
Period of Performance: June 2020 - March 2021

Research Institution:
CONNECT Centre
Trinity College Dublin
The University of Dublin
Ireland

Contents

1 Summary

Prof Nicola Marchetti leads the Wireless Engineering and Complexity Science (WhyCOM) lab in Trinity College Dublin, which aims at proposing and investigating: (i) a new theory of wireless networks based on complex systems science and its application to network analysis and resource management; (ii) several component technologies for beyond 5G wireless communication systems, including massive antenna arrays and self-organising networks. Dr Irene Macaluso is a senior research fellow in WhyCOM.

This project enabled two research activities over the past year (performed during the no-cost extention period from June 2020 till March 2021)[1], one of them related to the PI interaction with another WhyCOM member, Prof Indrakshi Dey.

A brief description of such activities is provided later on in this Summary section, and more details will be available in the actual research articles that are appended to this document:

1. I. Dey, H. Joshi and N. Marchetti, "Space-Time Spreading aided Distributed MIMO-WSNs", *IEEE Communications Letters*, vol. 25, no. 4, pp. 1338-1342, April 2021.

2. M. Dzaferagic, N. Marchetti, and I. Macaluso, "Fault detection and classification in Industrial IoT in case of missing sensor data", *IEEE Internet of Things Journal* (submitted).

These two works are related to different aspects of cyber-physical systems and modern large scale networks, namely:

- The first work focuses on reducing energy consumption in a Wireless Sensor Network (WSN) by enabling energy efficient transmission of sensor observations/decisions.

- The second work focuses on the issue of reliability in Industrial Internet of Things (IIoT) in case of missing sensors measurements due to network or hardware problems.

1.1 The Energy Problem

Extending the lifetime of a Wireless Sensor Network (WSN) without compromising its sensing capabilities is a key objective if wireless solutions will ever replace wired ones [1, 2]. The great advantage of having untethered sensors is that they can be positioned where most needed or even randomly scattered in an environment. Specific applications in which sensors are difficult to reach or difficult to replace

[1]Please notice that the research performed in the first two years of the project (periods Jun 2018-Jun 2019 and Jun 2019-Jun 2020) has been described in the corresponding annual reports already submitted.

(e.g. sensors embedded in concrete structures [3] or spread over agricultural areas [4]) call then for a careful analysis of lifetime-extending strategies, identifying the key components of energy efficiency in WSNs [5]. Aside from creating more energy efficient sensors, or designing specific radio protocols to enhance energy conservation, a simple yet effective solution is to sense only when and where necessary.

A unique way of reducing energy consumption in a WSN is to enable energy efficient transmission of sensor decisions. This can be done by reducing the transmission energy required for fighting impairments like fading, shadowing, scattering inherent to the wireless propagation environment, as well as intrinsic interference resulting from superposition of data streams transmitted from multiple sensors at the same time and over the same frequency band or closely-spaced frequency sub-bands. Therefore Space-Time Spreading (STS) of sensor decisions is implemented in our work [6] before reporting them in order to reduce the chance of interference, thereby decreasing the requirement of transmission energy necessary to overcome interference due to superposition of transmitted data streams. The key idea of STS-aided WSN is to map the information transmitted from the sensors on indexed space-time blocks of fixed duration by multiplying individual information vectors with different dispersion vectors, such that only one transmit sensor is activated on a particular space-time block when all the other sensors are silent. Since no two sensors transmit on the same space-time slot, possibility of interference is almost eliminated.

The presence of multiple antennas at the receive side along with multiple transmit sensors resembles a 'virtual' multiple-input-multiple-output (MIMO) or massive MIMO system [7, 8] and allows for utilization of array processing gain providing spectral efficiency, fading mitigation and low-energy sensor adoption. The decision fusion center (DFC) receives a superposition of the sensor observations received over the multiple access channel (MAC) and then uses optimum or sub-optimum fusion rules to arrive at a reliable decision on the phenomenon observed or detected by the sensors. We compare and formulate log-likelihood ratio (LLR) based optimum and several sub-optimum (maximal ratio combining (MRC), modified MRC (mMRC), widely linear (WL), Chair-Varshney (CV)-maximum likelihood (CV-ML), CV-zero forcing (CV-ZF), CV-minimum mean squared error (CV-MMSE)) fusion rules to arrive at a reliable conclusion.

The total transmit energy consumption in the network can be calculated using the link budget relationship,

$$E_{\mathrm{STS}} = \frac{4\pi^2 d^{\sqrt{\rho^\lambda}} \mathcal{M}^\lambda \mathcal{N}_F^\lambda E_r}{\mathcal{G}_t \mathcal{G}_r \mathcal{L}^2 N} \tag{1}$$

where M is the total number of sensors in the network, E_r is the average energy consumption per bit required to maintain a certain bit error rate (BER) at the server/destination, d denotes the average distance, N is the number of antennas at the DFC, $\mathcal{G}_t$ and $\mathcal{G}_r$ are transmit and receive antenna gain respectively, $\mathcal{L}$ is the wavelength of operation, $\mathcal{M}^\lambda$ is the link margin and $\mathcal{N}_F^\lambda$ is the noise figure. Assuming

that the BER can be calculated using $p_b = \int_0^\infty \mathcal{Q}(\sqrt{2xE_r/N_0N})f(x)dx$, we can calculate E_r in terms of p_b. It is worth-mentioning here that $f(x)$ is the probability density function of the Q-function and will vary depending on the composite fading-shadowing distribution of $\mathbf{G}^\lambda$. Putting the expression of E_r in terms of p_b back in (1), we can obtain the total transmit energy consumption depending on the target BER of the network and the link quality and link distribution. Without STS, if all M sensors at the same time over the λth space time block, the total energy consumption for transmitting each bit will be equal to,

$$E_{\text{w/oSTS}} = \frac{4\pi^2 d\sqrt{\rho^\lambda} M^2 \mathcal{M}^\lambda \mathcal{N}_F^\lambda E_r}{\mathcal{G}_t \mathcal{G}_r \mathcal{L}^2 N} \tag{2}$$

Therefore STS reduce energy consumption by a factor of $1/M^2$, reducing energy with the increase in the number of sensors.

1.2 Reliability in IIoT in case of missing sensor data

In the era of Internet of Things (IoT) big data, the integration of cloud computing technologies and cyber-physical systems enables the full potential of Industry 4.0 to be harvested in manufacturing processes, with a multitude of sensors being installed around the industrial operating environment and equipment. A key component of the IIoT is a monitoring system to track and potentially predict the various indicators of the health of a machine/component or an entire industrial process. The reliability of such monitoring system should be guaranteed also in case of missing sensors' data so as to avoid disruption even if some measurements are not received. In fact values are often missing from the collected sensor data, and the related issue of missing value imputation becomes then very important. For example, high-frequency data collection can result in large gaps in the data and if the network stops working, all the measurements collected during the network downtime will be missing [9]. Other possible reasons behind missing data are: faulty sensors producing intermittent readings, loss of data during wireless communication owing to packet loss or to interference in the communication medium, or data removed purposely by attackers with malicious intentions during sensing, processing, storing or communication.

While fault detection and identification [10–12] and data imputation [9, 13] have been studied before, all the approaches reported to date in the literature focus on either of these issues separately. We instead propose a framework that unifies all three techniques, allowing us to optimize each of them in a way that results in the best overall performance of the monitoring system. In particular, we propose a data-driven decomposition of the process to monitor the various indicators of the health of a machine/component or an entire industrial process. The monitoring process includes three components: i) the fault detection module, ii) the fault identification module, and iii) the data imputation module. Specifically, the data imputation module in our framework relies on a Generative Adversarial Network (GAN) model

that learns the correlation between the data from the input layer to replace missing sensor measurements. The GAN was optimized by validating its performance based on the effect of the imputed measurements on the fault identification and detection modules, which ultimately constitute the two essential tasks performed by the monitoring system. We conducted a thorough evaluation of the proposed approach using the extended Tennessee Eastman Process dataset. Our evaluation shows that the GAN-imputed data mitigate the loss on the fault detection and identification modules, even in the case of persistently missing measurements from sensors that are critical for the correct functioning of the monitoring system.

2 References

References

[1] Y. Murillo, B. Reynders, A. Chiumento, S. Malik, P. Crombez, and S. Pollin, "Bluetooth now or low energy: Should ble mesh become a flooding or connection oriented network?" in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 2017, pp. 1–6.

[2] A. Chiumento, B. Reynders, Y. Murillo, and S. Pollin, "Building a connected ble mesh: a network inference study," in *2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, 2018, pp. 296–301.

[3] G. Loubet, A. Takacs, and D. Dragomirescu, "Towards the design of wireless communicating reinforced concrete," *IEEE Access*, vol. 6, pp. 75 002–75 014, 2018.

[4] C. Cambra, S. Sendra, J. Lloret, and L. Garcia, "An iot service-oriented system for agriculture monitoring," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.

[5] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Computer Networks*, vol. 67, pp. 104 – 122, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128614001418

[6] I. Dey, H. Joshi, and N. Marchetti, "Space-time spreading aided distributed MIMO-WSNs," *IEEE Commun. Letters*, vol. 25, no. 4, pp. 1338–1342, Apr. 2021.

[7] I. Dey, M. M. Butt, and N. Marchetti, "Throughput analysis for virtual MIMO WSNs over measured MIMO channels," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 297–299, Aug. 2019.

[8] D. Ciuonzo, G. Romano, and P. Salvo Rossi, "Channel-aware decision fusion in distributed MIMO wireless sensor networks: Decode-and-fuse vs. decode-then-fuse," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2976–2985, Aug. 2012.

[9] Y. Liu, T. Dillon, W. Yu, W. Rahayu, and F. Mostafa, "Missing value imputation for industrial iot sensor data with large gaps," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6855–6867, 2020.

[10] S. Yin, J. J. Rodriguez-Andina, and Y. Jiang, "Real-time monitoring and control of industrial cyberphysical systems: With integrated plant-wide monitoring and control framework," *IEEE Industrial Electronics Magazine*, vol. 13, no. 4, pp. 38–47, 2019.

[11] W. Dai, H. Nishi, V. Vyatkin, V. Huang, Y. Shi, and X. Guan, "Industrial edge computing: Enabling embedded intelligence," *IEEE Industrial Electronics Magazine*, vol. 13, no. 4, pp. 48–56, 2019.

[12] H. Hellstrom, M. Luvisotto, R. Jansson, and Z. Pang, "Software-defined wireless communication for industrial control: A realistic approach," *IEEE Industrial Electronics Magazine*, vol. 13, no. 4, pp. 31–37, 2019.

[13] C. Kalalas and J. Alonso-Zarate, "Sensor data reconstruction in industrial environments with cellular connectivity," in *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, 2020, pp. 1–6.

Space-Time Spreading Aided Distributed MIMO-WSNs

Indrakshi Dey, *Senior Member, IEEE*, Hemdutt Joshi, *Member, IEEE*, and Nicola Marchetti, *Senior Member, IEEE*

***Abstract*— In this letter, we consider the plaguing, yet rarely handled problem of interference resulting from superposition of multiple sensor signals in time, when sent over a multiple access channel (MAC) in wireless sensor networks (WSNs). We propose space-time spreading (STS) of local sensor decisions before reporting them over a MAC to i) minimize interference and ii) reduce energy required for combating interference due to superposition of sensor decisions. Each sensor decision is encoded on appropriately indexed space-time block of fixed duration using dispersion vectors, such that a single sensor is activated over each space-time block while all the other sensors are silent. At the receive side of the reporting channel, we assume a multi-antenna decision fusion center (DFC), thereby representing a distributed multiple-input-multiple-output (MIMO) communication scenario. We formulate and compare optimum and sub-optimum fusion rules for fusing sensor decisions at the DFC to arrive at a reliable conclusion. Simulation results demonstrate gain in fusion performance with STS-aided transmission by 3 to 6 times over performance without STS.**

***Index Terms*— Wireless sensor networks, decision fusion, space-time spreading, Internet of Things, distributed MIMO.**

I. Introduction

IN TRADITIONAL WSNs, each sensor is allocated a dedicated orthogonal channel for transmitting their local observations. However, in the emerging paradigm of IoT that involves coexistence of a multitude of sensors, the bandwidth requirement increases linearly with the number of sensors [1]. Hence, in large-scale WSNs, all sensors transmit their decisions simultaneously over a MAC to a DFC that receives a superposition of the sensor decisions.

