Missing Data Imputation with Bayesian Maximum Entropy for Internet of Things Applications

Aurora González-Vidal, Punit Rathore *Member, IEEE*, Aravinda S. Rao, *Member, IEEE*, José Mendoza-Bernal, Marimuthu Palaniswami *Fellow, IEEE* and Antonio F. Skarmeta-Gómez *Member, IEEE*

***Abstract*—Internet of Things (IoT) enables the seamless integration of sensors, actuators and communication devices for real-time applications. IoT systems require good quality sensor data in order to make real-time decisions. However, values are often missing from the sensor data collected owing to faulty sensors, a loss of data during communication, interference and measurement errors. Considering the spatiotemporal nature of IoT data and the uncertainty of the data collected by sensors, we propose a new framework with which to impute missing values utilizing Bayesian Maximum Entropy (BME) as a convenient means to estimate the missing data from IoT applications. Missing sensor measurements adversely affect the quality of data, and consequently the performance and outcomes of IoT systems. Our proposed framework incorporates BME in order to impute missing values in diverse IoT scenarios by making use of the combination of low- and high-precision sensors. Our approach can incorporate the measurement errors of low-precision sensors as interval quantities along with the high-precision sensor measurements, making it highly suitable for real-time IoT systems. Our framework is robust to variations in data, requires less execution time, and requires only a single input parameter, thus outperforming existing IoT data imputation methods. The experimental results obtained for three IoT datasets demonstrate the superiority of the BME framework as regards accuracy, running time and robustness. The framework can additionally be extended to distributed IoT nodes for the online imputation of missing values.**

***Index Terms*—Internet of Things (IoT), missing data, imputation, Bayesian Maximum Entropy (BME), spatio-temporal analysis**

I. Introduction

Advances in sensing phenomena, memory capacity, computational power and wireless communications have led to significant advances in the way in which we interact with our surroundings, people and businesses. Internet of Things (IoT) enables the seamless integration of sensors, actuators, and communication devices for real-time sensing, communication and the remote control of actuators [1]. There are currently about 26 billion IoT devices worldwide (up from 15 billion in 2015) and this number is projected to reach a massive 75 billion by 2025 [2]. This exponential increase in the number of devices is set to produce copious amounts of structured, semi-structured, unstructured and real-time data from billions of homogeneous and heterogeneous devices, resulting in *Big Data* [3]. This increase in the number of devices is possible as a result of cheap sensor nodes (that often have lower precision) and inexpensive computation capabilities.

The interconnected nature of sensors, actuators and infrastructure systems provides advanced solutions through the use of artificial intelligence (AI) and *Big Data* analytics across a wide range of sectors [4]. IoT's fundamental strength is derived from its ability to uniquely identify "things" (or the devices) and turns them into "smart objects" with low-to-moderate computing and communication capabilities. The capabilities are further enhanced by utilizing Edge, Fog and Cloud computing paradigms, along with visualization technologies. These developments have enabled several IoT applications to facilitate the emergence of the smart city [5] including building automation and energy efficiency in buildings [6], [7], precision agriculture [8], autonomous cars [9] and safer transportation mechanisms [10], managing water resources [11], [12] and the monitoring of structural health [13]. For example, the smart grid solution enables consumers to check their energy usage in real time, primarily thanks to sensors monitoring energy usage, which analyze the usage and communicate it to consumers. This helps consumers make savings as regards their energy bills while simultaneously reducing energy consumption from the power grid. This win-win solution is principally possible because of IoT. For applications involving real time decision-making, the quality of the spatiotemporal data is of the utmost importance. Obtaining reliable measurements with the deployment of low-cost sensors can be achieved by deploying a mix of fewer high-precision sensors at low spatial resolutions and a larger number of inexpensive, low-precision sensors at high spatial resolutions over an area of interest [14]. This strategy of using a combination of low- and high-precision sensors not only reduces the cost of networking, but also extends the reach of the network and provides opportunities for varieties of IoT applications.

However, values are often missing from the sensor data collected. There are a variety of reasons for missing data: (i)

This work has been sponsored by MINECO through the PERSEIDES project (ref. TIN2017-86885-R) and by the European Comission through the H2020 IoTCrawler (contract 779852), and DEMETER (grant agreement 857202) EU Projects and also co-financed by the European Social Fund (ESF) and the Youth European Initiative (YEI) under the Spanish Seneca Foundation (CARM). This work is also supported by Australian Research Council (ARC) Discovery Project (DP190102828).

A. González-Vidal, J. Mendoza-Bernal and A F. Skarmeta-Gómez are with the Department of Information and Communications Engineering, University of Murcia, Spain ({aurora.gonzalez2, jose.mendoza, skarmeta}@um.es)

Punit Rathore is with the Senseable City Lab, Department of Urban Planning and Studies, Massachusetts Institute of Technology, Cambridge, MA, USA. (E-mail: prathore@mit.edu). This work was performed when P. Rathore was with the Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, VIC - 3010, Australia.

A S. Rao and M. Palaniswami are with the Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, VIC - 3010, Australia (e-mail: aravinda.rao@unimelb.edu.au, palani@unimelb.edu.au).

faulty sensors producing intermittent readings, (ii) the loss of data during wireless communication owing to packet loss, (iii) the loss of data owing to interference in the communication medium, leading to unusable or unrecognizable values, and (iv) data removed purposely by attackers with malicious intentions during sensing, processing, storing or communication. The research challenge is to impute (in addition to assigning or representing) the missing values in order to enable the data to be analyzed while ensuring that the imputed values are as close as possible to the true values. The imputation of missing data in IoT is an important challenge as the data is diverse, and the techniques developed must, therefore, be robust to scale and provide a high level of confidence for different types of applications. Techniques must additionally be lightweight in order to cater for real-time IoT application requirements.

In this paper, we propose a new framework with which to impute missing values in IoT environments using Bayesian Maximum Entropy (BME). We demonstrate the use of our framework in three IoT scenarios: (1) an indoor office setting, measuring the temperature at Intel Berkeley Research Lab (IBRL), (2) outdoor weather data, measuring humidity at Docklands Library (situated along the harbor front in the City of Melbourne, Australia), and (3) outdoor weather station data, measuring the water temperature along the lakefront of Lake Michigan, Chicago. Our proposed scheme outperforms existing schemes as regards accuracy, execution time and robustness. The main contributions in the proposed framework are:

- A new missing data imputation framework is proposed that incorporates BME in order to impute missing values in diverse IoT scenarios by making use of a combination of low- and high-precision sensors.
- We demonstrate the robustness of our approach by validating it with three different real-world IoT datasets. Our approach for the imputation of missing data is robust to variations in high-variance data and outperforms existing state-of-the-art methods.
- We also demonstrate how our proposed framework requires significantly less execution time, making it highly suitable for real-time IoT systems. Our framework maintains lower and stable execution times when the percentage of missing values increases.
- Our framework also requires fewer input parameters, thus making it suitable for distributed IoT systems. In fact, we show that the value of this input parameter does not affect the performance of our approach to any great extent.

