Received May 17, 2020, accepted May 28, 2020, date of publication June 11, 2020, date of current version June 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3001275

A Predictor-Corrector Algorithm Based on Laurent Series for Biological Signals in the Internet of Medical Things

BORJA BORDEL[1], RAMÓN ALCARRIA[2], (Member, IEEE), TOMAS ROBLES[1], AND ILSUN YOU[3], (Senior Member, IEEE)

[1]Department of Information Systems, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[2]Department of Geospatial Engineering, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[3]Department of Information Security Engineering, Soonchunhyang University, Seoul 13557, South Korea

Corresponding author: Ilsun You (ilsunu@gmail.com)

This work was supported by the Spanish Ministry of Science, Innovation and Universities through the COGNOS project. It is also supported by the Soonchunhyang University Research Fund.

ABSTRACT In future engineered systems for medical applications, a tight real-time integration between physical and computational processes will be required. That integration is achieved using feedback control loops which need high quality input data streams. However, hardware platforms can barely provide such high-quality data sequences (especially if mobile nodes are considered), and mechanisms to improve and polish physical and biological signals are then necessary. This paper proposes a predictor-corrector algorithm to improve the quality and precision of data (biological) signals in Internet of Medical Things deployments, especially if composed of mobile nodes. The proposed algorithm employs an Artificial Intelligence approach and statistical learning techniques to predict future data samples and correct errors in received information. Employed mathematical models follow a prediction-correction scheme and are based on complex functions, Laurent series and the idea of complex envelope. Simulation techniques are used to evaluate the performance of the proposed solution, showing that it improves the precision of traditional linear interpolation techniques up to 85%, and cubic splines up to 20%. Processing delay during operation is, for the referred precision, around 200ms.

INDEX TERMS Data prediction, Internet-of-Medical-Things, statistical learning, artificial intelligence, Laurent series.

I. INTRODUCTION

Future engineered systems for medical applications are envisioned to be based on tight integrations between computational and physical processes [13]. This integration is supported by feedback control loops operating in real-time. In this scheme, information about physical and biological processes is usually captured by geographically sparse mobile nodes (typically low-cost physical devices), interconnected through the Internet of Medical Things (IoMT). These data and biological signals are, then, injected into computational processes and employed by control mechanisms to make decisions and actuate (once more) over the patients and the physical world [14]. In past, critical infrastructures, and other similar application scenarios (such as manufacturing systems), were totally revolutionized by this new approach [15].

However, feedback systems are greatly sensible, and high-quality data are needed to ensure the stability and adequate behavior of these new solutions. This requirement, nevertheless, is extremely hard to meet using mobile IoMT nodes because of their operation characteristics. On the one hand, as mobile elements, sometimes they might not be able to communicate (in shaded areas, for example). And, on the other hand, as low-cost devices, they operate using public frequency bands, where many interferences and noise may affect the information signals. In summary, data streams generated by mobile IoMT devices may present bursts of empty or erroneous samples. These data sequences cannot be directly employed into real-time feedback control loops, and data connectors and adapters are, then, necessary.

Although these connectors could be supported by some network technologies, such as Delay-Tolerant Networks (DTN) [16], this approach would also prevent the real-time operation of future systems (as these solutions are typically based on data storage and accumulation functions). Thus, innovative technologies to fill gaps in data sequences and

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wei.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

improve the quality of received samples are needed. Besides, all these operations must be developed in real-time. Artificial Intelligence mechanisms meet those requirements.

Therefore, in this paper, it is described a new Artificial Intelligence algorithm, focused on improving the quality of data sequences and biological signals generated by mobile IoMT nodes. The proposed algorithm includes two steps: a prediction phase and a correction procedure. The prediction phase operates in two independent domains, amplitude and frequency, thanks to the concept of complex envelope. Future samples are predicted to adapt to a complex model based on unknown holomorphic complex functions and Laurent series. On the other hand, correction procedure is supported by logical predicates, guaranteeing the global and local coherence of the information signal.

The rest of the paper is organized as follows. Section II describes the state of the art on predictive and Artificial Intelligence solutions for the Internet of Medical Things. Section III presents the proposed predictor-corrector algorithm, including the conceptual framework for the described Artificial Intelligence approach. Section IV describes an experimental validation, based on simulation techniques, carried out to evaluate the performance of the proposed solution. Section V presents the obtained experimental results and Section VI concludes the paper.

II. STATE OF THE ART

Different intelligent applications combining IoMT technologies and Artificial Intelligence (AI) have been reported during the last years. In general, all AI mechanism for IoMT applications may be classified according to three basic criteria [3]: the type of data to be processed, the processing tasks to be performed and the use case to be addressed.

Focusing on the first criterion, AI algorithms for IoMT solutions may be designed to process three different types of data [3]: stream data, massive data or historical data. Typical solutions based on historical data are heavy but highly precise mechanisms, supported by statistical estimators and specialized on global analyses and long-term behaviors [4]. Algorithms obtaining mean or most probable (common) parameters [1], dependencies among variables [12], or sets of all possible values [2] from databases with historical data are the most usual proposals. Massive data, on the other hand, are usually processed using hybrid Big Data and AI solutions [5]. Many authors consider this, a particular case of historical data processing [6], although it presents some specific characteristics. In general, massive data increase the understanding of AI solutions about application scenarios, and Big Data techniques allow an efficient use of these data [7] (although real-time operation is still a pending challenge [8]). Thus, more advanced mechanisms have been reported in this area, such as cognitive decision systems to manage networks [10] and other critical infrastructures [9], or eHealth platforms to manage all users' data [11]. Finally, stream data enable real-time AI applications, but with much lower reasoning capacity and shorter time validity than previous solutions. In that way, these techniques are typically supported by elemental mathematical algorithms such as the Fourier decomposition [17] or numerical models [18].

The proposed predictor-corrector algorithm in this paper employs two different types of data: stream data (generated in real-time by IoT devices), and some historical data, employed to enrich our models with a greater understanding about the data sequences. Both sources are balanced to allow the algorithm both, to operate in real-time and improve its capabilities. To do that, a new mathematical model based on Laurent series is proposed.

In respect to the second criterion, AI algorithms for Internet-of-Medical-Things systems perform five elemental processing tasks: classification, regression, clustering, feature extraction, anomaly detection [19]. Classification algorithms were the first AI mechanism, and typically are based on decision trees [20], the K-Nearest Neighbor algorithm [21], Bayesian networks [22] or Support Vector Machines [23]. All these technologies present a good behavior but require an initial training process (an approach known as supervised learning). Regression mechanisms try to find dependencies among observed objects. Depending on the mathematical dependency being searched, different unknown parameters must be discovered. Linear regression [24], Support Vector Regression [25], Neural Networks (which may be also employed in classification and clustering operations) [26], and Bagging techniques [27] are some of the most employed AI method for regression discovery. Clustering solutions, contrary to classification technologies, cannot label observed objects but only group them into different sets sharing relevant characteristics or patterns. Clustering for IoMT deployments may be based on hierarchical [28] or scalable [29] techniques, or high dimensionality data [30] or partitioning algorithms [31]. Feature extraction solutions analyze data to obtain the relevant (or interesting) characteristic of the observed data. In the context of IoMT devices, principal component [32] and canonical correlation [33] analyses are the most employed approaches, although deep learning networks [34] are being reported recently. The last operation, anomaly detection, is the least common, and usually one-class support vector machines [35] are used when employed this AI mechanism together with IoMT nodes.

The proposed predictor-corrector algorithm defines a new approach for AI regression solutions, where dependencies are represented by Laurent series, describing unknown functions whose characteristic parameters are obtained through an initial training phase. Besides, during the correction phase, feature extracted from IoMT nodes and signals will allow us to remove errors from data samples, so receive information will maintain the same characteristics along time.

Finally, different AI mechanisms have been reported according to the application scenario. Smart traffic scenarios are probably the most studied application [36], [37]. Smart health [38] and Smart cities [39] are also common, together with prediction solutions [17] (typically about weather).

Sparse works on AI and IoT for smart agriculture [40], [43] or traffic air control [5] have been also reported.

In this paper, the proposed solution is applicable to every scenario where a deployment of mobile IoT nodes are generating stream data sequences, although it is especially designed for feedback control systems.