In a MAC scenario, the bandwidth requirement no longer depends on the number of sensors. However, DF in MAC is corrupted with intrinsic interference resulting from inter-sensor-element interference (ISEI) and inter-sensor-channel interference (ISCI) [2]. Interference caused by partial overlap of multiple sensor signals in time results in ISEI, while ISCI is caused by the superposition of sensor signals when sent over a MAC. Together with interference, the wireless MAC suffers from random time-varying fading and shadowing.

Manuscript received September 11, 2020; revised November 12, 2020; accepted December 14, 2020. Date of publication December 21, 2020; date of current version April 9, 2021. This material is based upon work supported by the Air force Office of Scientific Research under award number FA9550-18-1-0214 and co-funded by Science Foundation Ireland (SFI) and is co-funded under the European Regional Development Fund under Grant Number 13/RC/2077. The associate editor coordinating the review of this letter and approving it for publication was E. Basar. *(Corresponding author: Indrakshi Dey.)*

Indrakshi Dey is with the CONNECT, National University of Ireland, Maynooth 1, W23 F2H6 Ireland (e-mail: indrakshi.dey@mu.ie).

Hemdutt Joshi is with the Department of Electronics and Communication Engineering (ECED), Thapar Institute of Engineering and Technology, Patiala 147004, India (e-mail: hemdutt.joshi@thapar.edu).

Nicola Marchetti is with the CONNECT, Trinity College Dublin, Dublin 2, D02 PN40 Ireland (e-mail: nicola.marchetti@tcd.ie).

Digital Object Identifier 10.1109/LCOMM.2020.3046231

In order to combat fading, shadowing and interference inherent to MAC, the transmission energy required for covering a single-hop can be quite high to compensate for the signal losses incurred by the environment. But sensor nodes are energy-constrained and low-power, and high energy transmission is taxing on the battery life of the sensors. The rise in the number of IoT-based applications incorporating WSNs, has therefore increased the need for energy-efficient transmission over wireless MAC suffering from ISEI and ISCI along with fading and shadowing.

Implementing multiple antennas at the DFC is recommended in [3], [4] to improve fusion performance in deep fading and shadowing condition, thereby leveraging a 'virtual' MIMO or massive MIMO (mMIMO) channel between the sensors and the DFC. The results in [3], [4] are extended in [5] to scenarios with non-perfect CSI for both antipodal and non-antipodal signalling formats. However, MIMO and mMIMO based DF in MAC is still corrupted with ISEI and ISCI. An alternative solution is to group sensors into clusters [6], where sensors within each cluster report their observations to their cluster-head and each cluster-head reports its decision to the DFC over each time-slot. However, such a technique, though capable of minimizing co-channel interference, involves two-stage fusion and introduces extra error in the fusion process.

Orthogonality in Space-time coding (STC) aided MIMO [7]–[9] can fight interference and fading in multi-access environments, but at the cost of high encoding and decoding complexities. However, STC-aided MIMO or mMIMO based WSN is not a viable option owing to energy, bandwidth, complexity constraints and higher system knowledge (like channel parameters, sensor local decisions) requirements.

The primary contribution of this letter is to propose i) space-time spreading (STS) of sensor decisions on the transmit side to minimize interference in an energy-efficient way before receiving them over a wireless MAC and ii) fusing the decisions at multi-antenna DFC with the aim of achieving significant improvement in fusion performance in presence of deep fading and shadowing. The key idea of STS aided WSN is to map local sensor decisions on indexed space-time blocks of fixed duration by multiplying individual decision vectors with different dispersion vectors, such that each sensor transmits on a particular space-time slot when all the other sensors are silent. Since no two sensors transmit on the same space-time slot, possibility of ISEI and ISCI is almost eliminated. Moreover, no extra energy is needed to overcome interference, thereby resulting in energy-efficient interference-free transmission over large-scale WSNs at a reduced complexity. We derive two sets of sub-optimum DF techniques with reduced complexity for the received signal at the DFC consisting of a) Decode-and-fuse and b) Decode-then-fuse

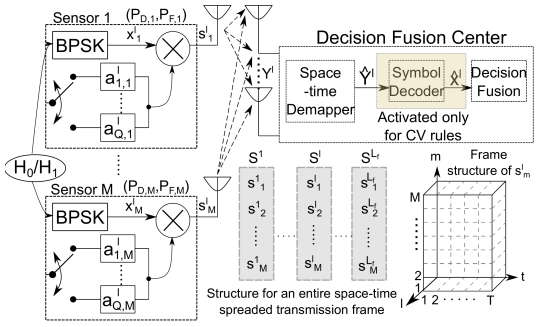


Fig. 1. ST Spreading aided WSN with distributed MIMO DF.

rules, generalizing to our set-up those introduced in [4], [10] for MIMO and mMIMO DF context. Towards this end, this letter will clarify on how distributed DF will perform in STS aided WSNs.

II. System Overview

Fig. 1 portrays the schematic diagram for an STS-aided WSN employing M transmit sensors ($m \triangleq \{1, \ldots, M\}$) and N receive antennas at the DFC over T symbol duration over L_f space-time (ST) blocks in each transmission frame [11].

A. System Model

The sensors take a local decision autonomously based on a binary hypothesis test, $\mathcal{H}_0$ or $\mathcal{H}_1$, concerning absence and presence of a target of interest respectively. The local decision taken by the mth sensor is first mapped to a binary phase shift keyed (BPSK) symbol, $x_m^l \triangleq \{+1, -1\}$ multiplied by a T -length dispersion vector, $\mathbf{a}_{q,m}^l$, transmitted by the mth sensor in the lth time-slot to yield, $\mathbf{s}_m^l = x_m^l \mathbf{a}_{q,m}^l \in \mathbb{C}^{1\times T}$ for $(l = 1, 2, \ldots, L_f)$. Here, $\mathbf{a}_{q,m}^l = [a_{q,m,1}^l, a_{q,m,2}^l, \ldots, a_{q,m,T}^l] \in \mathbb{C}^{1\times T}$ is the mth row of the qth ST dispersion matrix $\mathbf{A}_q^l = [\mathbf{a}_{q,1}^l, \mathbf{a}_{q,2}^l, \ldots, \mathbf{a}_{q,M}^l]^t \in \mathbb{C}^{M\times T}$ selected out of the Q ST matrices taken from the set $\{\mathbf{A}_q^l\}_{q=1}^Q$. The encoded set of sensor decisions $\mathbf{S}^l \in \mathbb{C}^{M\times T} \triangleq \left[\mathbf{s}_1^l, \mathbf{s}_2^l, \ldots, \mathbf{s}_M^l\right]^t = \left[x_1^l \mathbf{a}_{q,1}^l, x_2^l \mathbf{a}_{q,2}^l, \ldots, x_M^l \mathbf{a}_{q,M}^l\right]^t$ must include ST dispersion matrices (DMs) that satisfy the power constraint of $Tr({\mathbf{A}_q^l}^H \mathbf{A}_q^l) = T \ \forall \ q$ to ensure unity energy over each ST block. For ease of representation, we employ parametric system definition in terms of (M, N, T, Q) for any STS-aided WSN. It is to be noted here that STS includes the space spreading (SS) or spatial modulation-like [12] arrangement $(M, N, 1, Q = M)$ as a special case.

Example Scenario: Let us assume the DMs to be 4×4 matrices, where a single sensor out of 4 sensors over one out of 4 ST blocks is activated. Hence we have $T = M = 4$ with $Q = 1$ DM activated out of $Q = 4$, such that the activated DM, $\mathbf{A}_q^l = [1\ 0\ 0\ 0; 0\ 1\ 0\ 0; 0\ 0\ 1\ 0; 0\ 0\ 0\ 1]$. The mapping of the sensor decisions on the chosen DM can be exhibited as, Time'1', Sensor'1' $\rightarrow \mathbf{s}_1^l = [x_1^l\ 0\ 0\ 0]$, Time'2', Sensor'2' $\rightarrow \mathbf{s}_2^l = [0\ x_2^l\ 0\ 0]$, Time'3', Sensor'3' $\rightarrow \mathbf{s}_3^l = [0\ 0\ x_3^l\ 0]$ and Time'4', Sensor'4' $\rightarrow \mathbf{s}_4^l = [0\ 0\ 0\ x_4^l]$ to create the encoded set of sensor decision,

$$\mathbf{S}^l \triangleq [\mathbf{s}_1^l\ \mathbf{s}_2^l\ \mathbf{s}_3^l\ \mathbf{s}_4^l]^t = \begin{bmatrix} x_1^l & 0 & 0 & 0 \\ 0 & x_2^l & 0 & 0 \\ 0 & 0 & x_3^l & 0 \\ 0 & 0 & 0 & x_4^l \end{bmatrix}$$

B. Transmission Signal Model

Following the sensor decision mapping, the generic $N \times T$ ($n \triangleq \{1, \ldots, N\}$, $t \triangleq \{1, \ldots, T\}$) discrete-time received signal matrix at the DFC is denoted by,

$$\mathbf{Y}^l = \sqrt{\rho^l}\mathbf{G}^l\mathbf{S}^l + \mathbf{w}^l \triangleq \left[\mathbf{y}_1^l, \mathbf{y}_2^l, \ldots, \mathbf{y}_N^l\right]^t \in \mathbb{C}^{N\times T} \quad (1)$$

where $\mathbf{Y}^l$, $\mathbf{G}^l \in \mathbb{C}^{N\times M}$ and $\mathbf{w}^l \sim \mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_{N\times T}, {\sigma_w^l}^2 \mathbf{I}_{N\times T}\right)$[1,2] are the received signal vector, the channel matrix and the noise vector respectively. The constant ρ^l denotes the energy spent by any of the sensors during the reporting phase and $\mathbf{G}^l$ includes all the samples of the channel impulse response (CIR) between the sensors and the DFC over the lth ST block.

C. Channel Model

The generic channel coefficient vector $g_{n,m}^l$ is expressed as $g_{n,m}^l = \sqrt{\lambda_m} h_{n,m}^l$, where λ_m models the geometric attenuation and shadow fading and remains constant over n and l. Based on these assumptions, we have $\mathbf{G}^l = \mathbf{H}^l\sqrt{\mathbf{D}}$ where $\mathbf{G}^l \in \mathbb{C}^{N\times M}$ denotes the matrix of the generic channel coefficients, $\mathbf{H}^l \in \mathbb{C}^{N\times M}$ denotes the matrix of the fast fading coefficients and $\mathbf{D} \in \mathbb{C}^{M\times M}$ is a diagonal matrix with $d_{m,m} = \lambda_m$.