The main idea in our framework is to allow processes that depend on online data to continue functioning normally even if there are missing values. We achieve this by employing a method that is superior to other state-of-the-art methods in terms of accuracy, and which is better adapted to dynamic scenarios.

II. Related Work

The data in IoT applications are often used in clustering, classification, or prediction problems to extract useful information. These data should be clean and complete in order to use them for reliable decision making. However, missing data in IoT networks severely affects the data quality. Related literature contains four principal broad categories of data imputation techniques [15]: (1) deletion of missing data, (2) imputation or estimation of missing data using statistical methods and/or machine learning, (3) estimating the missing values on the basis of modeling the known distribution (such as Expectation-Maximization, Gaussian Mixture Models), and (4) classifying data that contains missing data by means of machine learning (ML), wherein the ML model handles missing data without explicitly providing the missing information.

Missing data is, moreover, broadly categorized into 3 groups [15], [16]: (1) missing completely at random (MCAR), (2) missing at random (MAR), and (3) missing not at random (MNAR). An MCAR is a missing data mechanism in which the missing value is independent of the variable itself, *i.e.*, the missing value is not influenced by any external factors, but is unavailable owing to random events; this could, for example, be owing to a sensor node breaking down because of an accident. In the case of MAR, the missing variable is independent, but can nevertheless be predicted using data variables *i.e.*, the missing value is influenced by other factors, such as sensor failings during a cleaning event, when the power supply to the sensors was disturbed. In the case of MNAR, the missing values are dependent on the variable itself and the event is non-random.

A. Imputations of Missing Data in WSN

Using spatiotemporal correlations as a basis, k-nearest neighbor (k-NN) [17] was was adopted in order to estimate the missing data in wireless sensor networks (WSN) [18]. The k-NN is a non-parametric method used in clustering, regression and/or classification tasks. in [18], the missing data were estimated by employing spatial correlations among the k-NN and using a linear regression model, whereas the temporal relations were ignored. There are also examples of the use of Compressed Sensing (CS) to recover missing data from WSN in literature. CS is a signal processing technique in which the data is recovered by utilizing the signal sparsity from fewer samples, adhering to the Nyquist-Shannon sampling theorem [19]. Missing values are recovered by applying the principles of CS to WSN data, which featured spatial correlation, temporal stability and low-rank structure [20].

Several machine learning models have also been introduced to address the missing value problem, such as Neural Networks [21], Generative Adversarial Networks (GANs) [22] and the sparse auto-encoder, which was modified in order to deal with missing values and was used along with the Restricted Boltzmann Machine [23]. These methods tend to introduce bias into the models learned when data are limited or are of poor quality. The preliminary work [24] tested several black box models in order to interpolate missing values from the specific context of buildings. It was found that the probability distribution of the lengths of the gaps in the data closely follows a log-logistic distribution, despite the diverse causes of missing values (occupants interfering with the sensors, disconnected wifi or software problems).

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

Recommender systems use *collaborative filtering* to gather information from multiple users (or agents) in order to recommend a choice for users. Netflix, YouTube, and Spotify are some examples of recommender systems. Collaborative filtering is also useful as regards predicting missing values. In [25], spatiotemporal correlations were captured by grouping the sensor nodes, after which matrix factorization was utilized to learn latent factors so as to predict missing values. However, this type of system is vulnerable as regards being biased towards predicting the view of the majority.

B. Imputation of Missing Data in IoT Systems

IoT systems largely differ from WSN in that WSN were primarily used to acquire data, process small amounts of data, transmit data from leaf nodes to clusters to be aggregated and, occasionally, to send control signals to actuators. External interventions by humans were required in order to make decisions based on the WSN data and actuate certain devices. However, IoT systems subsume not only the WSN system, but also (i) Radio Frequency Identification (RFID) tags for the identification of objects (or devices); (ii) middleware for software services; (iii) cloud and edge computing in order to provide services over the Internet, and (iv) the enabling of machine-to-machine (M2M) and Human Computer Interaction (HCI) [26]. In other words, IoT systems integrate devices with intelligence, decision-making capabilities and also humans into the loop.

As the complexity and nature of the IoT systems are significantly different from those of WSN, the techniques proposed for WSN may not be directly applicable, as IoT and WSN differ in many aspects of the application requirements. From a technical standpoint, we have to be aware of the following key aspects [26]: scalability, robustness, quality of service, heterogeneity of devices and networks, deployment and coverage, mobility, power management, identification of devices, autonomous networking, data management, communication, and security and privacy.

Yan *et al.* [27] proposed a Gaussian Mixture Model (GMM) to handle missing values in IoT systems. These authors proposed the recovery of 21 missing temperature sensor values from a set of 220 observations. Their experiments do not reveal how many distributions were present in the data and neither do the authors provide rationale for adopting GMM. Mary *et al.* [28] proposed an extended spatial and temporal correlated proximate (ESTCP) model with which to impute missing data from the City Pulse [1] air pollution dataset consisting of $17,569$ observations (sampled every 5 minutes). ESTCP mainly employs time lagged correlations to impute the missing values, and it is consequently suitable for datasets with temporal correlations.

Collaborative filtering (or recommender systems) is used to predict users' preferences on the bases of their historical preferences, including samples or choices obtained from people among the population [29], [30]. We often see this on a daily basis in e-commerce and movie recommendation systems, such as Netflix, Amazon, YouTube and others. The idea is to enable collaborative filtering techniques to impute the missing value [31], [32]. One of the major drawbacks of collaborative filtering is that it cannot efficiently handle very large datasets or large numbers of users [33]. The issue of handling large datasets and number of users, was addressed through the introduction of Probabilistic Matrix Factorization (PMF). PMF is a decomposition technique in which a given matrix is decomposed into two low-rank matrices [33]. Fekade *et al.* [34] proposed an extension of PMF in order to recover missing data in IoT networks. The advantage of PMF is that it scales linearly with the number of observations and also performs well in the case of large, sparse, and very imbalanced data [33]. This is important in the case of IoT systems as they accumulate large datasets from multiple users over the course of time. It should be noted that PMF techniques do not naturally incorporate the location of the data, but that they have to be incorporated when PMF is used in an IoT system. For instance, spatial information was considered by clustering the sensors according to their locations using the well-known k-means algorithm, while the PMF algorithm was later applied to the sensor data from each of the clusters [34].

Kriging is another important technique that is commonly used to impute missing values. The objective of Kriging techniques is to interpolate the intermediate value (or point) by following a Gaussian process in the neighborhood of the point. The Kriging family has traditionally been one of the most popular techniques for the analysis of the spatial characteristics of an attribute when compared to several other geostatistical methods [35]. When applied to geostatistical mapping, Kriging takes into account the attribute's mean trends, spatial covariance or semi-variance, and observed attribute values. Kriging originated in the areas of mining and geostatistics, which involve spatially and temporally correlated data and has become a generic methodology for several closely related least-squares methods that provide Best Linear Unbiased Predictions (BLUP) and also some non-linear types of prediction. Kriging models are able to predict the missing values at new locations from which data has not been acquired optimally, *i.e.*without bias and with minimum variance. We find the use of Kriging techniques in [36], [37], where it is used to impute missing data; however, Kriging techniques have proven to be sub-optimal and restrictive when modelling complex spatiotemporal data. Moreover, Kriging techniques do not consider the measurement errors of low-precision sensors in estimation [38], which limits the use of Kriging in IoT applications, in which the majority of the nodes will have low-precision sensors.