III. DATA PREDICTOR-CORRECTOR ALGORITHM

A data predictor-corrector algorithm enables the improvement of the data sequences and biological signals quality by completing gaps and empty samples with coherent and realistic values. Besides, received samples may be corrected to meet the expected global and local features from information signals. This section describes the proposed algorithm and AI framework.

A. OVERVIEW

The proposed algorithm (see Figure 1) includes three basic steps: signal processing, prediction step and correction step.

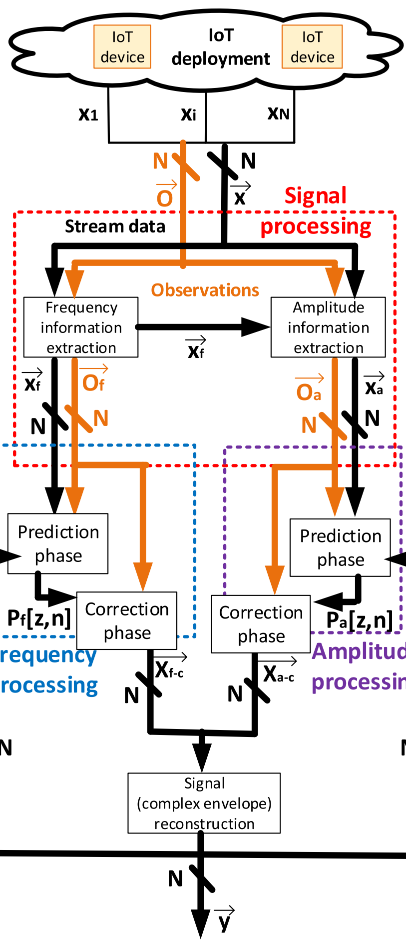

FIGURE 1. Global scheme of the proposed predictor-corrector AI algorithm.

In general, IoMT devices generate pass-band signals which contain two different and independent information: frequency information and amplitude information. In order to maximize the understanding of AI algorithms about signals, a signal processing step separating both information is considered. On the other hand, in general, it is considered IoMT deployments generate a set of N stream data sequences $\vec{x}$, which in the most generic case, may present interdependencies among them (1).

$$\vec{x} = \{x_i[n]\, i = 1, \ldots, N\} \tag{1}$$

These signals are discrete, typical in digital solutions, and should report a new sample each T_i seconds. Besides, the IoMT deployment, apart from stream sequences, also generates a set of observations $\vec{\mathcal{O}}$, employed to train AI algorithms. Described AI algorithm in this work is a Machine Learning (ML) solution, based on supervised learning. The proposed ML solution is not a standard technique, such as classifiers, although (as we are seeing) it integrates some existing technologies, such as decision-making solutions during the reconstruction phase. Supervised learning is, actually, the appropriate solution for this work; as we are solving regression problems where we have an important a priori knowledge [48].

Then, after signal processing, two N-dimensional data sequences are produced. The first one $\vec{x_a}$ contains the amplitude information (low-pass signals); and the second one $\vec{x_f}$ the frequency information (numerical series). In parallel, both sequences feed two identical prediction algorithms supported by AI learning and Laurent series. In this step, an AI regression algorithm is running. However, instead of linear or logistic functions (as usual), this regression mechanism considers unknown functions developed as Laurent series to obtain the a priori probability distribution. After obtaining density functions $P[z, n]$ describing the probability of each possible value to be the next sample, a correction phase is performed. Global and local restrictions are applied to decide if received samples must be corrected or directly replaced by the predicted ones. Restrictions are logical predicates about signals that were previously extracted. Corrected samples, $\overrightarrow{x_{a-c}}$ and $\overrightarrow{x_{f-c}}$, are finally combined to reconstruct the complete sample $\vec{y}$, using the concept of complex envelope.

B. SIGNAL PROCESSING: FREQUENCY AND AMPLITUDE INFORMATION EXTRACTION

The first step in the proposed solution is a signal processing phase. Figure 2 shows the described algorithm.

Each physical or biological signal $x_i[n]$ is, in general, a pass-band signal. Thus, it may be rewritten as a combination of two new signals (2) using the complex envelope $x_a[n]$ (3) and the instantaneous carrier frequency $x_f[n]$ (4).

$$\begin{aligned} x_i[n] &= \mathcal{R}e\left\{x_a^i[n] \cdot e^{j \cdot x_f[n] \cdot n \cdot T_i}\right\} = \\ &= x_i^{PH}[n] \cdot cos[\omega_c \cdot n \cdot T_i] - x_i^{Q}[n] \cdot sin[\omega_c \cdot n \cdot T_i] \end{aligned} \tag{2}$$

$$x_a^i[n] = x_i^{PH}[n] + j \cdot x_i^{Q}[n] \tag{3}$$

$$x_f^i[n] = \omega_c^i[n] \tag{4}$$

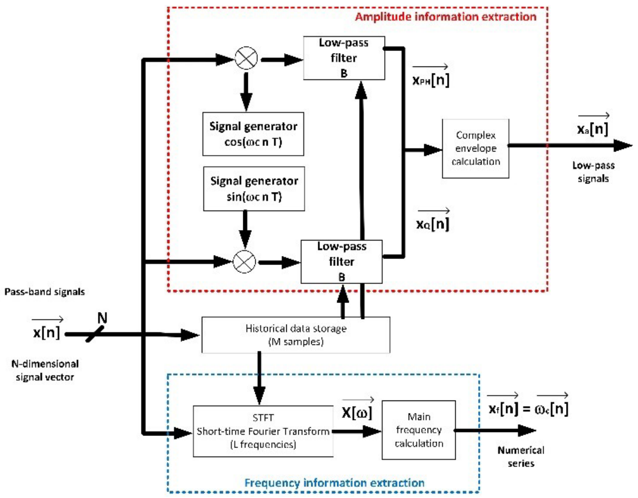

FIGURE 2. Signal processing phase. Block diagram.

The in-phase signal $x_i^{PH}[n]$ and quadrature signal $x_i^{Q}[n]$ are baseband signals and may be easily obtained through a sinusoidal signal generator, a multiplier and a low-pass filter. In fact, and considering previous definitions (2), the multiplication of any pass-band signal and a sinusoidal signal generates both, a baseband signal and a high frequency signal (5,6). Frequency information to configure the sinusoidal signal generators will be obtained from frequency information extraction module.

$$
\begin{aligned}
x_i^{cos}[n] &= x_i[n] \cdot cos[\omega_c \cdot n \cdot T_i] = \\
&= \frac{1}{2} x_i^{PH}[n] \cdot (1 + cos[2\omega_c \cdot n \cdot T_i]) \\
&\quad - \frac{1}{2} x_i^{Q}[n] \cdot sin[2\omega_c \cdot n \cdot T_i] \qquad (5) \\
x_i^{sin}[n] &= x_i[n] \cdot sin[\omega_c \cdot n \cdot T_i] = \\
&= \frac{1}{2} x_i^{Q}[n] \cdot (-1 + cos[2\omega_c \cdot n \cdot T_i]) \\
&\quad - \frac{1}{2} x_i^{PH}[n] \cdot sin[2\omega_c \cdot n \cdot T_i] \qquad (6)
\end{aligned}
$$

Then, to extract the amplitude information from the original signal (removing all information about carrier frequency), it is enough to employ a low-pass filter. As frequency components to be removed are far away from the baseband signals, and amplitude should be extracted without any distortion, in our proposal we are using a Butterworth filter with a bandwidth of B Hertz. This filter presents a wide transition band, but it is totally flat in the pass band. The digital cut frequency f_{cut}^i may be obtained considering the sampling theory (7), and the transference function (in the Z domain) $H_K(z)$ for every filter of order K (8) may be deducting using the Butterworth polynomials $B_K(z)$, and calculating finite differences through the backward procedure (9). In order to compensate attenuation coefficients in previous steps, Butterworth filter is designed to amplify the received signals, so $G = 2$.