D. Reception Signal Model

At the receiver, i.e. the DFC, by applying the vectorial stacking operation $vec()$ to the received signal block $\mathbf{Y}^l$ at the ST de-mapper, we arrive at the linearized equivalent received signal model formulated as,

$$\hat{\mathbf{Y}}^l = \sqrt{\rho^l}\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l\hat{\mathbf{K}}^l + \hat{\mathbf{w}}^l = vec(\mathbf{Y}^l) \in \mathbb{C}^{NT\times 1} \quad (2)$$

where $\hat{\mathbf{G}}^l = \mathbf{I}_g \otimes \mathbf{G}^l \in \mathbb{C}^{NT\times MT}$ (where $\mathbf{I}_g = \mathbf{I}_{T\times T}$) is the estimated channel matrix [3] and $\otimes$ is the Kronecker product, $\hat{\mathbf{w}}^l = vec(\mathbf{w}^l) \in \mathbb{C}^{NT\times 1}$, $\hat{\mathcal{A}}^l = \mathbf{I}_a \otimes \mathbf{A}^l \in \mathbb{C}^{MT\times MQ}$ (where $\mathbf{I}_a = \mathbf{I}_{M\times M}$ and $\mathbf{A}^l \triangleq \left[\mathbf{A}_1^l, \mathbf{A}_2^l, \ldots, \mathbf{A}_Q^l\right]^t$) and $\hat{\mathbf{K}}^l = vec(\mathbf{K}^l) \triangleq vec\left([\mathbf{0}, \ldots, \mathbf{0}, \mathbf{x}^l, \mathbf{0}, \ldots, \mathbf{0}]\right) \in \mathbb{C}^{MQ\times 1}$ and $\mathbf{x}^l = [x_1^l, x_2^l, \ldots, x_M^l]^t \in \mathbb{C}^{M\times 1}$ is the transmit vector.

E. Decision Fusion

The DFC is in charge of providing a reliable decision (i.e. $\mathcal{H}^1, \ldots, \mathcal{H}^{L_f}$) on the basis of the superimposed received decisions taken locally by the sensors independently over each ST block (i.e. $\hat{\mathbf{Y}}^1, \ldots, \hat{\mathbf{Y}}^{L_f}$) by employing optimum or sub-optimum fusion rules. In this letter, we will consider and compare two types of fusion rules to arrive at a reliable choice depending on the scenario.

The first set of rules (Decode-and-fuse) aims at concluding on the presence or absence of the target directly from the

[1]The noise vector also accounts for different levels of channel state information (CSI) estimation error, where the estimated channel on the receiver side is contaminated by additive Gaussian noise.

[2]$(\cdot)^t$ denotes transpose, $\mathbb{R}(\cdot)$, $\mathbb{E}\{\cdot\}$, $\mathbb{V}\{\cdot\}$, $\angle(\cdot)$, $(\cdot)^\dagger$, $||\cdot||$, $p(\cdot)$ and $P(\cdot)$ represents real-part, expectation, variance, phase, conjugate transpose, Frobenius norm operators, probability density function and probability mass function respectively; $\mathcal{N}(\mu, \Sigma)$ denote normal distribution with mean μ and co-variance matrix Σ.

[3]The DFC estimates the CSI, where half of the coherence interval of $2T$ is used for training to estimate the channel and establish the frequency and timing synchronization. It is worth mentioning that the linearized sensor-DFC system model contains M non-zero symbol components in $\hat{\mathbf{K}}^l$ given by $\mathbf{x}^l$.

received signal without processing the transmit signal; the optimum (opt) test statistics for which is given by,

$$\mathbf{\Gamma}^l_{\text{opt}} \approx \ln\left[\frac{\sum_{\mathbf{x}^l}\exp\left(-\frac{\left\|\hat{\mathbf{Y}}^l-\sqrt{\rho^l}(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q)\mathbf{x}^l\right\|^2}{\sigma^2_w}\right)P\left(\mathbf{x}^l \mid \mathcal{H}^l_1\right)}{\sum_{\mathbf{x}^l}\exp\left(-\frac{\|\hat{\mathbf{Y}}^2-\sqrt{\rho^l}(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q)\mathbf{x}^l||^2}{\sigma^l_w}\right)P\left(\mathbf{x}^l \mid \mathcal{H}^l_0\right)}\right] \tag{3}$$

assuming conditional independence of $\hat{\mathbf{Y}}^l$ from $\mathcal{H}^l_j$, given $\mathbf{x}^l$, and $\hat{\mathbf{G}}^l \in \mathbb{C}^{NT\times MT}$, $\hat{\mathcal{A}}^l_q \in \mathbb{C}^{MT\times M}$ and $\mathbf{x}^l \in \mathbb{C}^{M\times 1}$. The second set of fusion rules (Decode-then-fuse) firstly estimates the transmit signal from the received signal and then arrives at a global decision based on the estimated transmit signal vector $\bar{\mathbf{x}}^l$. The test statistics using Chair-Varshney (CV) rule for noiseless channel is given by,

$$\mathbf{\Gamma}^l_{\text{CV}} = \xi^l \ln\left(\frac{P(\mathbf{x}^l|\mathcal{H}_1)}{P(\mathbf{x}^l|\mathcal{H}_0)}\right) + \left(1-\xi^l\right)\ln\left(\frac{1-P(\mathbf{x}^l|\mathcal{H}_1)}{1-P(\mathbf{x}^l|\mathcal{H}_0)}\right) \tag{4}$$

where $\xi^l = \frac{\bar{\mathbf{x}}^l+1}{2}$. In this case, the symbol decoder block in Fig. 1 at the DFC computes $\bar{\mathbf{x}}^l$.

F. Fusion Rules

In the first group of fusion rules, we consider three sub-optimum rules, Maximal Ratio Combining (MRC), modified MRC (mMRC) and Widely Linear (WL) rules; the test statistics of which are given by (assuming identical sensor performances),

$$\begin{aligned}
\mathbf{\Gamma}^l_{\text{MRC}} &\propto \mathbb{R}\left\{\mathbf{1}^t_M\left(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q\right)^\dagger\hat{\mathbf{Y}}^l\right\}\\
\mathbf{\Gamma}^l_{\text{mMRC}} &\propto \mathbb{R}\left\{\mathbf{1}^t_M\left(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q\right)^\dagger(\mathbf{D}^l_g)^{-1}\hat{\mathbf{Y}}^l\right\}\\
\mathbf{\Gamma}^l_{i,\text{WL}} &\triangleq \left(\frac{\mathbf{\Sigma}^{-1}_{\underline{\hat{\mathbf{Y}}}^l|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_i}\underline{\hat{\mathbf{G}}^l}\hat{\mathcal{A}}^l_q\boldsymbol{\mu}^l_i}{\left\|\mathbf{\Sigma}^{-1}_{\underline{\hat{\mathbf{Y}}}^l|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_i}\underline{\hat{\mathbf{G}}^l}\hat{\mathcal{A}}^l_q\boldsymbol{\mu}^l_i\right\|}\right)^\dagger\underline{\hat{\mathbf{Y}}}^l
\end{aligned} \tag{5}$$

where $\mathbf{D}^l_g = \frac{1}{N}(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q)^\dagger(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q)$ is a diagonal matrix for $N \gg M$, $\mathbf{\Sigma}^{-1}_{\underline{\hat{\mathbf{Y}}}^l|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_i} = \left(\rho^l\underline{\hat{\mathbf{G}}^l}\hat{\mathcal{A}}^l_q\mathbf{\Sigma}_{\mathbf{x}^l|\mathcal{H}^l_i}(\underline{\hat{\mathbf{G}}^l}\hat{\mathcal{A}}^l_q)^\dagger + \sigma^2_{w,l}\mathbf{I}_{2N}\right)^4$ is chosen such that the deflection measure $\mathcal{D}_i(\cdot)$ is maximized, $\mathcal{D}_0(\cdot)$ correspond to the normal and and $\mathcal{D}_1(\cdot)$ corresponds to the modified deflection [13], $\boldsymbol{\mu}^l_i \triangleq 2\left[\left(P^l_{D,1}-P^l_{F,1}\right)\dots\left(P^l_{D,M}-P^l_{F,M}\right)\right]^t$, $\mathbf{P}^l_{D,m} \triangleq P\left(\hat{\mathbf{K}}^l_m = [0,\dots,0,x^l_m=1,0,\dots,0]^t|\mathcal{H}_1\right)$ and $\mathbf{P}^l_{F,m} \triangleq P\left(\hat{\mathbf{K}}^l_m = [0,\dots,0,x^l_m=1,0,\dots,0]^t|\mathcal{H}_0\right)$ respectively, the probabilities of detection and false alarm of the mth sensor on the lth ST block.

In the second group, we consider three different decoders to estimate $\bar{\mathbf{x}}^l$ from $\hat{\mathbf{Y}}^l$; the global decision $\hat{\mathcal{H}}$ is taken on the basis of $\bar{\mathbf{x}}^l$, which includes maximum likelihood (ML), zero forcing (ZF) and minimum mean squared error (MMSE) detector given by,

$$\begin{aligned}
\bar{\mathbf{x}}^l_{\text{ML}} &= \overset{\text{argmin}}{\mathbf{x}^l}\left\|\hat{\mathbf{Y}}^l-\sqrt{\rho^l}(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q)\mathbf{x}^l\right\|^2\\
\bar{\mathbf{x}}^l_{\text{MMSE}} &= \left(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q\right)^\dagger\left(\left(\mathbf{D}^l_g+\frac{{\sigma^l_w}^2}{\sqrt{\rho^l}}\mathbf{I}_M\right)^{-1}\right)^\dagger\hat{\mathbf{Y}}^l\\
\bar{\mathbf{x}}^l_{\text{ZF}} &= \left(\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q\right)^\dagger\left(\left(\mathbf{D}^l_g\right)^{-1}\right)^\dagger\hat{\mathbf{Y}}^l
\end{aligned} \tag{6}$$

Once $\bar{\mathbf{x}}^l$ is obtained, we plug it in the CV-rule to obtain the test statistics for CV-ML and CV-MMSE rules.

III. Performance Metrics

In this section, some insights on the probability of detection or false-alarm, design criteria for the DMs and the complexity associated with different fusion rules when employed in an STS-aided WSN are provided.

A. Performance Measures

Combining the decisions from all the M sensors independently over each ST block, we can arrive at the total probabilities $P^l_{D_0,\text{rule}}(\triangleq \Pr\{\Gamma^l_{\text{rule}} > \gamma^l|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_1\})$ and $P^l_{F_0,\text{rule}}(\triangleq \Pr\{\Gamma^l_{\text{rule}} > \gamma^l|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_0\})$ respectively for the presented network and each fusion rule, where Γ^l_{rule} is the generic test statistics and γ^l is the threshold with which the test statistics for each rule is compared. Assuming $\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_0\} \triangleq (2\ P^l_F-1)\mathbf{I}_M$ and $\mathbb{E}\{(\mathbf{x}^l-\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_0\})(\mathbf{x}^l-\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_0\})^t|\mathcal{H}^l_0\} \triangleq [1-(2\ P^l_F-1)^2]\mathbf{I}_M$, we can compute $P^l_{F_0,\text{rule}}$ for the formulated fusion rules as below. Alternatively, it is to be noted here that, we can also compute $P^l_{D_0,\text{rule}}$ by assuming, $\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_1\} \triangleq (2\ P^l_D-1)\mathbf{I}_M$ and $\mathbb{E}\{(\mathbf{x}^l-\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_1\})(\mathbf{x}^l-\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_1\})^t|\mathcal{H}^l_1\} \triangleq [1-(2\ P^l_D-1)^2]\mathbf{I}_M$.