C. Probability Matrix Factorization (PMF) for Missing Data Imputation

As mentioned earlier, very little research has been conducted in order to address missing data imputation in IoT systems. The PMF-based method [34] is the state-of-the-art technique for missing data imputation in IoT that we compare in our experiments with our proposed framework. Before moving onto our proposed framework, we briefly explain the PMF method [34] in order to provide a better understanding of how

[1] http://iot.ee.surrey.ac.uk:8080/

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

our proposed framework is different. The steps in PMF [34] are:

1) Normalize the observations of each sensor by re-scaling the observations in the range (0, 1):

$$z_i = \frac{x_i - min(x)}{max(x) - min(x)}, \tag{1}$$

where i indicates the timestamp, x_i is the data value at time i and z_i is the normalised observation at time i.

2) Represent the normalized dataset as a matrix **R** with dimensions $N \times M$, where N is the number of rows representing sensors and M is the number of columns representing observations. It is assumed that **R** follows a Gaussian distribution as the data has been normalized:

$$R = \begin{pmatrix} R_{11} & \dots & R_{1M} \\ \vdots & \ddots & \vdots \\ R_{N1} & \dots & R_{NM} \end{pmatrix} \tag{2}$$

3) Generate random matrices **U** $[N \times K]$ and **V** $[M \times K]$, where each row follows a Gaussian distribution with mean $\mu = 0$ and a small standard deviation (σ_U for U and σ_V for V). K represents the number of latent feature column-vectors.

4) Let **I** be a matrix that indicates the positions of missing values in **R** as follows:

$$I_{ij} = \begin{cases} 1, & \text{if } I_{ij} \text{ is a known value} \\ 0, & \text{if } I_{ij} \text{ is a missing value} \end{cases} \tag{3}$$

5) Compute the root mean square error (RMSE), where a quadratic regularization is added as follows:

$$\text{RMSE}_q = \sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}\left(R_{ij} - U_i {V_j}^T\right)^2 - \sum_{i=1}^{N} \lambda_U ||U||^2 + \sum_{j=1}^{M} \lambda_V ||V||^2 \tag{4}$$

The regularization parameters λ_U and λ_V will control the magnitudes of updated matrices U and V and this will, in turn, help to avoid overfitting the data.

6) Update the values of **U** and **V** such that:

$$U'_{ij} = U_{ij} + \alpha \frac{\partial RMSE_{q_{ij}}}{\partial U_i} \tag{5}$$

$$V'_{ij} = V_{ij} + \alpha \frac{\partial RMSE_{q_{ij}}}{\partial V_j} \tag{6}$$

where α is the slope value that defines how much of **U** and **V** need to be adjusted. Choosing the right α is important to attain convergence.

7) After some iterations (when the values attain the RMSE threshold or the maximum number of iterations is obtained), the missing values comprising the matrix product are extracted using the previously defined identity matrix given by I in (4).

8) Finally, the values are converted back or unnormalized in order to obtain the estimation on the right scale.

The computational complexity of PMF might seem high owing to the matrix multiplications; however, there are many calculations that can be avoided. Only those calculations that are used to obtain the components of the original matrix R that are missing are required. In this case, if n is the number of missing values, then the complexity of PMF is $O(kn)$ for each iteration in the model.

D. Limitations of PMF in IoT Use Cases

The adoption of PMF with the purpose of imputing data has several drawbacks in dynamic IoT scenarios. Some of the major limitations that make PMF unsuitable for IoT scenarios are listed below:

1) PMF is an iterative procedure. It may get stuck in a local minimum, thus leading to sub-optimal solutions.

2) PMF does not naturally incorporate the location of the data. This problem is solved by using a clustering technique whose choice does not appear to cover all the possibilities in IoT systems. It is not always true that geographically close sensors are always related to each other. For example, rooms with different purposes might be closer in measurements than others that are closer in space *i.e.*, a personal office could be closer to a meeting room or the library than to other personal office that might encounter similar usage patterns and sensors might, therefore, provide a similar value.

3) PMF cannot be used for Big Data or large-scale sensor networks owing to the vastness of data and the computational complexity of the matrix operations. In our experiments, we show that PMF requires more execution time as the number of missing values increases.

4) PMF fails to incorporate uncertainty in the measurements present in inexpensive low-precision sensor measurements. This is again a significant drawback, as IoT systems will include low-precision sensors.

In the following section, we introduce our BME-based generic framework, which addresses many of the issues encountered in the PMF and Kriging approaches whose purpose is to solve the problem of missing values.

III. Our Approach: A BME Framework for the Recovery of Missing Data

In our framework, we utilize the spatiotemporal characteristics of the IoT data in order to impute missing values. We employ a knowledge-based BME model in our framework, since this model is better than the traditional stochastic estimation methods [39]. BME is a mapping method for spatiotemporal estimation [40] that allows various knowledge bases to be incorporated in a logical manner—definite rules for prior information, hard and soft data into modeling [38] . This does not occur with other approaches, which do not incorporate prior information such as knowledge of the physics of the phenomena involved, geological interpretations and experience of similar site conditions into the modeling. To illustrate this point, we utilized BME for the spatiotemporal modeling of a park's humidity, the indoor temperature of an office and the water temperature in a lake in different parts of the world.

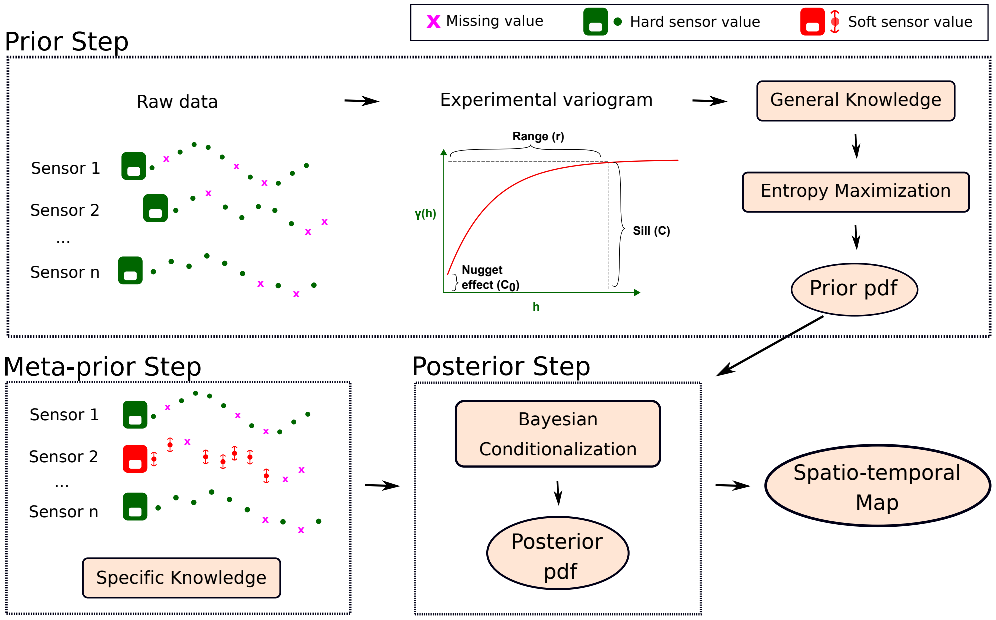

Fig. 1. Missing sensor data imputation using BME: (1) The prior step extracts the general knowledge from an experimental variogram (computed using data from all sensors) and later entropy maximization is applied to compute prior probability density function (pdf); (2) The meta-prior step extracts specific knowledge from the hard and soft sensors; (3) The posterior step uses Bayesian conditionalization to obtain the posterior pdf for creating spatiotemporal maps.