$$
\begin{aligned}
f_{cut}^i &= B \cdot T_i \\
H_K(z) &= \frac{G}{B_K(z)} \qquad (7) \\
z &= \sum_{j=0}^{K} c_j \cdot \left(\frac{\frac{z-1}{z} \cdot \frac{1}{T_i}}{2\pi f_{cut}^i} \right)^j \\
c_j &= \sum_{r=0}^{j} \frac{cos\,(r-1)\,\frac{\pi}{2K}}{sin\left(r \cdot \frac{\pi}{2K}\right)} \qquad (8) \\
\dot{x}[n] &= x[n] - x[n-1] \qquad (9)
\end{aligned}
$$

Finally, the in-phase and quadrature signal can be computed using the convolution operator and the inverse Z transform (10).

$$
\begin{aligned}
x_i^{PH}[n] &= x_i^{cos}[n] * h_K[n] = \\
&= \sum_s x_i^{cos}[s] \cdot h_K[\text{n-s}] \\
x_i^{Q}[n] &= x_i^{sin}[n] * h_K[n] = \\
&= \sum_s x_i^{sin}[s] \cdot h_K[\text{n-s}] \\
h_K[n] &= TZ^{-1}\{H_K(z)\} \qquad (10)
\end{aligned}
$$

As convolution sum requires an infinite number of samples; last M_1 received samples must be stored in a historical register. In order to guarantee a good behavior, at least as many samples as indicated by the filter's order must be stored (11).

$$
M_1 \geq K \qquad (11)
$$

Previously described algorithm only extracts amplitude information from received signals. In order to capture also the frequency information (main or central frequency), it is employed the Short-Time Fourier Transform (STFT), $X_i^{STFT}[f_k, r]$. The STFT employs the Fast Fourier Transform (FFT) algorithm to compute the Discrete Fourier Transform in L different discrete frequencies, $f_k = \frac{k}{L}$, each r time instants (12).

$$
X_i^{STFT}[f_k, r] = \sum_{j=0}^{M_2-1} x_i[r+j] \cdot w[j] \cdot e^{-j \cdot 2\pi \cdot f_k \cdot j}
$$
$$
k = 0, \ldots, L-1 \qquad (12)
$$

All frequencies belong to the range [0, 1], containing all possible frequencies for discrete signals. The window function $w[n]$ is considered to have a length of M_2 samples. Thus, the number of frequencies must fulfill the condition $L \geq M_2$, so the obtained STFT represents correctly the data sequence. Any existing window function could be employed, although in this case (as we desire the best spectra resolution) we are employing the rectangular window.

Then, the storage for past signal samples in the signal processing module can be obtained as $M = max\{M_1, M_2\}$. Finally, the STFT, in order to preserve the global features and continuity between different windows, must be calculated each r time instants, being $r = \dot{R}.R$, besides, must be selected so that windows overlap (13).

$$L \geq M_2 \geq R \tag{13}$$

Once obtained the STFT, the frequency information may be easily obtained detecting the discrete frequency so the STFT is maximum (14).

$$x_f^i[n] = \omega_c^i[n] = 2\pi f_k^i[n] = 2\pi \frac{k_i}{L} : \quad X_i^{STFT}[f_k, r] = max\left\{X_i^{STFT}\right\} \tag{14}$$

C. ARTIFICIAL INTELLIGENCE PREDICTIVE ALGORITHM

Information signals $\vec{x_f}$ and $\vec{x_a}$, obtained from the signal processing module, are totally independent sequences (with different characteristics and properties). However, the proposed predictive algorithm in this section is totally agnostic regarding all these details. Therefore, we are describing the proposed AI algorithm using a generic signal $\vec{x_s}$. In the most general case, $\vec{x_s}$ is a N-dimensional vector composed of complex data sequences (15). These sequences take values from a universe U_i.

$$\vec{x_s} = \overrightarrow{x_s[n]} = \left(x_s^1[n], \ldots, x_s^i[n], \ldots, x_s^N[n]\right) : \quad x_s^i[n] \in U_i \subseteq \mathbb{C} \;\; \forall i, n \tag{15}$$

These complex sequences contain frequency information and amplitude information (managed as complex envelopes) and force us to employ complex functions to operate with them. Other techniques based on real functions could be developed, but (then) two main disadvantages should be considered. First, for each complex envelope two real functions are defined (3), so the number of functions to be considered duplicates and the processing delay, algorithm's spatial and temporal complexity as well. And, second, sequences $x_i^{PH}[n]$ and $x_i^{Q}[n]$ in the complex envelope are not independent (as they refer the same amplitude information), so they cannot be manipulated separated from each other in a precise manner. Consequently, we are employing complex functions and Laurent series, instead of real variables and Taylor series.

This vector signal $\overrightarrow{x_s[n]}$ represents the superposition of N stochastic processes $p_i[z, n]$, which (in general) will not be independent. z is a variable varying within the corresponding universe U_i; and n is the temporal variable. About these stochastic processes, a set $\vec{\mathcal{O}}$ of N_o observations (known as training set) is collected (16).

$$\vec{\mathcal{O}} \in \left(\mathbb{C}^N\right)^{N_o} \tag{16}$$

In this context, we look for a density function $\overrightarrow{\rho[z, n]}$ estimating the weight (probability) of each point $\vec{z} = \vec{u_i} \in U_1 \times \ldots \times U_N$ at each time instant n; i.e. we are trying to obtain a vector of estimations $\widehat{p_i}[z, n]$, for the collection of stochastic processes associated to the signal $\vec{x_s}$ (17).

$$\overrightarrow{\rho[z, n]} = \left(\widehat{p_1}[z_1, n], \ldots, \widehat{p_i}[z_i, n], \ldots, \widehat{p_N}[z_N, n]\right) \tag{17}$$

In practice (as infinite measurements are not feasible), this density function must be computed regarding only the collected observations (not the entire universe), so obtained function $\overrightarrow{\rho_{\mathcal{O}}[z, n]}$ is essentially a restriction of $\overrightarrow{\rho[z, n]}$ to set $\vec{\mathcal{O}}$ (18).

$$\overrightarrow{\rho_{\mathcal{O}}[z, n]} = \overrightarrow{\rho[z, n]}\Big|_{\vec{\mathcal{O}}} \tag{18}$$

Thus, we must obtain a density function such that its restriction to set $\vec{\mathcal{O}}$ can be easily generalized to the entire universe $U_1 \times \ldots \times U_N$.

In this paper we are considering, at this point, two assumptions:

- The global density function $\overrightarrow{\rho[z, n]}$ is identical to the restricted function to set $\vec{\mathcal{O}}$ (19). Actually, to be an AI algorithm, the proposed solution must, from observing set $\vec{\mathcal{O}}$, understand the entire phenomenon and be applicable to the global universe. To meet this requirement, both density functions must be equal.

$$\overrightarrow{\rho[z, n]} = \overrightarrow{\rho_{\mathcal{O}}[z, n]} = \overrightarrow{\rho[z, n]}\Big|_{\vec{\mathcal{O}}} \tag{19}$$

- The density function $\overrightarrow{\rho_{\mathcal{O}}[z, n]}$ (for each time instant) may be analytically expressed as a function of the n_o past observed events and their d_0 first backward finite differences (20). In total, $T = N \cdot (n_0 + d_0)$ variables are considered.

$$\begin{aligned} \overrightarrow{\rho_{\mathcal{O}}[z, n]} = F\Bigg[\vec{z}, &\left\{\overrightarrow{x_s[n-i]}, i = 1, \ldots, n_o\right\}, \\ &\left\{\dot{\overrightarrow{x_s[n-i]}}, i = 1, \ldots, n_o - 1\right\}, \ldots, \\ &\left\{\overbrace{\overrightarrow{x_s[n-i]}}^{\cdots\cdot}, i = d_0, \ldots, n_o - d_0\right\}\Bigg] \end{aligned} \tag{20}$$

Vector function $F[\cdot]$ is usually supposed to have a certain form (linear, logistic, etc.), but in our proposal we are assuming this function is totally unknown; we only require from it to be holomorphic. As a N-dimensional vector function, $F[\cdot]$ is made of N components $f_i[\cdot]$ which must be holomorphic as well (21).

$$F[\cdot] = \left(f_1[\cdot], \ldots, f_i[\cdot], \ldots, f_N[\cdot]\right) \tag{21}$$

In these conditions, it is possible to develop this unknown function as Laurent series [47] in the interior of a poly-annulus A_z (22), as shown at the bottom of the next page. In fact, each component $f_i[\cdot]$ is developed as a different Laurent series (23), as shown at the bottom of the next page. Only for clarity, hereinafter all variables $\overrightarrow{x_s[n-i]}$, $\dot{\overrightarrow{x_s[n-i]}}$, etc., are named as ξ_i being $i = 1, \ldots, T$.