Assuming $P(\mathbf{Y}^l|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_j)$ follow Gaussian mixture distribution, $\mathbf{\Gamma}^l_{j,\text{rule}}|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_j$ is also distributed according to Gaussian mixture model. Using Gaussian moment matching [14], we have, $\mathbf{\Gamma}^l_{j,\text{rule}}|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_j \overset{\text{approx}}{\sim} \mathcal{N}\left(\mathbb{E}\{\mathbf{\Gamma}^l_{j,\text{rule}}|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_j\},\mathbb{V}\{\mathbf{\Gamma}^l_{j,\text{rule}}|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathcal{H}^l_j\}\right)$. Under simplifying assumptions of $\mathbb{E}\{\mathbf{x}^l|\mathcal{H}^l_0\} \triangleq (2\ P^l_F-1)\mathbf{1}_M = \delta^l$, we arrive at the low signal-to-interference-plus-noise ratio (SINR) approximation for $P^l_{F_0,\text{rule}}$ as,

$$P^l_{F_0,\text{rule}} = \lim_{N\to\infty}\mathcal{Q}\left(\frac{\gamma^l-\frac{\mathbb{E}\{\mathbf{\Gamma}^l_{j,\text{rule}}|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathbf{x}^l\}}{\sqrt{\mathbb{V}\{\mathbf{\Gamma}^l_{j,\text{rule}}|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathbf{x}^l\}}}}{\sqrt{1/2((1-{\delta^l}^2)M+{\sigma^l_w}^2)}}\right). \tag{7}$$

where $\mathcal{Q}(\cdot)$ is used to denote the complementary cumulative distribution function (CCDF). Since at low SINR, the components of the Gaussian mixture gets concentrated within a certain region, we need to evaluate the mean and variance of $\mathbf{\Gamma}^l_{j,\text{rule}}$ for each fusion rule summarized in Table I with $\mathbf{V}^l_j \triangleq \mathbf{I}_M - \left(\frac{1+{\sigma^l_w}^2}{2\mathbf{D}^l_g\rho^l N\sqrt{N}\mathbf{\Sigma}^{-1}_{\mathbf{x}^l|\mathcal{H}^l_j}}\right)^{-1}$ and $\mathbf{B}^l_g = \left(\mathbf{D}^l_g+\frac{{\sigma^l_w}^2}{\sqrt{\rho^l}}\mathbf{I}_M\right)^{-1}$. Putting the values from Table I in (7), we can arrive at $P^l_{F_0,\text{rule}}$ for each fusion rule.

B. Design Criteria for DMs

For the performance analysis, we have chosen $Q = M = T$. Accordingly, we choose our DM based on random search following the steps:

- Randomly generate Q unitary dispersion matrix set $\{\mathbf{A}^l_q\}^Q_{q=1}$ using Gaussian distribution, such that $Tr({\mathbf{A}^l_q}^H\mathbf{A}^l_q) = T\ \forall\ q$.

[4] $\underline{\mathbf{u}}$ (resp. $\underline{\mathbf{U}}$) denotes the augmented vector (resp. matrix) of $\mathbf{u}$ (resp. $\mathbf{U}$) i.e., $\underline{\mathbf{u}} \triangleq [\mathbf{u}^t\ \ \mathbf{u}^\dagger]^t$ (resp. $\underline{\mathbf{U}} \triangleq [\mathbf{U}^t\ \ \mathbf{U}^\dagger]^t$)

TABLE I

MEAN AND VARIANCE OF $\mathbf{\Gamma}^l_{j,\text{RULE}}$ FOR THE CONSIDERED FUSION RULES

Fusion Rule	$\mathbb{E}\{\mathbf{\Gamma}^l_{j,\text{rule}}\|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathbf{x}^l\}$	$\mathbb{V}\{\mathbf{\Gamma}^l_{j,\text{rule}}\|\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q,\mathbf{x}^l\}$
MRC	$\sqrt{\rho^l}\mathbb{R}\{(\mathbf{1}_M)^t\mathbf{D}^l_g\mathbf{x}^l\}$	$\frac{{\sigma^l_w}^2}{2N}\|\|(\mathbf{1}_M)^t\mathbf{D}^l_g\mathbf{1}_M\|\|$
mMRC	$\sqrt{\rho^l}\mathbb{R}\{(\mathbf{1}_M)^t\mathbf{x}^l\}$	$\frac{{\sigma^l_w}^2}{2N}\|\|(\mathbf{1}_M)^t\mathbf{D}^l_g\mathbf{1}_M\|\|$
WL	$\sqrt{\rho^l}(\mu^l_j)^t\mathbf{V}^l_j\mathbf{D}^l_g\mathbf{x}^l$	$\frac{{\sigma^l_w}^2}{2N}\|\|\left(\mu^l_j\mathbf{V}^l_j\right)^t\mathbf{D}^l_g\mu^l_j\mathbf{V}^l_j\|\|$
CV-ML	$\sqrt{\rho^l}\mathbf{D}^l_g\mathbf{x}^l$	$\frac{{\sigma^l_w}^2}{2N}\|\|\mathbf{D}^l_g\|\|$
CV-MMSE	$\sqrt{\rho^l}\mathbf{B}^l_g\mathbf{D}^l_g\mathbf{x}^l$	$\frac{{\sigma^l_w}^2}{2N}\|\|\left(\mathbf{B}^l_g\right)^t\mathbf{D}^l_g\mathbf{B}^l_g\|\|$
CV-ZF	$\sqrt{\rho^l}\mathbf{x}^l$	$\frac{{\sigma^l_w}^2}{2N}\|\|(\mathbf{D}^l_g)^{-1}\|\|$

TABLE II

COMPLEXITY ANALYSIS FOR THE CONSIDERED FUSION RULES

Fusion Rule	Complexity for each realization of $\mathbf{x}^l$
Optimum	$\mathcal{O}(NMQT2^{MQ})$
(m)MRC, WL	$\mathcal{O}(NT)$
CV-ML, CV-MMSE, CV-ZF	$\mathcal{O}(NMQT)$

- The matrix generation is repeated for 10 times.
- For each of the 10 sets of $\{\mathbf{A}^l_q\}_{q=1}^Q$, calculate the corresponding $P^l_{D_0,\text{Opt}}((\gamma^l,\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q))$

$$= 1 - \lim_{N\to\infty}\mathcal{Q}\left(\frac{\gamma^l - \frac{N\sqrt{2\rho^l}\mathbf{D}^l_g\{P(\mathbf{x}^l|\mathcal{H}^l_1)/P(\mathbf{x}^l|\mathcal{H}^l_0)\}}{{\sigma^l_w}^2||\mathbf{D}^l_g\{P(\mathbf{x}^l|\mathcal{H}^l_1)/P(\mathbf{x}^l|\mathcal{H}^l_0)\}||}}{\sqrt{1/2((1-{\delta^l}^2)M+{\sigma^l_w}^2)}}\right). \quad (8)$$

- Out of the sets of $\{\mathbf{A}^l_q\}_{q=1}^Q$, the set exhibiting the maximum $P^l_{D_0,\text{Opt}}((\gamma^l,\hat{\mathbf{G}}^l\hat{\mathcal{A}}^l_q))$ is chosen for simulating performance.

C. Complexity Analysis

In Table II, we compare the computational complexity of the formulated rules, for each new $\mathbf{x}^l$ transmitted, in terms of the Landau notation $\mathcal{O}(\cdot)$, i.e. the order of complexity.

IV. PERFORMANCE ANALYSIS

We simulate performance of a (M,N,T,Q) STS-aided WSN, where M sensors are randomly deployed and uniformly distributed in a circular annulus around the DFC with radii $\phi_{\text{max}} = 1000$ m and $\phi_{\text{min}} = 100$ m. We assume $Q = T = M$ to exploit full diversity order and enhance opportunistic network throughput. The ST spreaded local decisions of the sensors are transmitted over a log-normal shadowed and Rayleigh block faded channel, such that $\mathbf{h}^l_{n,m} \sim \mathcal{N}_{\mathbb{C}}(0,\text{diag}(\mathcal{B}^l_m))$, where $\lambda^l_m = \psi_m(\frac{\phi_{\text{min}}}{\phi_m})^\eta$, $\mathcal{B}^l_m = \left(\beta^l_m(0),\ldots,\beta^l_m(T-1)\right)^t$ is the channel power delay profile with $\sum_{\tau=0}^{T-1}\beta^l_m(\tau) = 1$, $10\log_{10}(\psi_m) \sim \mathcal{N}(\mu_\lambda \text{ dB}, \sigma^2_\lambda \text{ dB})$, η is the pathloss exponent and ϕ_m is the distance of the mth sensor to the DFC. We also assume $\rho^l = 1/\sqrt{N}$ and independently and identically distributed (iid) decisions with $(P^l_D, P^l_F) = (0.5, 0.05)$.

In Fig. 2 and Fig. 3, we present the Receiver Operating Characteristics (ROC) of all the fusion rules for two different configurations of WSNs, a) fully-loaded MIMO set-up ($M = 8$, $N = 8$) and b) virtual mMIMO set-up ($M = 10$, $N = 100$). We simulate performance of the formulated fusion rules over a MAC with pathloss exponent, η of 2, experiencing moderate shadowing, $(\mu_\lambda, \sigma_\lambda) = (4, 2)$ dB.

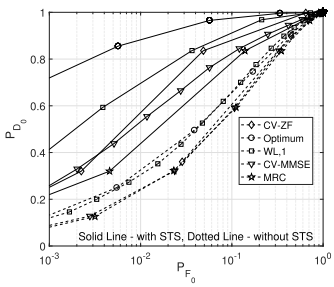


Fig. 2. Comparative ROC (P_{D_0} v/s P_{F_0}) of different fusion rules in $(8,8,8,8)$ STS-aided WSNs with that in WSNs without STS-aided decision transmission over a fixed SINR of 15 dB. Both the sensors and DFC are deployed in a variety of indoor environments.

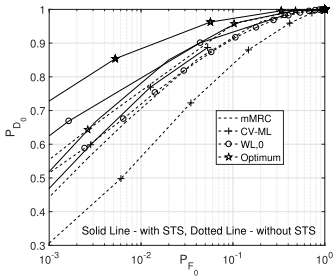


Fig. 3. Comparative ROC (P_{D_0} v/s P_{F_0}) of different fusion rules in $(10,100,10,10)$ STS-aided WSNs with that in WSNs without STS-aided decision transmission over a fixed SINR of 15 dB. Both the sensors and DFC are deployed in a variety of indoor environments.

The above-mentioned parameters are representative of a variety of indoor environments. For the two network set-ups, STS aided sensor decision transmission offers significant improvement in performance over that without STS. Each fusion rule gains in performance from a minimum of 3 times (MRC) to a maximum of 6 times (Opt.) in case of fully-loaded MIMO, and a maximum of 8-9 times (Opt.) to a minimum of 3-4 times (CV-ML) in case of virtual mMIMO set-up. For both the set-ups, MRC (mMRC for mMIMO case) and CV-ML perform worst respectively, as corroborated in [4], [10].

In Fig. 4, we establish the validity of our derived expressions for performance evaluation of the formulated fusion rules in Section III-A, by comparing simulated and analytical performances. The analytical performances match closely with and in many cases, almost identically to the simulated performances. The reason can be attributed to the fact that the channel samples for the analytical results are generated using the same channel model used for generating the simulation results.

In Fig. 5, we plot P_{D_0} of the presented fusion rules as a function of N under $P_{F_0} \leq 0.01$; we depict the case $M = 4$. Performance of all fusion rules improves with the increase in N, except MRC, however reaches saturation depending on the SINR and the chosen fusion rule. Some rules like CV-MMSE, and CV-ZF ($N > 40$) proceeds to saturation faster than other rules like WL,0 ($N > 90$). It is also evident that MRC performs worse than any other fusion rule, as MRC does not exploit STS aided local sensor performance at the decoding stage like WL, CV-MMSE or CV-ZF. Indeed, the probability of detection with MRC is dependent only on the channel statistics.

In Fig. 6, we demonstrate P_{D_0} of the presented fusion rules as a function of $(SINR)_{dB}$, where the SINR measure includes channel noise, varying levels of CSI estimation errors

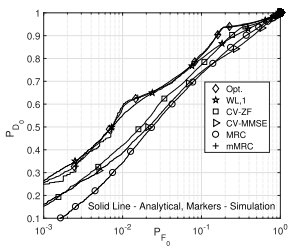


Fig. 4. Comparative analytical and simulation performance for different fusion rules in $(4, 20, 4, 4)$ STS-aided WSNs over a fixed SINR of 15 dB. Both the sensors and DFC are deployed in a variety of indoor environments.