BME has, in the past, been used to estimate ozone [41], for the risk assessment of soil pollution [42], to optimize the deployment of solar monitoring stations [43] and to estimate particulate matter (PM) 2.5 [44]. An introduction to the BME method and its ability to incorporate prior information and measurement uncertainties in its estimation model is provided below.

A. BME modelling

Let $x_{data} = [x_1, x_2, ..., x_n]^T$ denote the set of physical variable x measured at locations s_i (where $i = 1, 2, ..., n$ denoting the n geographical locations). Physical data points can be of two kinds:

- *Hard data* is obtained from reliable high-precision sensors. Examples of *hard data* include thermostat values, weather monitoring stations, etc. In this work, we denote *hard data* as $x_{hard} = [x_{1_h}, x_{2_h}, ..., x_{n_h}]^T$ measured at locations $s_h = [s_{1_h}, ...s_{n_h}]$.
- *Soft data*. Soft data is the data that may have some uncertainties. These include observations, opinions, knowledge, experience, etc. Soft data can be represented by intervals or in a probabilistic form. In our work, we use sensor measurements in the form of interval data. We denote *soft data* as $x_{soft} = [x_{1_l}, ..., x_{n_l}]^T$ measured at locations $s_s = [s_{1_s}, ...s_{n_s}]$, such that the soft data value x_i lies within a known interval I_i, that is: $x_i \in I_i = [l_i, u_i]$.

The physical knowledge regarding a natural process used by BME [45] can be divided into general knowledge K_G (such as a scientific law and summary statistics) and specificatory knowledge, K_S (obtained through experience and specific situations associated with physical data points).

In this respect, $x_{data} = \{x_{hard} \cup x_{soft}\}$ and the total knowledge is $K = \{K_G \cup K_S\}$. In this work, we estimate the realization x_k at a location $s_k \in s_E$, where s_E is a set of locations at which data is missing. These realizations are then used to generate a spatiotemporal map $x_{map} = x_{data} \cup x_E$ at locations $s_{map} = \{s_1, s_2, ...s_n, s_E\}$.

BME modeling includes three stages of knowledge acquisition, integration and processing: (i) a prior stage, (ii) a pre-posterior (or meta-prior) stage, and (iii) a posterior stage. Each stage processes certain knowledge and data as shown in Fig. 1.

*a) **Prior Stage:*** BME uses prior information as auxiliary constraint information to guide towards a more accurate spatiotemporal prediction. This prior information could comprise a physical law or a principle that is applicable to the natural process of interest [46]. In order to integrate this general knowledge (K_G), BME utilizes the Shannon information entropy (1948) [47] and the Maximum Entropy principle. The Shannon information entropy is the average amount of information produced by a stochastic source of data and formally is expressed as: $I(x_{map}) = -log_2(\Phi_G(x_{map}))$, where x_{map} is the realization of an ST map and $\Phi_G(x_{map})$ is the prior PDF model, which refers to the general knowledge K_G. The expectation or entropy (H) contained in the prior PDF can, therefore, be expressed as follows:

$$\begin{aligned} H(x_{map}) &= \int \Phi_G(x_{map}) I(x_{map})\ dx_{map} = \\ &= -\int \Phi_G(x_{map}) log_2(\Phi_G(x_{map}))\ dx_{map} \end{aligned} \tag{7}$$

Since the Maximum Entropy principle is followed, the best model is that which allows the most uncertainty (entropy) from the data. The shape of prior PDF is, therefore, derived by maximizing the entropy $(H(x_{map}))$ and by taking the following constraint into consideration:

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

$$\overline{g_\alpha} = \int g_\alpha(x_{map})\Phi_G(x_{map})\ dx_{map},\ \alpha = 1, ..., N_p, \quad (8)$$

where g_α are properly chosen functions, signifying that their expectations $\overline{g_\alpha}$ provide the space/time statistical moments of interest (by convention $g_0 = 1$ and $\overline{g_0} = 1$). The N_p value is chosen in such a way that the stochastic moments included with N_p involve all points. For example, by choosing g_α as mean and covariance function with $g_0 = 1$, the total number of constraints required is $N_p = 1 + (n + 1)(n + 4)/2$, where n represents the total number of geographical locations.

The prior knowledge regarding the monitoring area can be obtained in the form of the mean, the covariance function or any other higher order moments in the space-time domain.

The mean function $\overline{x_{map}}$ of the spatiotemporal random field (the bar denotes stochastic expectation) characterizes trends and systematic structures in space/time; and the space-time variability of $\overline{x}$ can be expressed in terms of a centered covariance function as:

$$C_{map} = H[(x_{map}(p) - \overline{x_{map}}(p))(x_{map}(p') - \overline{x_{map}}(p'))]\ \forall p, p' \quad (9)$$

In this work, the mean and the covariance function are used as general prior knowledge. An exponential function is used to model a spatio-temporal covariance structure known as a variogram. A variogram (or a semi-variogram) [48] is an experimental function used to determine spatial correlations in observations measured in a defined area. This provides a means to analyze how one point has an influence on other points in different spatial and time separations (lags). We provide more details about variograms in Section IV.

The optimization for this maximization is done using the Lagrange multipliers λ_α. In this respect, the prior PDF can be expressed as $\Phi_G(x_{map}) = H^{-1}e^{\Theta_G[x_{map}]}$, where $H = e^{-\lambda_0}$ is a normalization constant, Θ_G represents the operator processing the general knowledge K_G and is given by $\sum_{\alpha=1}^{N_c} \lambda_\alpha g_\alpha(x_{map})$.

*b) **Meta-prior Stage:*** This stage considers the specificatory knowledge including hard and soft physical data points. In this work, we use interval I to express the soft data, where interval ranges are defined using the measurement error/uncertainty of low precision sensors.

*c) **Posterior Stage:*** In this last stage, both knowledge bases (general and specificatory) are integrated with the objective of maximizing the posterior PDF, given total knowledge K. The prior PDF is updated by taking the site-specific knowledge into consideration. The conditional PDF is expressed in terms of prior PDF (prior stage), specific knowledge (meta-prior stage) and information available at a posterior stage as follows:

$$\Phi_K(x_k|x_{data}) = J^{-1}\Theta_S[\Phi_G(x_{map}), x_{soft}] \quad (10)$$

where J is the normalization parameter given by:

$$J = \Theta_S[x_{soft}, \Phi_G(x_{data})] = \int_I \Phi_G(x_{data})\ dx_{soft} \quad (11)$$

and Θ_S represents the posterior operator that incorporates the soft data.