In these expressions, ∂^* indicates the frontier of a domain in a complex space. The center of the annulus A_z, as seen, affects the Laurent series. However, simple mathematical manipulations may remove this effect, provided the function $f_i[\cdot]$ is holomorphic in all points within the U_i (24).

$$f_i\left[z_i, \vec{\xi}\right] = = \sum\nolimits_{\lambda_0,\ldots,\lambda_T \in \mathbb{Z}} b_{\lambda_0,\ldots,\lambda_T} \cdot (z_i)^{\lambda_0} \cdot (\xi_1)^{\lambda_1} \cdot \ldots \cdot (\xi_T)^{\lambda_T} \tag{24}$$

With this expression, Laurent series behave exactly the same the unknown function $f_i[\cdot]$. Nevertheless, infinite terms must be considered in the resulting polynomial. Accepting a certain error ε (25), the Laurent series may be truncated to the Σ terms with lower order (26), both negative and positive powers (27). This truncated series is employed hereinafter.

$$\varepsilon = \sum\nolimits_{\lambda_0,\ldots,\lambda_T \notin \mathbb{Z}_\Sigma} b_{\lambda_0,\ldots,\lambda_T} \cdot (z_i)^{\lambda_0} \cdot (\xi_1)^{\lambda_1} \cdot \ldots \cdot (\xi_T)^{\lambda_T} \tag{25}$$

$$\Sigma = (2C_{max} + 1)^{T+1} \tag{26}$$

$$\begin{aligned} f_i\left[z_i, \vec{\xi}\right] &\simeq \tilde{f}_i\left[z_i, \vec{\xi}\right] \\ &\simeq \sum\nolimits_{\lambda_0,\ldots,\lambda_T \in \mathbb{Z}_\Sigma} b_{\lambda_0,\ldots,\lambda_T} \cdot (z_i)^{\lambda_0} \cdot (\xi_1)^{\lambda_1} \cdot \ldots \cdot (\xi_T)^{\lambda_T} \\ \mathbb{Z}_\Sigma &= \{-C_{max}, \ldots, 0, \ldots, C_{max}\} \end{aligned} \tag{27}$$

Thus, as function $f_i[\cdot]$ is unknown and, therefore, coefficients $b_{\lambda_1,\ldots,\lambda_T}$ cannot be analytically calculated, the described Laurent series presents Σ unknown coefficients. Considering a training set $\vec{\mathcal{O}}$ big enough, these coefficients may be obtained using algebraic equation resolution techniques. This procedure is known as the training phase of the AI mechanism.

As density functions depend on past events $\{\xi_1, .., \xi_{n_0}\}$ (and their finite differences), obtained estimations for stochastic processes are conditioned to the past observed events (28). These events, at the same time, are observations whose probability must be obtained through the proposed density functions. Final predictive algorithm must address this recursive problem.

$$\widehat{p_i}[z_i, n] = g\left(z_i; n \left| \xi_1 = e_1, .., \xi_{n_0} = e_{n_0}\right.\right) = = \tilde{f}_i\left[z_i, \vec{\xi}\right] \tag{28}$$

First, we are considering all points in the universe have the same probability in the first time instant, $n = 0$ (29).

$$P_{\widehat{p_i}}[z_i, 0] = \frac{1}{card\{U_i\}} \forall z_i \in U_i \tag{29}$$

Then, using the Bayes theory and the total probability theorem, it is possible to obtain the probability of each point in the universe U_i at each time instant (30). This expression could be expressed also in a non-recursive form, but it is not necessary as this formula may be easily implemented in the proposed predictive algorithm, which is finally shown in Algorithm 1.

Algorithm 1 Predictive Algorithm

Output: Set of reticula $\{P_{\widehat{p_i}}[z, n], i = 1, .., N\}$
Initiate the set of reticula $\left\{P_{\widehat{p_i}}[z, 0] = \frac{1}{card\{U_i\}}, i = 1, .., N\right\}$
for each time instant n **do**
  Initiate $\{P_{\widehat{p_i}}[z, n] = 0, i = 1, .., N\}$
  for each reticula $P_{\widehat{p_i}}$ **do**
    for each point $z \in U_i$ **do**
      for each possible variation (with repetition)
        $\vec{v} = (e_1, \ldots, e_{n_0})\ e_j \in U_i$ **do**
        Obtain $\tilde{f}_i[z, \vec{v}]$
        Calculate $B = P_{\widehat{p_i}}[e_1, n-1] \cdot \ldots \cdot P_{\widehat{p_i}}[e_{n_0}, n - n_0]$
        $P_{\widehat{p_i}}[z, n] = P_{\widehat{p_i}}[z, n] + \tilde{f}_i[z, \vec{v}] \cdot B$
      end for
    end for
  end for
end for

$$P_{\widehat{p_i}}[z_i, n] = \sum_{\forall e_1 \in U_i} \ldots \sum_{\forall e_{n_0} \in U_i} g\left(z_i; n \left| \xi_1 = e_1, .., \xi_{n_0} = e_{n_0}\right.\right) \cdot \prod_{j=1}^{n_0} P_{\widehat{p_i}}[e_j, \text{n-j}] \tag{30}$$

The proposed algorithm, basically, employs the density functions based on Laurent series, and the total probability theorem (30), to update (at each time instant) a reticulum for

$$\begin{aligned} A_z &= A\left(\overrightarrow{z_0}; l_0, \mathcal{L}_0, l_1, \mathcal{L}_1, \ldots, l_T, \mathcal{L}_T\right) = \\ &= A\left(z_0^0; l_0, \mathcal{L}_0\right) \times \ldots \times A\left(z_0^T; l_T, \mathcal{L}_T\right) \\ A\left(z_0^i; l_i, \mathcal{L}_i\right) &= \left\{z \vdots l_i < \left|z - z_0^i\right| < \mathcal{L}_i\right\} \\ f_i\left[z_i, \vec{\xi}\right] &= \\ &= \sum\nolimits_{\lambda_0,\ldots,\lambda_T \in \mathbb{Z}} b_{\lambda_0,\ldots,\lambda_T} \cdot \left(z_i - z_0^0\right)^{\lambda_0} \cdot \left(\xi_1 - z_0^1\right)^{\lambda_1} \cdot \ldots \cdot \left(\xi_T - z_0^T\right)^{\lambda_T} \end{aligned} \tag{22}$$

$$\begin{aligned} b_{\lambda_0,\ldots,\lambda_T} &= \\ &= \frac{1}{(2\pi j)^{T+1}} \int_{\mathcal{C}} \frac{f_i[z_i, \xi_1, \ldots, \xi_i, \ldots, \xi_T]}{\left(z_i - z_0^0\right)^{\lambda_0+1} \cdot \left(\xi_1 - z_0^1\right)^{\lambda_1+1} \cdot \ldots \cdot \left(\xi_T - z_0^T\right)^{\lambda_T+1}} dz_i \xi_1, \ldots, \xi_T \\ \mathcal{C} &= \partial^* A\left(\vec{z_0}; \Lambda_1, \ldots, \Lambda_T\right) \quad \forall l_i < \Lambda_i < \mathcal{L}_i \end{aligned} \tag{23}$$

each i-th component in $\vec{x_s}$ describing the probability of each point in the universe U_i to be the following observed event. Figure 3 represents the described reticulum.

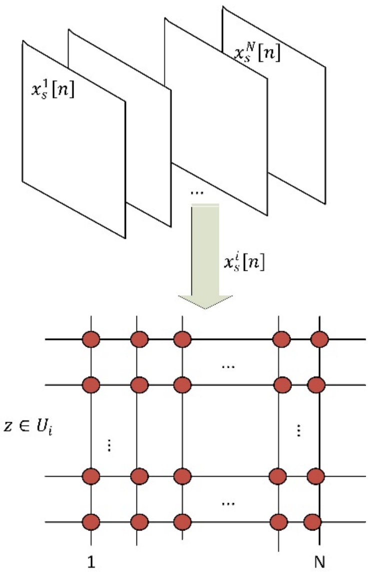

FIGURE 3. Reticulum for calculations in the prediction phase.