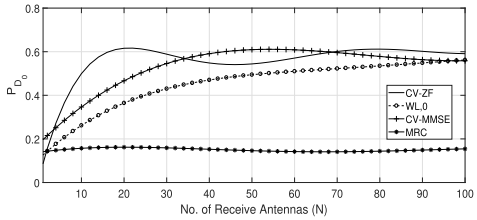


Fig. 5. Variation in probability of detection of different fusion rules with changing N (P_{D_0} v/s N) for $(4, N, 8, 8)$ STS-aided WSNs over a fixed SINR of 15 dB and $P^l_{F_0} = 0.01$. The sensors are deployed in a tunnel-like environment.

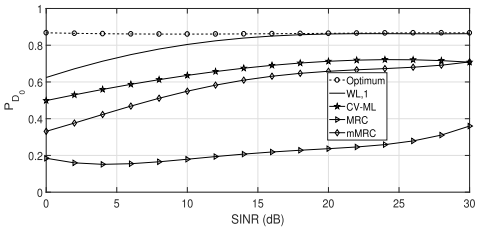


Fig. 6. Comparative probability of detection performance (P_{D_0} v/s SINR(dB)) for different fusion rules in an $(8, 32, 8, 8)$ STS-aided WSN where outdoor sensors communicate with an indoor DFC.

and interference. In this case, $N = 3, M = Q = T = 8$ and $\eta = 5$ and $\mu_\lambda = 4$ dB, a representative condition of outdoor sensors communicating with indoor DFC. CV-MMSE, CV-ZF and WL,1 rules approach the optimal performance at moderate to high SINRs. However, MRC, mMRC and CV-ML rules fail to achieve optimal performance even at high SINRs, as opposed to the observations made in [15]. It has been demonstrated in [16], that for $T > 1$, diversity increases but at the cost of reliability for STSK modulated systems. For $Q > 1$, throughput increases but at the cost of degraded bit error rate (BER). Here, we have chosen $T = Q = M$ for STS thereby sacrificing reliability of system knowledge (like CSI statistics, statistics of sensor decision vectors) and lower probability of error for the sake of gain in diversity and network throughput. It can be broadly concluded that CV-ML performs poorly in any network scenario and propagation condition as the CV-ML statistics is dependent on the channel SINR which is kept fixed for Figs. 2, 3, and 5.

V. Conclusion

In order to strike a flexible balance between interference minimization and energy efficiency in massive WSNs for IoT applications, we conceive the novel idea of ST spreading the local sensor decisions before transmission in a WSN. The resultant network will not only benefit from improvement in opportunistic throughput but also from ISCI and ISEI free transmission in a densely deployed scenario. The STS scheme used can be modified depending on the chosen Q and T to include Spatial Modulation (SM) [12], Space-Shift Keying (SSK) [17] and Space-Time Shift Keying (STSK) [16] arrangements. However, using different values for Q, T, M will involve multiple information symbols for carrying sensing decisions, an interesting generalization which we leave for our future work. Our presented simulation results demonstrate the potential of STS-aided WSN in outperforming the conventional MIMO and mMIMO based WSN arrangements. Motivated by this observation, in future, we plan to extend our results under different conditions of dispersion matrix optimization, multi-slot decision transmission and correlated sensor observations in sensing performance.

References

[1] A. Argyriou and Ö. Alay, "Distributed estimation in wireless sensor networks with an interference canceling fusion center," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 2205–2214, Nov. 2015.

[2] V. Rao, R. Prasad, T. Prabhakar, C. Sarkar, M. Koppal, and I. Niemegeers, "Understanding and improving the performance of constructive interference using destructive interference in WSNs," *IEEE/ACM Trans. Netw.*, vol. 27, no. 2, pp. 505–517, Feb. 2019.

[3] I. Dey, M. M. Butt, and N. Marchetti, "Throughput analysis for virtual MIMO WSNs over measured MIMO channels," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 297–299, Oct. 2019.

[4] D. Ciuonzo, G. Romano, and P. S. Rossi, "Channel-aware decision fusion in distributed MIMO wireless sensor networks: Decode-and-fuse vs. Decode-then-fuse," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2976–2985, Aug. 2012.

[5] A. Chawla, A. Patel, A. K. Jagannatham, and P. K. Varshney, "Distributed detection in massive MIMO wireless sensor networks under perfect and imperfect CSI," *IEEE Trans. Signal Process.*, vol. 67, no. 15, pp. 4055–4068, Jun. 2019.

[6] S. Althunibat and R. Mesleh, "Index modulation for cluster-based wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6943–6950, Aug. 2018.

[7] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Sel. Areas Commun.*, vol. SAC-16, no. 8, pp. 1451–1458, Oct. 1988.

[8] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *Proc. URSI Int. Symp. Signals, Syst., Electron.*, Oct. 1998, pp. 295–300.

[9] B. Hassibi and B. M. Hochwald, "High-rate codes that are linear in space and time," *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1804–1824, Jul. 2002.

[10] D. Ciuonzo, P. S. Rossi, and S. Dey, "Massive MIMO channel-aware decision fusion," *IEEE Trans. Signal Process.*, vol. 63, no. 3, pp. 604–619, Feb. 2015.

[11] I. Dey *et al.*, "Decision fusion in space-time spreading aided distributed MIMO WSNs," 2020, *arXiv:2005.08011*. [Online]. Available: http://arxiv.org/abs/2005.08011

[12] R. Y. Mesleh, H. Haas, S. Sinanovic, C. W. Ahn, and S. Yun, "Spatial modulation," *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2228–2241, Jul. 2008.

[13] Z. Quan, S. Cui, and A. H. Sayed, "Optimal linear cooperation for spectrum sensing in cognitive radio networks," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 28–40, Feb. 2008.

[14] Y. Bar-Shalom, T. Kirubarajan, and X. R. Li, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. Hoboken, NJ, USA: Wiley, 2004.

[15] I. Dey, P. S. Rossi, M. M. Butt, and N. Marchetti, "Virtual MIMO wireless sensor networks: Propagation measurements and fusion performance," *IEEE Trans. Antennas Propag.*, vol. 67, no. 8, pp. 5555–5568, May 2019.

[16] S. Sugiura, S. Chen, and L. Hanzo, "Coherent and differential space-time shift keying: A dispersion matrix approach," *IEEE Trans. Commun.*, vol. 58, no. 11, pp. 3219–3230, Nov. 2010.

[17] J. Jeganathan, A. Ghrayeb, L. Szczecinski, and A. Ceron, "Space shift keying modulation for MIMO channels," *IEEE Trans. Wireless Commun.*, vol. 8, no. 7, pp. 3692–3703, Jul. 2009.

Fault detection and classification in Industrial IoT in case of missing sensor data

Fault detection and classification in Industrial IoT in case of missing sensor data

Merim Dzaferagic, Nicola Marchetti, Irene Macaluso

***Abstract*—This paper addresses the issue of reliability in Industrial Internet of Things (IIoT) in case of missing sensors measurements due to network or hardware problems. We propose to support the fault detection and classification modules, which are the two critical components of a monitoring system for IIoT, with a generative model. The latter is responsible of imputing missing sensor measurements so that the monitoring system performance is robust to missing data. In particular, we adopt Generative Adversarial Networks (GANs) to generate missing sensor measurements and we propose to fine-tune the training of the GAN based on the impact that the generated data have on the fault detection and classification modules. We conduct a thorough evaluation of the proposed approach using the extended Tennessee Eastman Process dataset. Results show that the GAN-imputed data mitigate the impact on the fault detection and classification even in the case of persistently missing measurements from sensors that are critical for the correct functioning of the monitoring system.**

***Index Terms*—Generative Adversarial Network, Industrial IoT (IIoT), data imputation, fault detection, fault classification.**

I. Introduction

The Internet of Things (IoT) is a computing paradigm that relies on ubiquitous connection to the Internet, where common objects are turned into connected devices. Up to trillions of smart objects are being deployed, that are capable of sensing their surroundings, transmit and process acquired data, and then in turn feedback relevant information to the environment. A subset of IoT, the Industrial Internet of Things (IIoT) encompasses machine-to-machine (M2M) and industrial communication technologies with applications in the automation sector. IIoT paves the way for better understanding of the manufacturing process, with positive repercussions on the efficiency and sustainability of the production system [1], [2].

In the era of IoT big data, the integration of cloud computing technologies and cyber-physical systems enables the full potential of Industry 4.0 to be harvested in manufacturing processes, with a multitude of sensors being installed around the industrial operating environment and equipment. The networked sensors would continuously send monitoring data, allowing for proactive maintenance to take place, leading to a reduction in the unplanned downtime via data analysis techniques [3], [4], [5], [6], [7].

The work proposed in this article is based on the FIREMAN project funded by CHIST-ERA, which focuses on modelling and analysing IIoT networks based on specific machine learning algorithms capable of detecting rare events in industrial setups in an ultra-reliable way [8]. An important aspect in assuring such ultra-reliability in the IIoT is how to guarantee we have a functioning system in place, even in case some of the measurements are missing due to network or hardware issues. In fact values are often missing from the collected sensor data, and the related issue of missing value imputation becomes then very important. For example, high-frequency data collection can result in large gaps in the data and if the network stops working, all the measurements collected during the network downtime will be missing [3]. Other possible reasons behind missing data are: faulty sensors producing intermittent readings, loss of data during wireless communication owing to packet loss or to interference in the communication medium, or data removed purposely by attackers with malicious intentions during sensing, processing, storing or communication. A related research challenge is to impute the missing values, to enable the data to be analyzed while ensuring that the imputed values are as close as possible to the true values. What complicates things with regard to the imputation of missing data in IoT, is that the data to be collected in such systems is diverse, and the techniques developed must therefore provide a high level of confidence for different types of applications, besides the need to be robust to the increase in the scale of IoT (and IIoT) deployments. Furthermore, techniques must be lightweight to be able to fulfil real-time IoT application requirements [9].

All the approaches reported to date in the literature focus on either data imputation, anomaly detection or fault classification for an industrial process. In this paper we instead propose a framework that unifies all three techniques, allowing us to optimize each of them in a way that results in the best overall performance of the monitoring system. We propose a data-driven decomposition of the process to monitor the various indicators of the health of a machine/component or an entire industrial process. Instead of proposing a new tailored solution to collect, communicate and process data in an industrial environment, we focus on the detection and classification of system failures based on a dataset with missing values, investigating in particular the impact of missing data on the monitoring system in an industrial setting. This is a very important issue to tackle, as small reconstruction errors of missing sensor data could greatly affect the capability of the monitoring system. The data imputation module in our framework relies on a generative adversarial network (GAN) model that learns the correlation between the data from the input layer to replace missing sensor measurements. The GAN was optimized by validating its performance based on the effect of the imputed measurements on the fault identification and detection modules, which ultimately constitute the two

M. Dzaferagic, N. Marchetti, and I. Macaluso are with CONNECT centre, Trinity College Dublin, Ireland.

essential tasks performed by the monitoring system. As we will discuss later, the GAN-imputed data mitigate the loss on these two modules even in the case of persistently missing measurements from sensors that are critical for the correct functioning of the monitoring system.

Section II reports an account of the relevant literature, while also positioning our work highlighting the main differences and advantages of our approach. Section III describes the proposed framework, describing the adopted fault detection, fault classification and GAN-based missing data imputation techniques. Section IV analyses the performance of our techniques in terms of recall and precision, and showing the impact the proposed data imputation has on both metrics. Section V concludes the paper and outlines some possible promising directions for future research.