The posterior PDF provides a complete statistical distribution of the estimation situation, which can be used to obtain different estimators, including the mean and mode of PDF with estimation uncertainty. The missing values are estimated by maximizing the posterior PDF with respect to x_E such that the BME estimate minimizes the mean squared estimation error. In this work, we compute the mean estimate by employing the updated PDF of a missing value.

B. Advantages of proposed BME framework

The main advantages of the BME method over the state-of-the art methods PMF and Kriging are highlighted below.

BME vs Kriging:

- BME does not assume that the data follow a certain distribution, whereas Kriging does assume that data follow a normal distribution, since they are driven by Gaussian processes.
- BME can use the stochastic moments of any order, whereas Kriging is restricted to first and second order moments
- BME can incorporate uncertain data, given its characteristics, whereas kriging has provided limited results when carrying out tasks of this type.

BME vs PMF:

- BME can incorporate physical knowledge of the phenomena under study.
- BME naturally incorporates the location of the sensors as part of its process, and benefits from this information.
- Once the variograms have been computed, the complexity of BME is linear and independent from past measures, which makes it suitable for fast Big Data scenarios and edge computing.
- BME incorporates uncertain data intrinsically, while PMF is not able to do so.

IV. Experimental Setup

Different IoT applications measure different spatiotemporal phenomena, and we have, therefore, considered a general way in which to model the *prior information* regarding the spatial variability of the variates being estimated. A variogram (or a semi-variogram) [48] is an experimental function that is used to determine spatial correlations in observations measured in a defined area. They provide a means to analyze how one point has an influence on other points in different spatial and time separations (lags).

In this work, the mean and the covariance functions were used as the *prior* knowledge. In order to obtain covariance estimates, it is necessary to compute the experimental variogram [49] at sample locations in the area of interest. The idea behind variogram analysis is to build a variogram that will obtain the best estimate of the auto-correlation structure of the essential stochastic process. Computing the experimental variogram includes fitting a wide sense stationary, spatially isotropic, spatiotemporal covariance mode to the real data of sensor measurements [50]. As we performed the estimation of missing value iteratively in the time domain, only the spatial

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

covariance model was used to fit the data. Many variogram models are available, depending on the characteristics of the problem. These include *nugget effect*, linear, exponential, spherical, Gaussian and potential models [51], [52]. The most important part of a variogram is its shape near the origin, as the points closer to the origin are given more weight in the estimation process. A typical variogram model is depicted in Fig. 2. The main components of a variogram are *sill*, *range* and *nugget effect*. They are defined as:

- *nugget effect* (C_0) is the value at which the semi-variogram (almost) intercepts the y-axis.
- *sill* (C) is the point at which the model flattens out. For large values of separation distance (h), the variogram levels out, indicating that there are no more correlation between data points.
- *range* (r) is the distance between the origin and the sill, and this represents the general distance over which the samples are auto-correlated.

In our experiments, we use the exponential variogram, based on fitting the best model to the empirical variogram obtained for the datasets employed for evaluation. Its formula is:

$$\gamma(h) = C_0 + C[1 - exp(\frac{-3h}{r})], \tag{12}$$

where C_0 is a constant owing to the *nugget effect*, which raises the whole theoretical semi-variogram by C_0 units, r is the distance at which samples become independent of each other (denominated as the range of a sample) and C is the sill value of $\gamma(h)$ at which the semivariogram graph levels off.

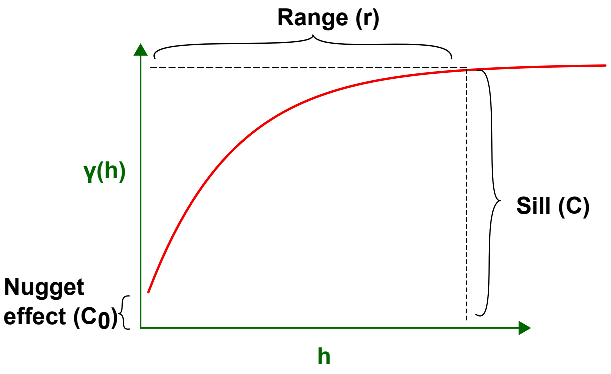

Fig. 2. Three components [nugget (c_0), sill (C) and range (r)] of a generic Variogram. Nugget represents the measurement error.

Hard and soft sensors measurements provide *specific knowledge* that can be incorporated into the meta-prior stage of BME. As we did not have the ground truth information for the low-precision sensors, we assumed the actual measured values as a ground truth for evaluation. These actual measurements were then pre-processed into interval values, using their measurement errors δ as a basis, in order to make them soft data. The estimation algorithm was evaluated by comparing the ground truth value (actually measured) at any sensor location with the estimated value at that location using other *high-* and *low-precision* measurements obtained from sensor nodes in the neighboring locations.

As the data did not contain information regarding the measurement errors, we added some uncertainty levels to the original measurements in order to convert them into low-precision measurements, *i.e.*, the *soft data* in our simulations. These levels were determined by the widths of the interval $I_i = [x - u\delta, x + u\delta]$ where x represents the measurements, i is the soft sensor ID and $u \in \{0, 1\}$ is a random number. A specific δ is selected for each of the datasets as the deviations will be based on the variability of sensor measurements in the spatio-temporal domain. The accuracy of a sensor can be expressed either as a full scale percentage or in absolute terms. We have calculated δ as 5 % of the measurements' ranges. This was a fixed solution owing to the fact that we performed several experiments. However, since interval soft data denote physical meanings with upper and lower bounds [53], we recommend changing δ according to the information available as regards the sensors' accuracy.

A. Metrics and Implementation

The accuracy of the methods was computed using the Root Mean Squared Error (RMSE) defined as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y_i})^2}, \tag{13}$$

where n is the total number of missing values, $\bar{y_i}$ is an estimated missing value, and y_i is the corresponding ground truth for samples $i = 1, \ldots, n$.

Our experiments included two scenarios:

- **Hard**. In this scenario, we considered that all the sensors were high-precision, and had reliable measurements.
- **Soft**: In this scenario, we considered that the majority of the sensors were low-precision (soft) and that the remaining sensors were high-precision (hard).

For BME, it is necessary to choose how many neighbor sensors we wish to consider in order to estimate the missing value. We chose the three nearest sensors in all scenarios, and as will be shown later (in Section V-D), this choice did not significantly affect the results. The number of iterations in PMF was set to 300 iterations in all the experiments, unless specified. The experiments were run on a computer with an Intel i5 Processor, 8GB RAM with Ubuntu 16.04 operating system and MATLAB 2018a software.

V. Results and Discussion

We evaluated our framework with three datasets from real-world IoT scenarios. To do so, we introduced several percentages of missing values randomly distributed through each sensor. The range of missing values introduced was 1-75% of the available dataset, signifying that the experiments were carried out 75 times for each dataset. We compare the methods on the basis of the mean and standard deviation of RMSEs computed over those 75 runs. Moreover, for all the datasets, we considered a hard scenario in which all the sensors were reliable, and a soft scenario, in which the majority of the sensors contained uncertainties. For all the experiments, the parameters of PMF were 300 iterations and K = 10. In the case of BME, the number of neighboring sensors parameter was 3 in both the hard and soft scenarios. Our experiments for the three datasets and the results for each of them are discussed below.