D. CORRECTION ALGORITHM

After the prediction step, a probability reticulum $P_{\widehat{p_i}}[z, n]$ for each i-th component of both $\vec{x_f}$ and $\vec{x_a}$ signals is obtained. Now, before obtaining the final (corrected) N-dimensional sample $\overrightarrow{x_{s-c}[n]}$, a correction phase is performed. This phase considers a set of logical predicates about points in the universe, the n_0 past generated samples and their d_0 first backward finite differences (31).

$$\begin{aligned} \vec{\mu} &= \{\mu_1, \ldots, \mu_w\} \\ \mu_i &= \mu_i\left(z_i, \vec{\psi}\right) = \beta \quad \beta \in \{true, false\} \\ \vec{\psi} &= \left\{\overrightarrow{y[n-k_1]}, k_1 = 1, \ldots, n_o\right\}, \\ &\quad \left\{\overrightarrow{\dot{y}[n-k_2]}, k_2 = 1, \ldots, n_o - 1\right\}, \ldots, \\ &\quad \left\{\overbrace{\overrightarrow{y[n-k_i]}}^{\cdots}, k_i = d_0, \ldots, n_o - d_0\right\} \end{aligned} \tag{31}$$

These predicates refer some signals' and IoT devices characteristics and describe the global and local consistence and coherence of data sequences. These predicates describe features such as the maximum value for signals (according to IoT physical devices), or the signal maximum possible variation per time unit.

Mathematically, the correction process is the calculation of a multidimensional stochastic process (32), considering the points in the universe U_i and the logical predicates. As both components are independent, this stochastic process may be calculated as the multiplication of two different probabilities (33). Moreover, logical predicates are independent among them, so the same reasoning may be applied to them (34).

$$P^c_{\widehat{p_i}}[z_i, n] = \Pi\left[z_i, \vec{\mu}\left(z_i, \vec{\psi}\right); n\right] \tag{32}$$

$$\Pi\left[z_i, \vec{\mu}\left(z_i, \vec{\psi}\right); n\right] = P_{\widehat{p_i}}[z_i, n] \cdot P\left[\vec{\mu}\left(z_i, \vec{\psi}\right), n\right] \tag{33}$$

$$P\left[\vec{\mu}\left(z_i, \vec{\psi}\right), n\right] = \prod_{j=1}^{w} P\left[\mu_i\left(z_i, \vec{\psi}\right), n\right] \tag{34}$$

In this step, logical predicates are considered to follow a Bernoulli distribution (35). This assumption is time independent.

$$P\left[\mu_i\left(z_i, \vec{\psi}\right), n\right] = \begin{cases} p_b & if\ \mu_i\left(z_i, \vec{\psi}\right) = true \\ q_b & if\ \mu_i\left(z_i, \vec{\psi}\right) = false \end{cases} \tag{35}$$

Thus, the final corrected reticulum $P^c_{\widehat{p_i}}[z_i, n]$ (32) may be easily obtained evaluating logical predicates for all points in the universe. In that way, certain points z_i may be less probable because of physical restrictions. The final corrected sample $\overrightarrow{x_{s-c}[n]}$ is obtained, for each i-th component, as the barycenter of the corrected reticulum $P^c_{\widehat{p_i}}[z_i, n]$ (36).

$$x^i_{s-c}[n] = \frac{\sum_{\forall z_i \in U_i} z_i \cdot P^c_{\widehat{p_i}}[z_i, n]}{\sum_{\forall z_i \in U_i} P^c_{\widehat{p_i}}[z_i, n]} \tag{36}$$

E. RECONSTRUCTION

At this point, two predicted and corrected samples $\overrightarrow{x_{f-c}[n]}$ and $\overrightarrow{x_{a-c}[n]}$ are obtained. In order to reconstruct the final and global samples it is only necessary to apply the definition of complex envelope (37).

$$y^i_{p-c}[n] = \mathcal{R}e\left\{x^i_{a-c}[n] \cdot e^{j \cdot x_{f-c}[n] \cdot n \cdot T_i}\right\} \tag{37}$$

At the same time this predicted sample is generated, a physical sample $\vec{x}$ was received from IoT devices. Then, a decision must be taken about both samples. The following decision scheme was implemented (38):

- If $x_i[n]$ is an empty sample, the final corrected sample to be generated and transmitted (to applications, back to signal processing module, etc.) is $y^i_{p-c}[n]$.
- If $x_i[n]$ belongs to the open ball $B(y^i_{p-c}[n], \delta_e)$, i.e. absolute error in $x_i[n]$ compared to expected sample $y^i_{p-c}[n]$ is lower than δ_e units, the final accepted sample is $x_i[n]$.
- If $x_i[n]$ belongs to the closed annulus $C(y^i_{p-c}[n], \delta_e, \delta_E)$, the final sample is the arithmetic average between $x_i[n]$ and $y^i_{p-c}[n]$.
- Otherwise, the proposed final sample is $y^i_{p-c}[n]$.

$$y_i[n] = \begin{cases} y^i_{p-c}[n] & if\ x_i[n]\ is\ empty \\ x_i[n] & if\ x_i[n] \in B(y^i_{p-c}[n], \delta_e) \\ \dfrac{x_i[n] + y^i_{p-c}[n]}{2} & if\ x_i[n] \in C(y^i_{p-c}[n], \delta_e, \delta_E) \\ y^i_{p-c}[n] & otherwise \end{cases} \tag{38}$$

IV. EXPERIMENTAL VALIDATION: A FIRST CASE STUDY USING SIMULATION TOOLS

In order to validate the proposed AI solution, a simulation scenario was built. Proposed experiments evaluate the performance of the described predictor-corrector algorithm in terms of accuracy and computational efficiency.

To demonstrate the algorithm effectiveness, we investigate the precision in predicted samples, comparing the generated predicted and corrected data with real data obtained from IoT devices (before occurring any error caused by mobility, noise, etc.). As data streams are treated as real number, the traditional Euclidean distance enable us to estimate that precision measure as a standard relative error. This first experiment was repeated for different communication signals (with different entropy values). To show the computational efficiency of the proposed solution, the converge time during the training phase, and the calculation delay during operation, are analyzed. Finally, to evaluate the performance of the proposed artificial intelligence solution, the confusion matrix is calculated and discussed.

To perform these studies, a simulation scenario, described and executed using MATLAB 2017a software, is employed. All simulations were performed using a Linux architecture (Linux 16.04 LTS) with the following 604 hardware characteristics: Dell R540 Rack 2U, 96 GB RAM, two processors Intel Xeon Silver 4114 2.2G, HD 2TB SATA 7,2K rpm. Using MATLAB, three different agents were implemented. Each one describes the behavior and interfaces of a different mobile IoMT device. The first agent model represents a camera for medical pictures. Generated data (three data streams) represent the three components (red, blue and green) of pixels. The second agent represents an ultrasound sensor to detect masses and create an image of the organs. Generated data (one stream) represent distances. Finally, the third agent is a standard fixed body temperature sensor. Only one stream is represented with digital temperature information. Every agent may generate signals with different entropy values, in the range $\left[0, \frac{1}{2}\right]$.

MATLAB precision was established in a fixed manner to one hundred significative digits. In that way, the numerical error caused by MATLAB data processing methods was around 10^{-100}. This value is much lower than any other variable or expected result, so we can consider negligible its effect on obtained results. Besides, to avoid MATLAB algorithms to increase the calculation speed by reducing the desired precision this functionality was disabled.

The simulation scenario contained the same number of agents of each type. The number of devices in the simulation scenario is selected to match realistic applications reported in the scientific literature [41], [42].

All these agents, once deployed, transmitted their data to a new agent acting as central server where the proposed predictor-corrector algorithm is running. This communication process occurs through a wireless channel where noise, electromagnetic interferences and temporary displacement caused by mobility are applied. Both, real data generated by IoMT nodes, the corrupted data received in the central server and the final samples predicted and corrected by the proposed AI solution are collected.

The study was repeated for different numbers of terms in the Laurent series Σ. For each case, simulations where repeated twelve times, and mean values where extracted as final result. Table 1 shows the values for the resting parameters in the proposed algorithm.