II. State of the art

IoT is based on the idea of connecting the physical and the digital worlds [10]. Initially, Radio-Frequency IDentification (RFID) was the main technology in this space, allowing microchips to identify themselves to a reader wirelessly [11]. Today, IoT applications have moved from the simple RFIDs, by integrating diverse sensing, computing and communication technologies. An IoT platform provides services and features like: node management, connectivity and network management, data management, analysis and processing, security, access control and interfacing [12]. Domains like transportation, healthcare, industrial automation and education are all affected by the fast development of these platforms. Different domains face different problems related to the deployment of IoT solutions (e.g. low latency communication, high reliability, massive data transfer, energy efficiency). Hence, different IoT platforms are needed to run specific applications.

In an industrial environment, the detection and prediction of anomalies are important for both economic and security reasons. Difficulties related to these tasks originate from the fact that anomalies are rare events within datasets, making it difficult to apply most of the existing algorithms which result in either false alarms or misdetections [13], [14]. The authors of [8] made an attempt at providing a general framework to model a wide range of cases based on the advances in IIoT networks and Machine Learning (ML) algorithms. Similarly, the authors of [15], [16] describe several deployed solutions of cyber-physical systems in an industrial environment. The most promising solutions include those presented in [17], [18], [19]. Unlike the work in the above-mentioned papers, instead of proposing a new tailored solution to collect, communicate and process data in an industrial environment, we focus on the detection and classification of system failures based on a dataset with missing values.

Fault detection in an industrial environment has always been a challenging task [20], [21], [22], [23], [24]. Due to the issues related to interoperability and communication between different devices, collecting a large dataset in such an environment is not an easy task. However, even if we assume that the dataset was collected, different problems arise when using such a dataset for anomaly detection (e.g. unbalanced dataset, noise in the measurements, missing data). Similar to the work proposed by the authors of [25], [3], we also investigate the impact of missing data on the detection of rare events in an industrial setting. The authors of [25] propose a sensor data reconstruction scheme that exploits the hidden data dynamics to accurately estimate the missing measurements. In [3], the authors focus on missing data imputation for large gaps in univariate time-series data and propose an iterative framework, using multiple segmented gap iteration to provide the most appropriate values. All the approaches mentioned above focus on either data imputation, anomaly detection or fault classification for an industrial process. We, on the other hand, propose a framework that unifies all three techniques, allowing us to optimize each of them in a way that results in the best overall performance. Indeed we train the imputation model to minimize the false alarm rate of the anomaly detection model and the classification error of the fault identification model, so that the monitoring system performance is robust to missing data.

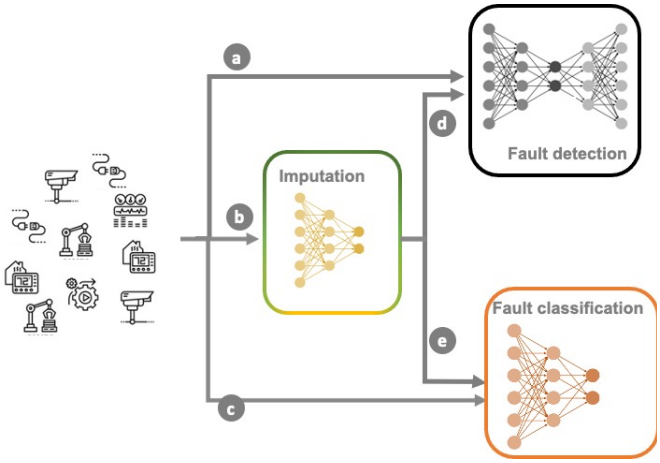
III. Framework

In this work we propose a data-driven decomposition of the process to monitor the various indicators of the health of a machine/component or an entire industrial process. Since faults account only for a very small fraction of the data collected in an IIoT scenario, we separately address the problem of fault or anomaly detection and the problem of fault classification. The latter is only triggered in case an anomaly is detected, as shown by the data flow in Figure 1. The fault detection, for which we employ an autoencoder as discussed in Section III-A, can be trained using data collected during normal operation. The fault classification, discussed in Section III-B, can be deployed as soon as enough labeled data related to faults become available. Finally, the imputation module, described in Section III-C, requires both faulty and non-faulty data to replace missing sensors' data so that the monitoring process can continue without disruption even if some measurements are not received.

Figure 1 shows the overall system and its data flow. In case all sensors measurements are received, the fault detection procedure is always activated, while the fault classification is performed only if an anomaly is detected (path $a-c$ in Figure 1). In case of missing measurements, the data imputation module is first triggered, followed by the fault detection and eventually by the fault classification if an anomaly is detected.

A. Fault detection

We built the fault detection module using an autoencoder that receives as input the N sensor measurements used to monitor the industrial process. The outputs of the autoencoder are the reconstructed values from the input. By minimizing the Root Mean Square Error (RMSE) of the reconstructed values, the model learns a representation for the input data and filters the noise. By training the model on fault-free data only, the autoencoder will learn the patterns of normal operating conditions. This way, when faulty data will be inputed the resulting reconstruction error should be larger than



Possible Data Path	Scenario
a	No missing data; no anomaly detected
a c	No missing data; anomaly detected; anomaly classification
b d	Missing data; no anomaly detected
b d e	Missing data; anomaly detected; anomaly classification

Fig. 1: Data flow within the monitoring system. If all measurements are received the fault detection is activated, followed by the fault classification in the event of an anomaly (paths a and $a-c$). If some measurements are not received, the fault detection and identification are preceded by the data imputation module (paths $b-d$ and $b-d-e$).

the error corresponding to fault-free data. It is worth noting that fault-free measurements represent the normal operation of the system, these measurements constitute the majority of the collected data. Hence the fault detection module can be readily deployed as soon as enough measurements are collected. After training the model to minimize the RMSE, we choose an RMSE threshold, which will indicate the presence/absence of a system fault.

Besides its main purpose, which is fault detection, the autoencoder is also used to fine-tune the training of the model for the missing sensor measurement imputation.

B. Fault Classification

For the fault classification module, in this paper we adopt a deep neural network (DNN) that receives as input the time lags of the N sensor measurements used to monitor the industrial process. Another possible solution is to adopt a recurrent neural network. Fault classification is a multinomial (or multi-class) classification problem. While an unequal distribution of the faults might result in an imbalanced classification problem, we have addressed the most severe imbalance by training separately a model for the detection of anomalies that only requires fault-free data. In fact, fault-free sensor measurements, i.e. data collected in normal operating conditions, represent the vast majority of the data collected in IIoT. By separating the fault detection from the fault classification stage, the DNN for fault classification is only used after an unspecified fault has been detected (see Section III-A) to determine which fault has occurred. Therefore DNN for fault classification is trained only using sensor measurements collected during faulty operations to classify the different types of faults.

The DNN is not only used to determine the type of fault that has occurred, but it is also the basis to fine-tune the training of the model used to impute missing data.

C. GAN-based missing data imputation

The data imputation module relies on a Generative Adversarial Network (GAN) model that learns the correlation between the data from the input layer to replace missing sensor measurements. Considering that the main purpose of data imputation in our proposed architecture is to replace the missing values so that the fault detection and classification modules can operate correctly, the model requires both faulty and non-faulty data during training. We start from the Generative Adversarial Imputation Network (GAIN) model presented in [26]. In [26], the generator G observes the N-dimensional real data vector $\mathbf{x}$ with missing components. Let us denote by M the mask that indicates the missing values in the input dataset. The mask M, when multiplied with the complete input dataset, produces a dataset with missing values. Then G imputes the missing components conditioned on what is actually observed, and outputs an imputed vector $\hat{\mathbf{x}}$. The discriminator D receives the output of the generator as its input. It takes the complete vector generated by G and attempts to determine which components were actually observed and which were imputed. In other words the discriminator D attempts to predict the mask M. In addition to the output of the generator, D receives a hint vector, which reveals partial information about the missing values. In particular, the hint vector reveals to D all the components of the mask M except for one, which is randomly independently chosen for each sample.

The training of the GAIN model is a two step process. We first optimize the discriminator D with a fixed generator G using mini-batches of size K_D. D is trained according to equation (1), where $\mathcal{L}_D$ is defined with equation (2).

$$\min_D -\sum_{j=1}^{k_D} \mathcal{L}_D\left(\mathbf{m}(j), \hat{\mathbf{m}}(j), \mathbf{b}(j)\right) \tag{1}$$

$$\begin{aligned}\mathcal{L}_D\left(\mathbf{m}, \hat{\mathbf{m}}, \mathbf{b}\right) = \sum_{i:b_i=0} \big[&m_i \log(\hat{m}_i) \\ &+ (1-m_i)\log(1-\hat{m}_i)\big]\end{aligned} \tag{2}$$

We denote with $\mathbf{m}(j)$ the original mask associated with $j-$th sample in the mini-batch, while $\hat{\mathbf{m}}(j)$ is the corresponding predicted mask, i.e. the output of D, and $\mathbf{b}(j)$ is an

$N-$dimensional vector whose elements are all equal to 1 except for one element that is 0. The position of the 0 element in $\mathbf{b}(j)$ is the position of the only element of the mask $\mathbf{m}(j)$ that is not provided as input to D. In other words, by using (2) we train D only for the element of the mask vector that is unknown to the discriminator for each sample, which is randomly chosen for each sample.

$$\min_{G} \sum_{j=1}^{k_G} \mathcal{L}_G\left(\mathbf{m}(j), \hat{\mathbf{m}}(j), \mathbf{b}(j)\right) + \alpha \mathcal{L}_M\left(\mathbf{x}(j), \hat{\mathbf{x}}(j)\right) \tag{3}$$

After we run the training process for the discriminator D, we optimize the generator G according to equation (3) with mini-batches of size K_G. The cost function for G is the weighted sum with hyperparameter α of two components: one which applies to the missing sensors measurements - $\mathcal{L}_G$ (equation (4)); and one which applies to the observed measurements - $\mathcal{L}_M$ (equation (5)).

$$\mathcal{L}_G\left(\mathbf{m}, \hat{\mathbf{m}}, \mathbf{b}\right) = -\sum_{i:b_i=0} \left(1 - m_i\right) \log(\hat{m}_i) \tag{4}$$

$$\mathcal{L}_M\left(\mathbf{x}, \hat{\mathbf{x}}\right) = \sum_{i=1}^{d} m_i (\hat{x}_i - x_i)^2 \tag{5}$$

In [26], the authors use the RMSE as the metric to evaluate how well the model performs in terms of imputing the missing values. However, we noticed that even though the RMSE metric can be very low, in an industrial process certain measurements might have a lower tolerance to variations, while other measurements might have very limited impact on the capability to monitor the process. This is for example the case for the Tennessee Eastman (TE) process, which we used for the validation of our framework (see Section IV). Therefore, instead of relying on the RMSE, we use the feedback from the fault detection and classification modules to tune the hyperparameters of the GAIN model (see Figure 2).

The authors of [26] provide theoretical results for a dataset with values Missing Completely At Random (MCAR). Similarly, we use the MCAR approach for training of our model. However, in order to understand the impact of physical sensors failing, validation and testing are done on persistent sensor failures. That is where the feedback coming from the fault detection and classification modules plays an important role, because the imputed values have to provide enough information for these two modules to operate correctly.