A. *Intel Berkeley Research Lab (IBRL)*

The IBRL dataset was collected from sensors deployed in the Intel Berkeley Research Laboratory from February 28 to April 5, 2004[2]. Mica2Dot sensors with weatherboards were used to collect humidity, temperature, light and voltage values (along with topological information) once every 31 seconds, as shown in Fig. 3. A total of 2.3 million sensor readings were collected from these sensors. Nodes marked in red are considered as hard sensors and those marked in black as soft sensors. The sensors were deployed in a laboratory that has different rooms, including a server room, a laboratory, a kitchen, storage rooms, and offices. We used the temperature measurements obtained from 51 sensors, with 10 minutes' sampling and 1200 observations per sensor. The soft scenario was evaluated by considering 10 hard sensors and the remaining 41 soft sensors. The variogram parameters obtained from optimal fitting were sill = 55.31, range= 10.48 and nugget = 0. The δ chosen to create the soft data was δ = 1.5.

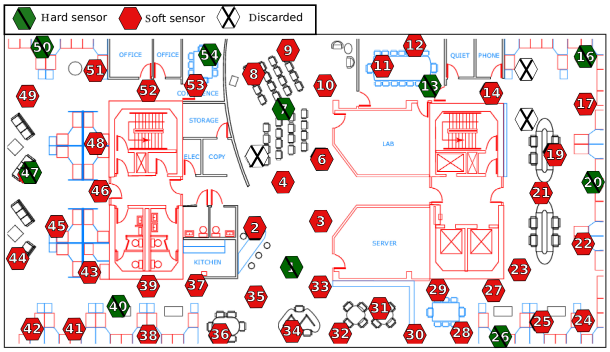

Fig. 3. Illustrates the arrangement of sensors at Intel Berkeley Research Laboratory (IBRL) between February 28th and April 5th, 2004. Mica2Dot sensors with weatherboards were used to collect humidity, temperature, light and voltage values (along with topological information) once every 31 seconds. Nodes marked in red are considered as hard sensors and those marked in black as soft sensors.

TABLE I
IBRL DATASET: RMSE MEAN AND STANDARD DEVIATION OBATINED FOR THE PMF APPROACH FOR DIFFERENT NUMBERS OF CLUSTERS. MISSING VALUES VARY FROM 1 TO 75%. BOLD TEXT INDICATES THE BEST VALUES (*i.e.*, THE LOWEST ERROR).

Number of clusters	**PMF hard** mean (std)	**PMF soft** mean (std)
1	2.45 (0.04)	2.75 (0.05)
2	2.2 (0.04)	2.66 (0.05)
3	2.25 (0.06)	2.51 (0.06)
4	1.94 (0.03)	2.26 (0.04)
5	**1.4 (0.09)**	**1.71 (0.09)**
6	1.41 (0.15)	1.83 (0.15)
7	1.54 (0.2)	1.91 (0.2)
8	2.14 (0.39)	2.37 (0.43)
9	1.62 (0.4)	2.06 (0.35)
10	1.68 (0.36)	2.12 (0.39)
11	2.12 (0.37)	2.51 (0.49)
12	1.91 (0.39)	2.37 (0.34)
13	2.14 (0.33)	2.6 (0.37)
14	2.02 (0.56)	2.29 (0.5)
15	2.21 (0.57)	2.66 (0.53)

For PMF, we apply k-means at 51 sensor locations in order to cluster them according to their coordinates as required by the PMF algorithm [34]. Table I shows the mean and standard deviation for both the hard and soft scenarios for the PMF method. Fig. 4 visually represents the PMF results for the hard-scenario using box plots. Fig. 5 shows the mean RMSE for both PMF and our BME-based framework for each missing value percentage. As can be seen in Table I and Fig. 4, the best result for PMF is obtained for 5 clusters in both hard and soft scenarios. However, its lowest RMSE (best result) for both hard ($\mu = 1.4$, $sd = 0.09$) and soft ($\mu = 1.71$, $sd = 0.09$) is still quite high when compared with the performance of the proposed BME hard ($\mu = 0.93$, $sd = 0.1$) and soft ($\mu = 0.97$, $sd = 0.1$) RMSE (see Fig. 5).

Fig. 6 illustrates the running time required for PMF (for 5 clusters, which is the best case) and BME for both the hard and soft scenarios for 1 to 75 percent of missing values. In Fig. 6, it is evident that the computation time of BME for a hard scenario is within the range 0.26 to 12.08 seconds, whereas for the PMF hard scenario, it ranges from 16.22 to 49.1 seconds. This evidence proves the superiority of BME based on the execution time for the hard scenario. Please note that the running time of PMF reported in Fig. 6 does not take into account the fact that PMF has to be run with several cluster configurations, which further increases its computational cost. In the case of the PMF approach, there are no significant differences between the running times of the hard and soft scenarios, as PMF does not incorporate measurement errors and thus considers both hard and soft data in a similar manner. However, it is possible to see the increased computational time for the BME soft scenario, in which the computational time ranges from 0.92 to 33.06 seconds. When more than 50% of the data is missing, the computation time of BME is higher than that of PMF for the soft scenario. However, this should not occur in real scenarios, since if more than half of the data missing, then the sensor no longer functions or has not been functioning for a long period of time.

The PMF method may not yield the same estimate during different iterations owing to the initial randomness present in the matrices **U** and **V** (defined in Step 3 of the PMF algorithm, Section II-C). It is therefore, necessary to compute several iterations in order to obtain robust results. We validated this by running PMF 100 times so as to estimate the missing values for each iteration. The parameters chosen for this experiment were 10% missing values, 5 clusters (best case). Fig. 7 shows the box-plot for the RMSE values and the average run-time (over 100 iterations). The number of iterations was chosen experimentally, according to the results obtained, since at 300 iterations PMF already exceeded the running time of BME

[2] http://db.csail.mit.edu/labdata/labdata.html

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

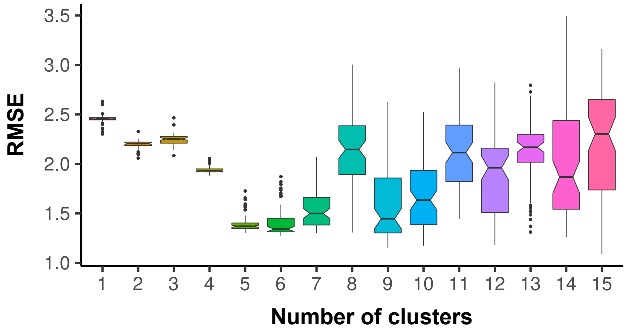

Fig. 4. IBRL dataset: RMSE values obtained for PMF approach for different number of clusters (k) in the hard scenario. Missing values vary from 1 to 75 %, and each boxplot corresponds to the mean and standard deviation of 75 RMSEs computed for each k.