TABLE 1. Values for parameters in predictor-corrector algorithm during the validation experiments.

Parameter	Value	Comments
δ_e	0,075	Max. error 7,5%
δ_E	0,2	Max. error 20%
μ_i	---	Max. value for signal and their finite differences
T_i	100μs	For all signals
n_0	100	Previous 10ms effect the prediction
N_0	12000	1.2 seconds for training
N	120	30 IoT devices, 10 of each type
d_0	4	Differences up to fourth order

Results are also compared to measures obtained from traditional prediction solution based on splines and linear regression.

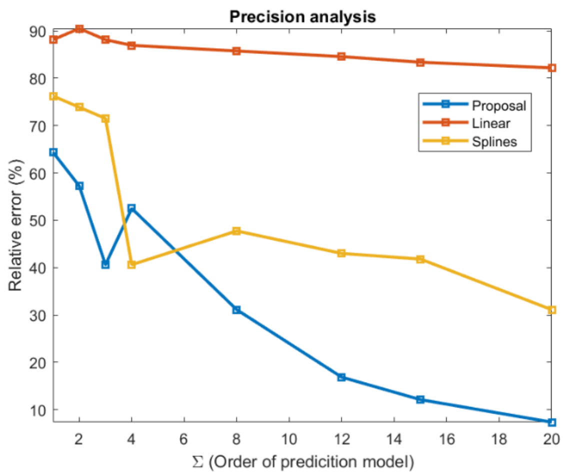

FIGURE 4. Precision analysis for different numbers of terms in the Laurent series. Entropy of communication signal: H = 0.25.

V. RESULTS

Figure 4 presents the mean precision for different scenarios, considering numbers of terms in the Laurent series Σ. Equally, when possible, the order of the state-of-the-art models were also changed in the same way. In this case (Figure 4) we are considering a signal with an entropy $H = \frac{1}{4}$.

As can be seen, linear regression (whose order is constant and immutable) shows an also constant error around 85%.

That's because a linear function is a very poor method to relate future samples with previous samples. If this method wants to be used, another type of variables should be considered. A slightly improved is shown, but this variation is caused by the statistical behavior of the stochastic processes.

Resting results show three different areas. For very low order (up to three), the correction phase of the proposed algorithm makes our solution to behave notoriously better than traditional schemes based on splines. However, splines may obtain a lower error with a lower number of terms in the metathetical model. Thus, for models with an order around three, splines present a better behavior. Finally, in the third, and wider, area our proposal is up to 25% more precise; as the proposed approach (once an enough number of terms are included in the Laurent series) is much more effective than existing schemes. In particular, for series (at least) considering the first twenty terms, the relative error is reduced below 10%. In order to show the behavior of the proposed solution, Figure 5 represents the original, received and reconstructed signals.

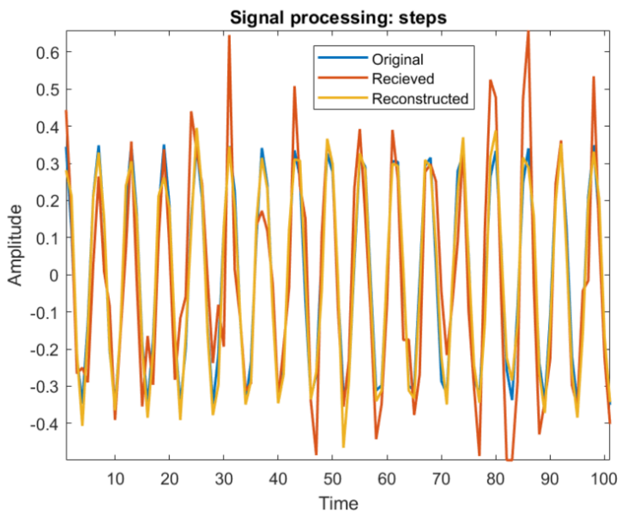

FIGURE 5. Comparison among the different steps in the signal processing and reconstruction process.

On the other hand, for Artificial Intelligence solutions, a very useful and common instrument to evaluate the performance is the confusion matrix. As, in the proposed mechanism, labels are not a discrete set but a continuous range; in order to make possible the calculation of the confusion matrix, we are dividing the global range $[\mu_i, -\mu_i]$ in four quartiles. Table 2 shows the obtained results. As can be seen, results are coherent with precision analysis in Figure 3. Models with a lower order present also a lower precision. For model of first order precision is around 35%. For models of eighth order, precision is around 70%, and for models of twentieth order precision goes around 93%. Besides, no significant differences are observed in results obtained for the different quartiles. Thus, the proposed mechanism is quite stable and homogenous in all its operating range.

TABLE 2. Confusion matrix.

Real value		Predicted value			
		Q1	Q2	Q3	Q4
$\Sigma = 1$	Q1	0.35	0.27	0.25	0.15
	Q2	0.25	0.33	0.25	0.23
	Q3	0.22	0.22	0.33	0.28
	Q4	0.18	0.18	0.17	0.34
$\Sigma = 8$	Q1	0.68	0.09	0.02	0.01
	Q2	0.16	0.70	0.18	0.10
	Q3	0.14	0.15	0.71	0.11
	Q4	0.02	0.06	0.08	0.68
$\Sigma = 20$	Q1	0.93	0.04	0.005	0.01
	Q2	0.04	0.89	0.015	0.03
	Q3	0.02	0.04	0.94	0.06
	Q4	0.01	0.03	0.03	0.9

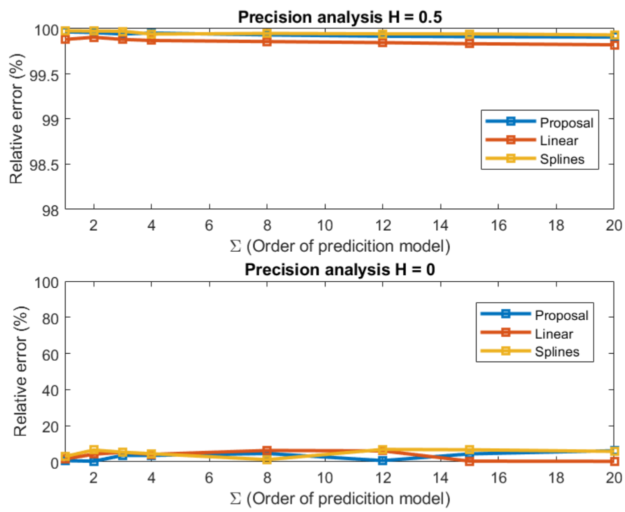

FIGURE 6. Precision analysis for different numbers of terms in the Laurent series. Entropy of communication signal: H = 0 and H = 0.5.

Figure 6 shows the results of this first experiment when considering communication signals with entropy values of $H = 0$ and $H = \frac{1}{2}$. As can be seen, in this case, the obtained results are not significant, as the input signal totally conditions the behavior of the global mechanism. Signals with null entropy are fixed sequences, so they can be predicted with no error using any simple algorithm (relative error is almost null). On the other hand, signals with entropy $H = \frac{1}{2}$ are random signals, and no underlying pattern may be detected or deducted. Thus, relative error is near 100%.

Now, an approach with a lower precision could be acceptable, if calculation delay and/or converge time were much lower. Figure 7 shows that analysis. In this case, as the calculation time is independent from the analyzed communication signal, we are using a signal with an entropy $H = \frac{1}{4}$.

As can be seen, both, the processing delay and the converge time for the training phase evolves linearly. In fact, as said in Section III, in respect to the unknown coefficients

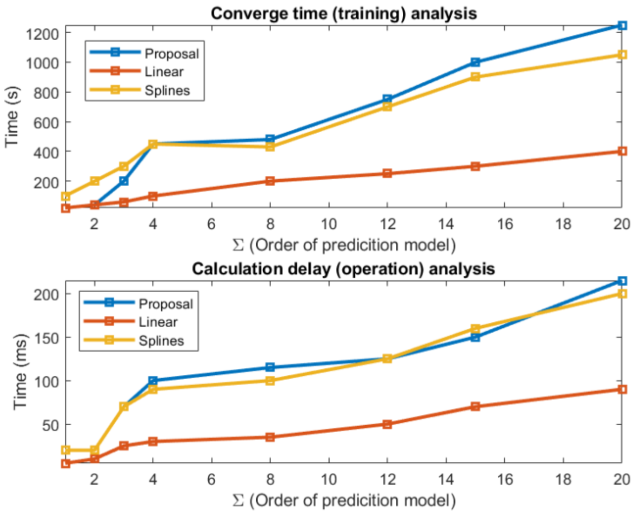

FIGURE 7. Converge time and calculation delay analysis for different numbers of terms in the Laurent series.