IV. Evaluation

To train the models and test the performance of the framework presented in the previous section we use the TE process dataset. The TE is a chemical process that was computationally modelled in 1993 and since then has become widely adopted for benchmarking process monitoring techniques. The TE process simulation produces data in correspondence to normal operation (fault-free data) and in correspondence to 21 different simulated process faults (faulty data). Two sets of data are generated - training and testing datasets. The data consists of a multivariate time series of $N = 52$ variables

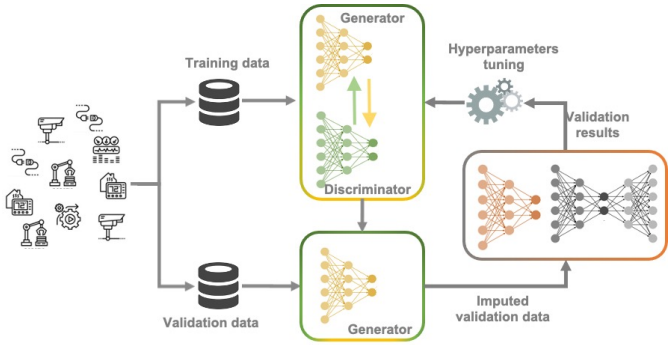


Fig. 2: Hyperparameteres tuning for the GAN.

sampled at an interval of 3 minutes. The training data and testing data span 25 hours of operation (i.e. 500 samples) and 48 hours of operation (i.e. 960 samples) respectively. While the original TE process dataset consisted of 22 runs, one normal and the remaining 21 for the faulty conditions, in this work we use the extended version provided by Reith et al. [27]. This extended dataset was generated by running 500 independent simulations for each of the runs, differing from the original ones for the random seeds used. Faults are introduced after 1 and 8 simulation hours in the training and testing files respectively. Our analysis presented in the remainder of this section does not included fault 21 since it was not part of the extended dataset. Finally, faults 3, 9, and 15 are not considered because of their unobservability from the data which results in high missed detection rates [28].

The remainder of this section presents a detailed analysis of the performance of the three components of the monitoring mechanism.

A. Fault detection

An autoencoder is a type of a neural network with a symmetric structure. The structure can be divided into two mirrored sub-structures (i.e. the encoder and the decoder). For the purpose of fault detection in this paper the autoencoder is constituted of: i) an input layer with N inputs, ii) H_A hidden layers with ReLu activation functions, and iii) an output layer with N outputs. The input layer and the first $H_A/2$ hidden layers form the encoder, and the remaining $H_A/2$ hidden layers and the output layer form the decoder. The neural network was trained with the Adam optimiser with a batch size of 10^3 samples and a constant learning rate 0.001 for 10^5 epochs in total. The training set consists of 300 of the 500 training files for the fault free scenario. We used 100 of the remaining 200 training files as validation set to optimize the number of hidden layers H_A and neurons of the autoencoder. The resulting autoencoder has $H_A = 12$ hidden layers of size $[52, 52, 48, 47, 46, 45, 45, 46, 47, 48, 52, 52]$. The resulting false alarm and missed detection rate computed on testing files are 0.12 and 0.17 respectively. In particular, we evaluated the false alarm rate using all 500 testing fault-free files and the missed detection rate on all the 500 faulty data files for each of the 17 faults under consideration.

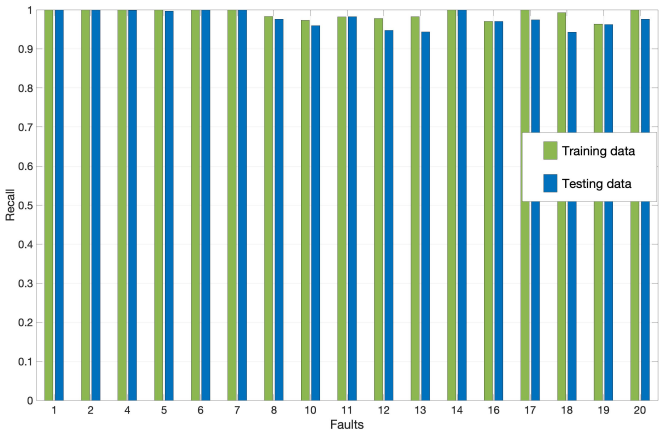


Fig. 3: Recall of the DNN for training and testing data

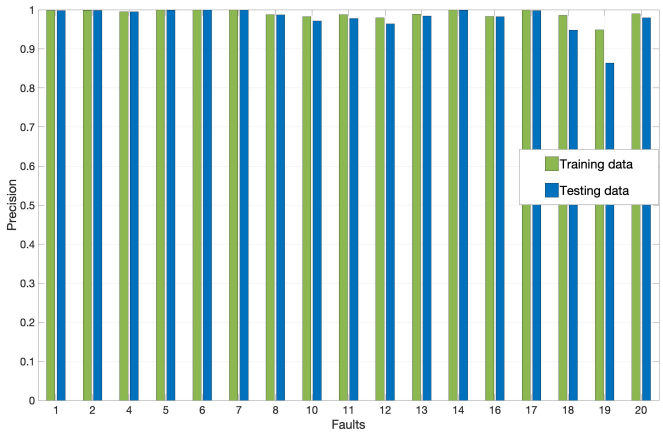


Fig. 4: Precision of the DNN for training and testing data

TABLE I: Dimensions of the discriminator and generator networks.

	Input layer size	Hidden layers size	Output layer size
Generator	624	$[1144, 1144]$	52
Discriminator	104	$[200, 200]$	52

B. Faults Classification

The DNN structure for the fault classifier is constituted by: i) an input layer with $N \times (L+1)$ inputs, ii) two hidden layers with ReLu activation functions and iii) N softmax functions as output layer. The number of lags L is 20, and the number of neurons in the two hidden layers is 250 and 60 respectively. We train the DNN model with Adam optimiser with a batch size of 512 samples and a constant learning rate 0.005 for 10^3 epochs in total. The training set consists of 300 of the 500 training files for all the considered faults. Training data corresponding to normal operating conditions are not employed to train the DNN since this classifier is only used after an anomaly has been detected. To tune the network hyperparameters (number of neurons, number of hidden layers, learning rate) we used 100 of the remaining 200 training data files not used during training. Finally, we tested the DNN using all 500 testing data files for each of the faults under consideration.

Figure 3 and Figure 4 show the recall and the precision for each fault for both training and testing data. With a minimum recall of 0.963 and 0.943 on the training set and on the test set, respectively, the DNN performs very well and correctly identifies most occurrences of each fault. The DNN also performs very well in terms of precision for each fault, i.e. what proportion of the samples identified by the model as an instance of a fault corresponds to an actual occurrence of that fault.

C. GAN-based missing data imputation

The GAN was trained using 400 of the 500 training files for all the considered faults and the normal operating conditions. These are the same 400 files that we used to train and validate the fault detection autoencoder and the fault classification DNN. We used the remaining 100 training files as validation data for the GAN. It is important to highlight that these 100 files were not used in the training or validation of the the fault detection autoencoder and the fault classification DNN, since these two models are used in the hyperparameters selection for the GAN. As mentioned earlier, the proposed architecture has to work as one interconnected system, meaning that to understand how the GAN model performs we cannot rely exclusively on validation metrics related to the GAN itself. We have to consider the system as a whole, and look at the impact on the fault detection and classification modules of the data reconstructed by the GAN. In particular, we used the precision and recall of the fault classification module and the missed detection and false alarm rate of the fault detection module to tune the GAN hyperparameters, i.e. to determine the number and size of the layers for the generator G and discriminator D, the value of α in (3) and the missing probability, i.e. how many sensors are missing for each sample during training. We considered 4 different values for α, 3 different options for the number of hidden neurons for G and D, 2 options for the number of hidden layers for G and D, and 3 values for the missing probability, resulting in 432 configurations of these hyperparameters. We randomly sampled 50 distinct configurations and trained and validated the GAN in correspondence to each of them. Table II shows the chosen α parameter and the missing probability p_m, while Table I reports the selected size for the generator and the discriminator. As shown in Table I, both the generator G and the discriminator D have one input layer, two hidden layers and one output layer. The input of the generator G consists of: the current sensor measurements (with missing values), the measurements for the last $L_G = 10$ time steps and the mask that indicates which measurements are missing. Considering that our system has $N = 52$ sensors, the size of the input layer is $N + N \times L_G + N = 624$. The input of the discriminator D consists of: the current sensor measurements (with imputed values - output of G) and the hinting vector. Hence, the size of the input layer of D is $N + N = 104$.

In the remainder of this section the performance of the

TABLE II: GAN hyperparameters

	α	p_m	Batch size	#epochs
Generator	100	0.1	1000	250,000
Discriminator	/	0.1	1000	250,000

GAN-based imputation and the performance of the moving average (MA) method that was used as benchmark are measured with respect to the impact on the fault-classification DNN and the fault detection autoencoder. We first run 10^3 simulations in which approximately 10% of the measurements are randomly missing, i.e. 5 of the $N = 52$ sensors are missing in each sample. We repeated the same experiment by considering persistent sensors' failures. The results of this preliminary analysis show that in most cases the MA mechanism achieves a very good performance. However, on closer examination of these results we observed a drop in performance of the fault-classification DNN when specific sensors were missing. In light of this, we conducted a thorough analysis by systematically removing sensors one by one and measuring the impact for each fault when imputing the missing values using the MA. This analysis showed that only 12 sensors have a significant impact on the fault classification mechanism. The set S_C of critical sensors is $S_C = \{0, 8, 12, 17, 18, 20, 21, 43, 44, 49, 50, 51\}$. Some of these sensors impact multiple faults (e.g. sensor 17), others are critical just for the detection of one of the faults (e.g. sensor 0). If any of the remaining 40 sensors is missing and its value is imputed using an MA or even using the average over fault-free data, the recall of DNN is always greater then 0.80. On the other hand, if even one of the 12 identified critical sensors is missing, the MA-based imputation does not result in an acceptable loss. In fact, in some cases the recall of the DNN can drop below 0.40. In light of these results, we focus our analysis on these 12 critical sensors to evaluate the performance of the imputation mechanism. The results of this analysis are shown in Figure 5. For each of the 500 testing files of each fault we simulated the unavailability of each of the critical sensors 10 time slots after the fault occurred and for the entire duration of the simulated process. Since the number of lags L_G used by GAN is equal to 10, for each testing file the first missing sensor measurement imputed by the GAN uses the actual sensors measurements. Henceforth, and until the end of the simulated process, the GAN relies on imputed values for the missing sensors. At each time slot, we performed the imputation using the generator of the trained GAN and the MA. We then fed the imputed data to the fault classifier DNN to determine the resulting performance of the two imputation mechanisms. Let us denote by $P(f)$ and $R(f)$ the precision and recall for fault f of the DNN in correspondence to the original testing data. We also denote by $R_{MA}(f, s)$ and $P_{MA}(f, s)$ the recall and precision for fault f of the DNN when classifying data imputed using MA for missing sensor s. In a similar way, we define $R_{GAN}(f, s)$ and $P_{GAN}(f, s)$ in case of the GAN-based imputation. By computing the average recall and precision over all the critical sensors for both imputation methods, we can compare it against $R(f)$ and $P(f)$. In particular, we define the difference in recall and precision of the two imputation methods with respect to the original recall and precision values as:

$$\Delta_R(x, f) = R(f) - \frac{\sum_{s \in S_C} R_x(f, s)}{|S_C|} \tag{6}$$

and

$$\Delta_P(x, f) = P(f) - \frac{\sum_{s \in S_C} P_x(f, s)}{|S_C|}, \tag{7}$$

where x denotes MA or GAN. Figures 5a and 5b show the resulting distribution of these differences. As we can observe, the GAN imputation results in a shift of the distribution of the difference in recall and difference in precision towards smaller values. In other words, the loss due to the missing sensor measurements is significantly mitigated by the adoption of the GAN-based imputation.