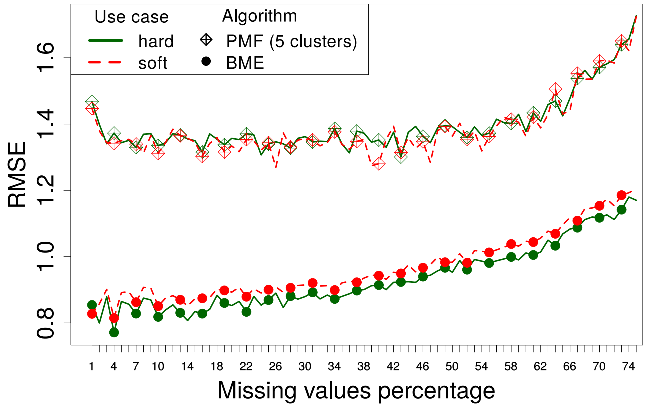

Fig. 5. RMSE mean and standard deviation values for PMF (5 clusters) and BME approaches using both hard and soft scenarios against the percentage of missing values on the IBRL dataset.

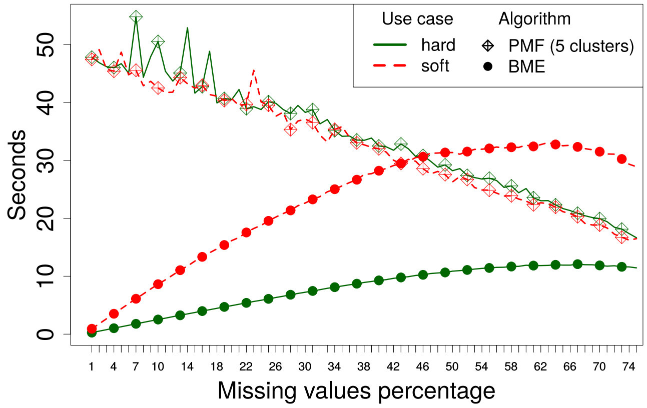

Fig. 6. Running time required for PMF and BME approaches against the percentage of missing values in the IBRL dataset. This shows that the running time required is less for the BME hard scenario. PMF takes a similar time for both the hard and soft scenario, as PMF does not distinguish between hard and soft.

and the estimation errors were stable and higher than using BME. As can be seen, the PMF model stabilizes only after 50 iterations and the lowest RMSE is achieved for 300 iterations, at the expense of a higher running time (40 seconds).

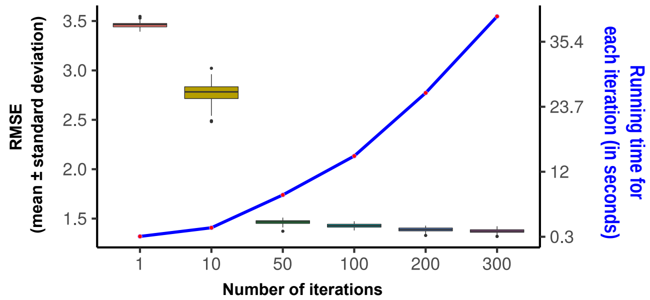

Fig. 7. Boxplot of RMSE values (for 100 iterations) obtained for PMF (hard scenario) for 300 iterations with 5 nearest neighbors and 10% of missing values on the IBRL dataset. The PMF model reaches stability only after 50 iterations and attains the lowest RMSE at 300 iterations.

B. *Sensors Deployed at Docklands Library, Melbourne, Australia*

This dataset was collected from an IoT network deployed in the City of Melbourne, Australia, for real-time urban microclimate monitoring[3] with the aim of studying the long term micro-scale relation between canopy coverage and environmental parameters [50]. The dataset included temperature, humidity and luminosity sensor measurements collected from four low-cost sensors, namely the Waspmote [54] coupled with Smart Cities sensor board, at each 10 minutes interval. The deployment locations of the sensor nodes are shown in Fig. 8. For this study, we have used the humidity measurements of four sensor nodes. These data were resampled at 30 minute intervals by averaging them in 30-minute windows, which provided us with 1200 observations. Since measurements were taken from low-cost, low-precision sensors, we consider all of them to be soft sensors in the soft scenario.

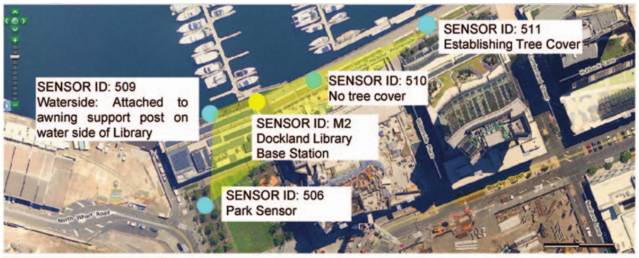

Fig. 8. Deployment of four sensor nodes at a base station at the Docklands Library, Melbourne Australia. The data was collected on 15 Dec 2014 and 26-Feb-2015 [55].

The variogram parameters obtained from optimal fitting on this dataset were sill = 102.9, range = 19.78 and nugget = 0. The δ chosen to create the soft data was δ = 1.5. Fig. 9 shows the RMSE values for the PMF and BME-based method in both scenarios for different percentages $(1 - 75)$ of missing values. As can be seen, the RMSE values for BME are significantly lower when compared to those of PMF. The mean and standard deviation of the RMSE values for both the PMF hard scenario (μ = 14.56, sd = 2.99) and the PMF soft scenario (μ =

[3] http://issnip.unimelb.edu.au/research_program/Internet_of_Things/iot_deployment

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

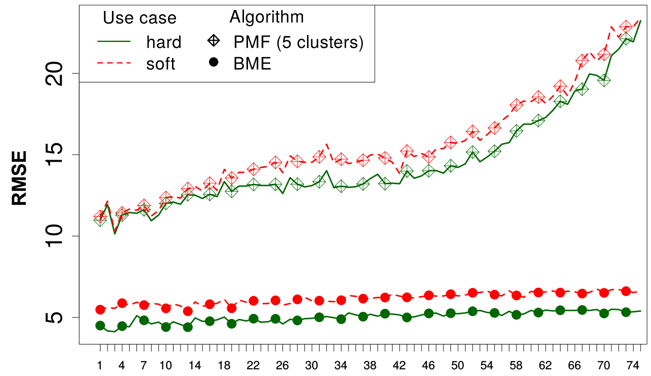

Fig. 9. Comparison of RMSE values for PMF and BME for hard and soft scenarios, tested using the Docklands dataset. Note that the RMSE values are significantly lower for BME when compared with PMF. This large difference is attributed to the high variance in the sensor measurements and the fact that BME handles the variance highly effectively in order to maintain a low RMSE.