$b_{\lambda_0,\ldots,\lambda_T}$ models (both, splines and Laurent series) are linear equations, so as the number of variables grows linearly, the number of operations (sums and multiplications) also grows linearly.

Linear regression models are, once more, almost independent from the model order (there is a small variation caused by a higher number of equations, although the number of variables is the same). Besides, they are the fastest approaches (also the simplest ones), requiring only 200 seconds for training and below 100ms for prediction.

Resting models (splines and proposed Laurent series) present almost the same behavior. In the worst case, obtaining a sample requires 200ms, what is an acceptable time for real-time operation. Thus, as required time for training and operation is almost the same for both approaches, we can conclude the proposed approach based on Laurent series is better, as it presents a lower error.

VI. CONCLUSIONS

In this paper we proposed a new Artificial Intelligence predictor-corrector algorithm focused on improving the quality of data sequences and biological signals generated by mobile IoMT devices. The proposed algorithm includes two steps: a prediction phase and a correction procedure. The prediction phase analyzes signals in two domains, amplitude and frequency, thanks to the concept of complex envelope. In each domain a complex model based on unknown holomorphic complex functions and Laurent series is employed. On the other hand, correction procedure is supported by logical predicates, guaranteeing the global and local coherence of the information signal.

An experimental validation, supported by simulation scenarios and tools, is also provided. Results show the proposed approach presents a better precision than traditional approaches based on linear functions or splines.

Future works will integrate more complex decision algorithms in the reconstruction phase, in order to improve the global precision. In particular, we are investigating solutions based on Bayesian trees [44] and other specific solutions for biometric signals [45]. On the other hand, algorithms based on signal processing technologies for signal reconstruction are also being evaluated [46]. All these algorithms may be easily integrated into the proposed schemes, by replacing the current reconstruction technique for any of these proposals. Future works will evaluate the performance of these alternative implementations.

REFERENCES

[1] X. Ma, Y.-J. Wu, Y. Wang, F. Chen, and J. Liu, "Mining smart card data for transit riders' travel patterns," *Transp. Res. C, Emerg. Technol.*, vol. 36, pp. 1–12, Nov. 2013.

[2] D. N. Monekosso and P. Remagnino, "Data reconciliation in a smart home sensor network," *Expert Syst. Appl.*, vol. 40, no. 8, pp. 3248–3255, Jun. 2013.

[3] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for Internet of Things data analysis: A survey," *Digit. Commun. Netw.*, vol. 4, no. 3, pp. 161–175, Aug. 2018.

[4] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. San Mateo, CA, USA: Morgan Kaufmann, 2016.

[5] C. Costa and M. Y. Santos, "Improving cities sustainability through the use of data mining in a context of big city data," *Proc. Int. Conf. Data Mining Knowl. Eng., World Congr. Eng.*, in Lecture Notes in Engineering and Computer Science, vol. 1. London, U.K., Jul. 2015, pp. 320–325.

[6] S.-K.-S. Fan, C.-J. Su, H.-T. Nien, P.-F. Tsai, and C.-Y. Cheng, "Using machine learning and big data approaches to predict travel time based on historical and real-time data from taiwan electronic toll collection," *Soft Comput.*, vol. 22, no. 17, pp. 5707–5718, Sep. 2018.

[7] O. Y. Al-Jarrah, P. D. Yoo, S. Muhaidat, G. K. Karagiannidis, and K. Taha, "Efficient machine learning for big data: A review," *Big Data Res.*, vol. 2, no. 3, pp. 87–93, Sep. 2015.

[8] L. Zhou, S. Pan, J. Wang, and A. V. Vasilakos, "Machine learning on big data: Opportunities and challenges," *Neurocomputing*, vol. 237, pp. 350–361, May 2017.

[9] S. Ayoubi, N. Limam, M. A. Salahuddin, N. Shahriar, R. Boutaba, F. Estrada-Solano, and O. M. Caicedo, "Machine learning for cognitive network management," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 158–165, Jan. 2018.

[10] M. G. Kibria, K. Nguyen, G. P. Villardi, O. Zhao, K. Ishizu, and F. Kojima, "Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks," *IEEE Access*, vol. 6, pp. 32328–32338, 2018.

[11] Z. Obermeyer and E. J. Emanuel, "Predicting the future-big data, machine learning, and clinical medicine," *New England J. Med.*, vol. 375, no. 13, pp. 1216–1219, Sep. 2016.

[12] B. Bordel, R. Alcarria, G. Rizzo, and A. Jara, "Creating predictive models for forecasting the accident rate in mountain roads using VANETs," in *Proc. Int. Conf. Inf. Theoretic Secur.* Cham, Switzerland: Springer, Jan. 2018, pp. 319–329.

[13] B. Bordel, R. Alcarria, T. Robles, and D. Martín, "Cyber-physical systems: Extending pervasive sensing from control theory to the Internet of Things," *Pervas. Mobile Comput.*, vol. 40, pp. 156–184, Sep. 2017.

[14] B. Bordel, R. Alcarria, D. S. de Rivera, and T. Robles, "Process execution in cyber-physical systems using cloud and cyber-physical Internet services," *J. Supercomput.*, vol. 74, no. 8, pp. 4127–4169, Aug. 2018.

[15] B. Bordel, D. S. de Rivera, Á. Sánchez-Picot, and T. Robles, "Physical processes control in industry 4.0-based systems: A focus on cyber-physical systems," in *Ubiquitous Computing and Ambient Intelligence*. Cham, Switzerland: Springer, 2016, pp. 257–262.

[16] A. V. Vasilakos, Y. Zhang, and T. Spyropoulos, *Delay Tolerant Networks: Protocols and Applications*. Boca Raton, FL, USA: CRC Press, 2016.

[17] P. Ni, C. Zhang, and Y. Ji, "A hybrid method for short-term sensor data forecasting in Internet of Things," in *Proc. 11th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Aug. 2014, pp. 369–373.

[18] W. Derguech, E. Bruke, and E. Curry, "An autonomic approach to real-time predictive analytics using open data and Internet of Things," in *Proc. IEEE 11th Int. Conf. Ubiquitous Intell. Comput. IEEE 11th Int. Conf. Autonomic Trusted Comput. IEEE 14th Int. Conf. Scalable Comput. Commun. Associated Workshops*, Dec. 2014, pp. 204–211.

[19] F. Chen, P. Deng, J. Wan, D. Zhang, A. V. Vasilakos, and X. Rong, "Data mining for the Internet of Things: Literature review and challenges," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 8, Aug. 2015, Art. no. 431047.

[20] J. Siryani, B. Tanju, and T. J. Eveleigh, "A machine learning decision-support system improves the Internet of Things' smart meter operations," *IEEE Internet Things J.*, vol. 4, no. 4, pp. 1056–1066, Aug. 2017.

[21] F. Alam, R. Mehmood, I. Katib, and A. Albeshri, "Analysis of eight data mining algorithms for smarter Internet of Things (IoT)," *Procedia Comput. Sci.*, vol. 98, pp. 437–442, Jan. 2016.

[22] X. Zhu, F. Kui, and Y. Wang, "Predictive analytics by using Bayesian model averaging for large-scale Internet of Things," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 12, Dec. 2013, Art. no. 723260.

[23] A. Shilton, S. Rajasegarar, C. Leckie, and M. Palaniswami, "DP1SVM: A dynamic planar one-class support vector machine for Internet of Things environment," in *Proc. Int. Conf. Recent Adv. Internet Things (RIoT)*, Apr. 2015, pp. 1–6.

[24] L. Li, M. Rong, and G. Zhang, "An Internet of Things QoE evaluation method based on multiple linear regression analysis," in *Proc. 10th Int. Conf. Comput. Sci. Educ. (ICCSE)*, Jul. 2015, pp. 925–928.