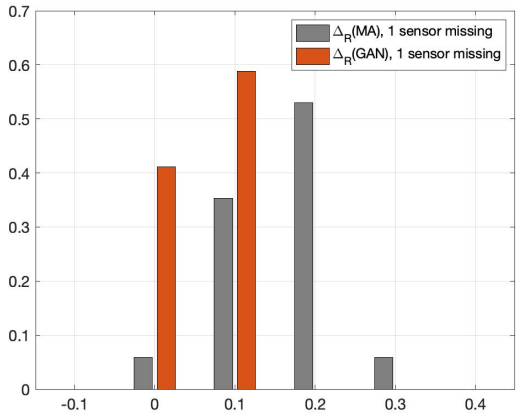
Figures 5c and 5d show the resulting distribution of the difference in recall and precision in case 2 critical sensors measurements are not available. In this case, for each of the 500 testing files of each fault we simulated the unavailability of each of the possible $2-$combinations of the critical sensors (i.e. $\binom{12}{2} = 66$). As before, we consider the worst case scenario of persistent sensors failures, i.e. the sensor measurements are unavailable for the duration of the simulated process. We then perform the imputation using the generator of the trained GAN and the MA and tested the DNN on the resulting imputed data. The results confirm that the GAN imputation significantly outperforms the MA also in this case.

As for the anomaly detection evaluation, we performed an analysis similar to that conducted for the fault-classification DNN. Table IV shows the comparison between the false alarm and the missed detection rate for the original testing files, for imputed values using MA and GAN. It is worth highlighting that to compute the missed detection rate the testing files corresponding to faulty operations have to be used, while the false alarm rate is computed on the testing files corresponding to normal operations. As before, for each of the 500 testing files of each of the 17 faults, used in this case to compute the missed detection rate, we simulated the unavailability of each of the 12 critical sensors 10 time slots after the fault occurred and for the entire duration of the simulated process[1]. In the case of false alarm rate, for each of the 500 testing files corresponding normal operations, we simulated the unavailability of each of the 12 critical sensors 10 time slots after the beginning of the simulation. Table III summarizes the number of samples used for each case. As we can see from Table IV, both the MA and GAN methods perform as well as the original data with respect to the miss detection rate. However, the false alarm rate is affected by the use of imputed values, with the GAN outperforming the MA method also in this case.

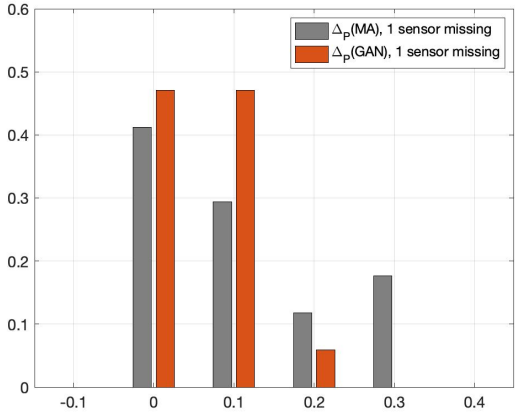
V. Conclusion

In this work we proposed a data-driven decomposition of the process to monitor the various indicators of the health

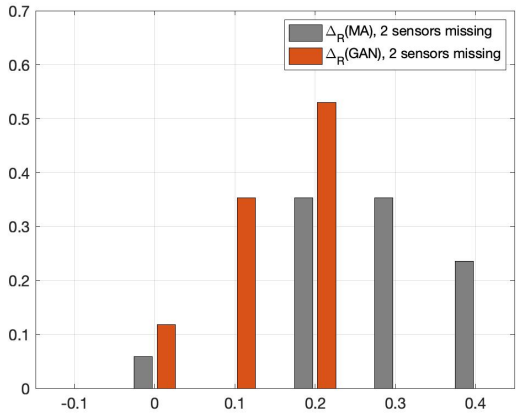
[1]Since 20 measurements per hour are collected and faults are introduced in each testing run after 8 simulation hours, the number of faulty samples in each testing file is $960 - 8 \times 20 = 800$.



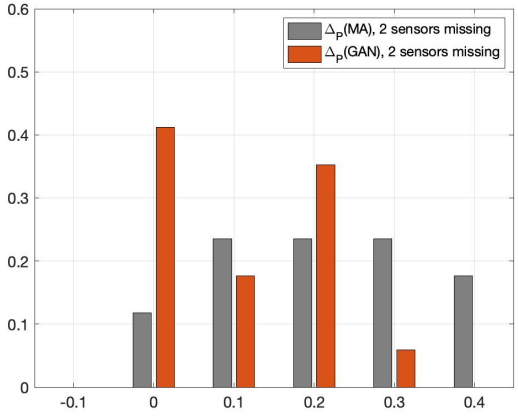
(a) Distribution of Δ_R, 1 critical sensor missing.



(b) Distribution of Δ_P, 1 critical sensor missing.



(c) Distribution of Δ_R, 2 critical sensors missing.



(d) Distribution of Δ_P, 2 critical sensors missing .

Fig. 5: Performance comparison on the test data.

TABLE III: Number of samples used for false alarm and miss detection rate computation.

	No missing data	MA/GAN
False alarm rate	500×960	$12 \times 500 \times 960$
Miss detection rate	$17 \times 500 \times 800$	$12 \times 17 \times 500 \times 800$

TABLE IV: False alarm and miss detection rate on the test set.

	No missing data	MA	GAN
False alarm rate	0.12	0.17	0.14
Miss detection rate	0.17	0.17	0.17

of a machine/component or an entire industrial process. We included in the monitoring system a module for data imputation that guarantees we have a functioning system in place even in case some of the critical sensors' measurements are missing, due for example to hardware or network issues. The data imputation module is based on GAN and was optimized by taking into account the feedback from the fault detection and classification modules, rather than using a metric, e.g. the RMSE, specific to the GAN model alone. This allowed us to fine-tune the data imputation so as to minimize the impact of missing sensor data on the capability of the system to detect and identify faults. We conducted a thorough evaluation of the proposed approach using the extended Tennessee Eastman Process dataset. Results show that the GAN-imputed data mitigate the impact on the fault detection and identification even in the case of persistently missing measurements from sensors that are critical for the correct functioning of the monitoring system.

ACKNOWLEDGMENT

This work is supported by CHIST-ERA (call 2017) via the FIREMAN consortium, which is funded by the following national foundations: Academy of Finland (n. 326270, n. 326301), Irish Research Council, Spanish and Catalan Government under grants TEC2017-87456-P and 2017-SGR-891, respectively. This material is also supported by the Air force Office of Scientific Research under award number FA9550-18-1-0214, and co-funded by Science Foundation Ireland (SFI) under the European Regional Development Fund with Grant Numbers 13/RC/2077 and 13/RC/2077_P2.

REFERENCES

[1] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," *IEEE Internet of Things Journal*.

[2] Ericsson, "Cellular networks for massive IoT," *[Online]. Available: https://www.ericsson.com/assets/local/publications/whitepapers/wp_iot.pdf*, 2020.

[3] Y. Liu, T. Dillon, W. Yu, W. Rahayu, and F. Mostafa, "Missing value imputation for industrial iot sensor data with large gaps," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6855–6867, 2020.

[4] F. Civerchia, S. Bocchino, C. Salvadori, E. Rossi, L. Maggiani, and M. Petracca, "Industrial internet of things monitoring solution for advanced predictive maintenance applications," *Journal of Industrial Information Integration*, vol. 7, pp. 4–12, 2017.

[5] J. Wan, S. Tang, D. Li, S. Wang, C. Liu, H. Abbas, and A. Vasilakos, "A manufacturing big data solution for active preventive maintenance," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2039–2047, 2017.

[6] B. Cheng, J. Zhang, G. Hancke, S. Karnouskos, and A. Colombo, "Industrial cyberphysical systems: Realizing cloud-based big data infrastructures," *IEEE Industrial Electronics Magazine*, vol. 12, no. 1, pp. 25–35, 2018.

[7] 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.

[8] P. Nardelli, C. Papadias, C. Kalalas, H. Alves, I. T. Christou, I. Macaluso, N. Marchetti, R. Palacios, and J. Alonso-Zarate, "Framework for the identification of rare events via machine learning and iot networks," in *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2019, pp. 656–660.

[9] A. González-Vidal, P. Rathore, A. Rao, J. Mendoza-Bernal, M. Palaniswami, and A. Skarmeta-Gómez, "Missing data imputation with bayesian maximum entropy for internet of things applications," *IEEE Internet of Things Journal*.

[10] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: A survey on the security of IoT frameworks," *Journal of Information Security and Applications*, vol. 38, pp. 8–27, 2018. [Online]. Available: https://doi.org/10.1016/j.jisa.2017.11.002

[11] X. Jia, Q. Feng, T. Fan, and Q. Lei, "RFID technology and its applications in Internet of Things (IoT)," *2012 2nd International Conference on Consumer Electronics, Communications and Networks, CECNet 2012 - Proceedings*, pp. 1282–1285, 2012.

[12] J. Mineraud, O. Mazhelis, X. Su, and S. Tarkoma, "A gap analysis of Internet-of-Things platforms," *Computer Communications*, vol. 89-90, pp. 5–16, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.comcom.2016.03.015

[13] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[14] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221–232, 2016.

[15] P. Leitao, S. Karnouskos, L. Ribeiro, J. Lee, T. Strasser, and A. W. Colombo, "Smart agents in industrial cyber–physical systems," *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1086–1101, 2016.

[16] S. J. Oks, A. Fritzsche, and K. M. Möslein, "An application map for industrial cyber-physical systems," in *Industrial internet of things*. Springer, 2017, pp. 21–46.

[17] S. Yin, J. J. Rodriguez-Andina, and Y. Jiang, "Real-time monitoring and control of industrial cyberphysical systems: With integrated plant-wide monitoring and control framework," *IEEE Industrial Electronics Magazine*, vol. 13, no. 4, pp. 38–47, 2019.

[18] W. Dai, H. Nishi, V. Vyatkin, V. Huang, Y. Shi, and X. Guan, "Industrial edge computing: Enabling embedded intelligence," *IEEE Industrial Electronics Magazine*, vol. 13, no. 4, pp. 48–56, 2019.

[19] H. Hellstrom, M. Luvisotto, R. Jansson, and Z. Pang, "Software-defined wireless communication for industrial control: A realistic approach," *IEEE Industrial Electronics Magazine*, vol. 13, no. 4, pp. 31–37, 2019.

[20] L. H. Chiang, R. D. Braatz, and E. L. Russel, "Fault detection and diagnosis in industrial systems," 2001.

[21] S. Yin and O. Kaynak, "Big data for modern industry: challenges and trends [point of view]," *Proceedings of the IEEE*, vol. 103, no. 2, pp. 143–146, 2015.

[22] D. Zurita, M. Delgado, J. A. Carino, and J. A. Ortega, "Multimodal forecasting methodology applied to industrial process monitoring," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 494–503, 2017.

[23] T. Chen, X. Liu, B. Xia, W. Wang, and Y. Lai, "Unsupervised anomaly detection of industrial robots using sliding-window convolutional variational autoencoder," *IEEE Access*, vol. 8, pp. 47 072–47 081, 2020.

[24] P. Hu and J. Zhang, "5g-enabled fault detection and diagnostics: How do we achieve efficiency?" *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3267–3281, 2020.

[25] C. Kalalas and J. Alonso-Zarate, "Sensor data reconstruction in industrial environments with cellular connectivity," in *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, 2020, pp. 1–6.

[26] J. Yoon, J. Jordon, and M. Schaar, "Gain: Missing data imputation using generative adversarial nets," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5689–5698.

[27] C. Rieth, B. Amsel, R. Tran, and M. Cook, "Additional tennessee eastman process simulation data for anomaly detection evaluation," *Harvard Dataverse*, vol. 1, 2017.

[28] Y. Zhang, "Enhanced statistical analysis of nonlinear processes using kpca, kica and svm," *Chemical Engineering Science*, vol. 64, no. 5, pp. 801–811, 2009.