[25] M. Jiang, J. Luo, D. Jiang, J. Xiong, H. Song, and J. Shen, "A cuckoo search-support vector machine model for predicting dynamic measurement errors of sensors," *IEEE Access*, vol. 4, pp. 5030–5037, 2016.

[26] L. Du, Y. Du, Y. Li, J. Su, Y.-C. Kuan, C.-C. Liu, and M.-C.-F. Chang, "A reconfigurable streaming deep convolutional neural network accelerator for Internet of Things," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 1, pp. 198–208, Jan. 2018.

[27] W. Wu and M. Peng, "A data mining approach combining K-means clustering with bagging neural network for short-term wind power forecasting," *IEEE Internet Things J.*, vol. 4, no. 4, pp. 979–986, Aug. 2017.

[28] J. S. Kumar and M. A. Zaveri, "Hierarchical clustering for dynamic and heterogeneous Internet of Things," *Procedia Comput. Sci.*, vol. 93, pp. 276–282, Jan. 2016.

[29] P. Rathore, D. Kumar, S. Rajasegarar, M. Palaniswami, and J. C. Bezdek, "A scalable framework for trajectory prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3860–3874, Oct. 2019.

[30] M. Imani, Y. Kim, T. Worley, S. Gupta, and T. Rosing, "HDCluster: An accurate clustering using brain-inspired high-dimensional computing," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2019, pp. 1591–1594.

[31] H. Bangui, M. Ge, and B. Buhnova, "Exploring big data clustering algorithms for Internet of Things applications," in *Proc. 3rd Int. Conf. Internet Things, Big Data Secur.*, 2018, pp. 269–276.

[32] S. Banerjee, T. Chattopadhyay, A. Pal, and U. Garain, "Automation of feature engineering for IoT analytics," *ACM SIGBED Rev.*, vol. 15, no. 2, pp. 24–30, Jun. 2018.

[33] X. Luo, J. Liu, D. Zhang, and X. Chang, "A large-scale Web QoS prediction scheme for the industrial Internet of Things based on a kernel machine learning algorithm," *Comput. Netw.*, vol. 101, pp. 81–89, Jun. 2016.

[34] H. Li, K. Ota, and M. Dong, "Learning IoT in edge: Deep learning for the Internet of Things with edge computing," *IEEE Netw.*, vol. 32, no. 1, pp. 96–101, Jan. 2018.

[35] X. Deng, P. Jiang, X. Peng, and C. Mi, "An intelligent outlier detection method with one class support tucker machine and genetic algorithm toward big sensor data in Internet of Things," *IEEE Trans. Ind. Electron.*, vol. 66, no. 6, pp. 4672–4683, Jun. 2019.

[36] Y. Qin, Q. Z. Sheng, N. J. G. Falkner, S. Dustdar, H. Wang, and A. V. Vasilakos, "When things matter: A survey on data-centric Internet of Things," *J. Netw. Comput. Appl.*, vol. 64, pp. 137–153, Apr. 2016.

[37] P. K. Agarwal, J. Gurjar, A. K. Agarwal, and R. Birla, "Application of artificial intelligence for development of intelligent transport system in smart cities," *J. Traffic Transp. Eng.*, vol. 1, no. 1, pp. 20–30, 2015

[38] I. Kotenko, I. Saenko, F. Skorik, and S. Bushuev, "Neural network approach to forecast the state of the Internet of Things elements," in *Proc. 18th Int. Conf. Soft Comput. Meas. (SCM)*, May 2015, pp. 133–135.

[39] K. Guo, Y. Lu, H. Gao, and R. Cao, "Artificial intelligence-based semantic Internet of Things in a user-centric smart city," *Sensors*, vol. 18, no. 5, p. 1341, Apr. 2018.

[40] S. Hu, "Research on data fusion of the Internet of Things," in *Proc. Int. Conf. Logistics Inform. Service Sci. (LISS)*, Jul. 2015, pp. 1–5.

[41] Z. Ning, P. Dong, X. Wang, X. Hu, L. Guo, B. Hu, Y. Guo, T. Qiu, and R. Y. Kwok, "Mobile edge computing enabled 5G health monitoring for Internet of medical things: A decentralized game theoretic approach," *IEEE J. Sel. Areas Commun.*, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9004568/references#references

[42] A. Nayyar, V. Puri, and N. G. Nguyen, "Biosenhealth 1.0: A novel Internet of medical things (IOMT)-based patient health monitoring system," in *Proc. Int. Conf. Innov. Comput. Commun.* Singapore: Springer, 2019, pp. 155–164.

[43] R. Alcarria, B. Bordel, M. Á. Manso, T. Iturrioz, and M. Pérez, "Analyzing UAV-based remote sensing and WSN support for data fusion," in *Proc. Int. Conf. Inf. Theoretic Secur.* Cham, Switzerland: Springer, Jan. 2018, pp. 756–766.

[44] S. Sahoo, A. Subudhi, M. Dash, and S. Sabut, "Automatic classification of cardiac arrhythmias based on hybrid features and decision tree algorithm," *Int. J. Autom. Comput.*, pp. 1–11, Mar. 2020, doi: 10.1007/s11633-019-1219-2.

[45] S. Türkcan and J.-B. Masson, "Bayesian decision tree for the classification of the mode of motion in single-molecule trajectories," *PLoS ONE*, vol. 8, no. 12, Dec. 2013, Art. no. e82799.

[46] B. Bordel and R. Alcarria, "Assessment of human motivation through analysis of physiological and emotional signals in industry 4.0 scenarios," *J. Ambient Intell. Humanized Comput.*, pp. 1–21, Dec. 2017, doi: 10.1007/s12652-017-0664-4.

[47] A. Bultheel, *Laurent Series and Their Padé Approximations*, vol. 27. Basel, Switzerland: Birkhäuser, 1987.

[48] T. O. Ayodele, "Types of machine learning algorithms," in *New Advances in Machine Learning*. London, U.K.: IntechOpen, 2010, pp. 19–48.

BORJA BORDEL received the B.S. and M.S. degrees in telecommunication engineering and the Ph.D. degree from the Technical University of Madrid, in 2012, 2014, and 2018, respectively. He is currently an Assistant Professor at the Computer Science School, Technical University of Madrid. His research interests include cyber-physical systems, wireless sensor networks, radio access technologies, communication protocols, and complex systems.

RAMÓN ALCARRIA (Member, IEEE) received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid, in 2008 and 2013, respectively. He is currently an Associate Professor at the Department of Geospatial Engineering, Technical University of Madrid. His research interests are service architectures, sensor networks, human-computer interaction, and prosumer environments. He has been involved in several research and development European and National projects related to future internet, the Internet of Things, and service composition.

TOMAS ROBLES received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid, in 1987 and 1991, respectively. He is currently a Full Professor of telematics engineering at the E.T.S.I. Telecommunication, Technical University of Madrid. His research interests include advanced applications and services for wireless networks and blockchain-based infrastructures.

ILSUN YOU (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer science from Dankook University, Seoul, South Korea, in 1997 and 2002, respectively, and the second Ph.D. degree from Kyushu University, Japan, in 2012. From 1997 to 2004, he was with THIN Multimedia Inc., Internet Security Company Ltd., and Hanjo Engineering Company Ltd., as a Research Engineer. He is currently an Associate Professor at the Department of Information Security Engineering, Soonchunhyang University. Especially, he has focused on 4/5G security, security for wireless networks and mobile internet, IoT security, and so forth while publishing more than 180 articles in these areas. He is a Fellow of the IET. He has served or is currently serving as the general chair or the program chair of international conferences and workshops, such as WISA'19-20, MobiSec'16-19, AsiaARES'13-15, MIST'09-17, and MobiWorld'08-17. He is the Editor-in-Chief (EiC) of *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* (JoWUA). He is in the Editorial Board for *Information Sciences* (INS), *Journal of Network and Computer Applications* (JNCA), IEEE Access, *Intelligent Automation & Soft Computing* (AutoSoft), *International Journal of Ad Hoc and Ubiquitous Computing* (IJAHUC), *Computing and Informatics* (CAI), and *Journal of High Speed Networks* (JHSN).