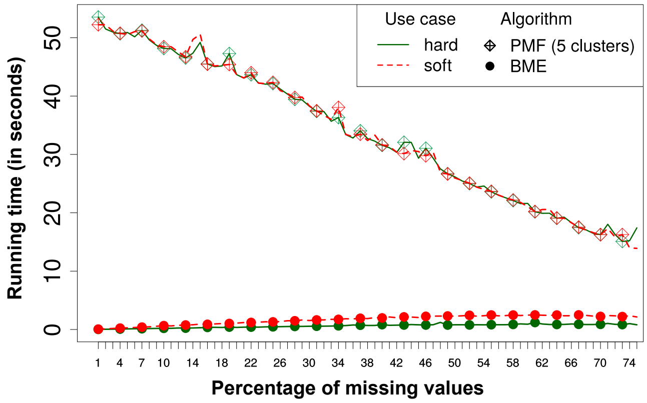

Fig. 10. Comparison of running time (in seconds) of PMF and BME for hard and soft scenarios, tested on the Docklands dataset. It is evident that BME requires significantly less computational time when compared with PMF for both hard and soft scenarios.

15.6, $sd = 3.15$) are high when compared with the BME hard ($\mu = 5.04$, $sd = 0.33$) and the BME soft scenario ($\mu = 6.17$, $sd = 0.35$). Note that the difference between the PMF and BME RMSE values is much higher for the Dockland dataset than for the IBRL dataset. This is because the IBRL data were collected in a controlled environment (office); signifying that the readings of all the sensors are pretty close to each other, which does not lead to many differences in the performances of the BME and PMF methods. However, for outdoor deployment (as is the case of deployment at Docklands Library), where we expect a higher variance between the measurements of each sensor, the BME-based algorithm performs much better than the PMF-based method. This demonstrates the effectiveness of the BME-based imputation method for IoT outdoor scenarios. Fig. 10 shows the running time for both the BME and the PMF-based methods in both scenarios. Note that the running time for BME is always lower than that of the PMF method in both the hard and soft scenarios. The differences between soft and hard running time for PMF are not significant, as there is no difference in the number of computations carried out.

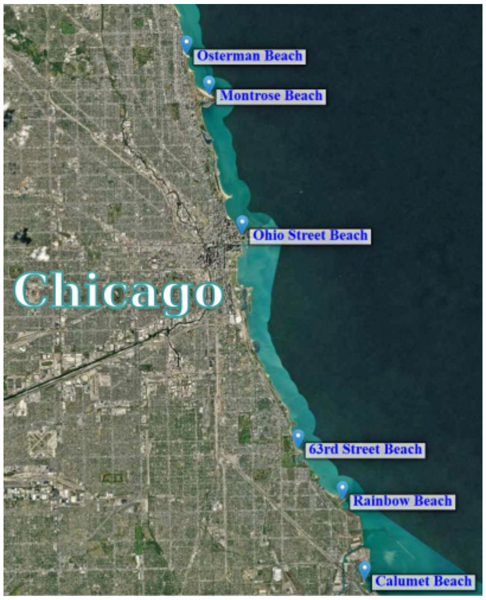

Fig. 11. Location of water temperature sensors on beaches along lakefront of Chicago Lake Michigan.

C. Beach dataset

We extracted the hourly water temperature measurements that are published in the City of Chicago's open data portal[4]. The sensor data is publicly available for use. The Chicago Park District maintains sensors in the water at beaches along the lakefront of Lake Michigan, Chicago. Fig. 11 depicts the locations of these sensors. In this work, we considered 5 soft sensors and 1 hard sensor for the soft scenario. The variogram parameters obtained from optimal fitting were sill = 0.103, range= 4.07 and nugget = 0. The δ chosen to create the soft data was $\delta = 1$. Figs. 12 and 13 show the RMSE values and running time for PMF and BME-based methods in both hard and soft scenario, respectively. The mean and standard deviation of the RMSE values for the PMF hard scenario ($\mu = 2.73$, $sd = 0.48$) and soft scenario ($\mu = 4.85$, $sd = 1$) are higher than those of the BME hard ($\mu = 2.4$, $sd = 0.2$) and soft scenario ($\mu = 2.47$, $sd = 0.23$). Although the RMSE values were initially close in the hard scenario for both PMF and BME, they started separating further as the percentage of missing values surpassed 15%. However, BME performed much better (it provided significantly less RMSE) than PMF in the soft scenario, in which the majority of the sensors were soft sensors (low-precision measurements), which is often the case in modern IoT deployments. As occurred with the IBRL and Docklands datasets, the BME was much faster (lower running time) than the PMF-based method in both scenarios (see Fig. 13).

D. Effect of BME Parameters on Computation Speeds

As mentioned in Section IV-A, BME uses the spatiotemporal property of the neighboring sensors in its estimation, signifying that BME requires the input choice of the maximum

[4] https://data.cityofchicago.org/

The final version of this article is available at http://dx.doi.org/10.1109/JIOT.2020.2987979

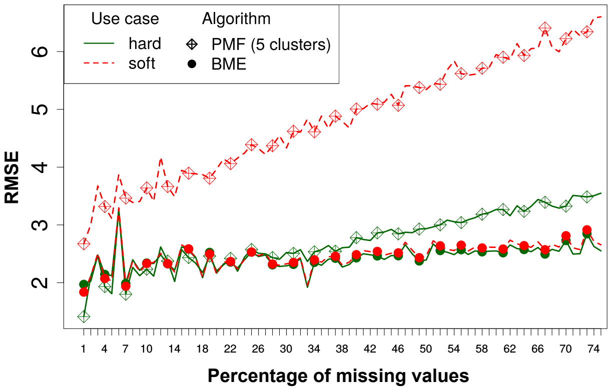

Fig. 12. Comparison of RMSE values for PMF and BME in hard and soft scenarios tested with the Beach dataset. As the percentage of missing values increases, the RMSE values for PMF increases at a much higher rate, when compared with those for BME.

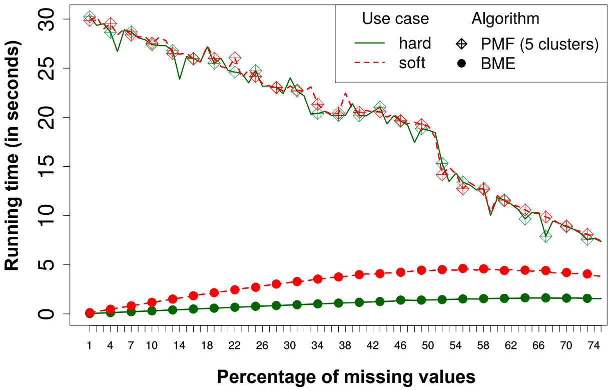

Fig. 13. Comparison of running time (in seconds) for PMF and BME in hard and soft scenarios tested with the Beach dataset. BME performs much better than PMF in both hard and soft scenarios.

number of neighboring sensors. In this experiment, we study how the choice of the maximum number of neighbors affects BME estimation when 10 % of the data are missing. Table II shows the average RMSE values and running time for all the three datasets for different (maximum) numbers of neighboring *hard* sensors, varying from 1 to 6, and the running time of BME (in seconds). Note that the RMSE decreases with the increasing number of neighboring sensors, and does not change much with more than 3 neighboring sensors. The lowest RMSE can be achieved by using higher number of neighboring sensors but that will also increase the computational-time for BME estimation. Table III shows the average RMSE values and running time for all three datasets for different maximum number of neighbouring *soft* sensors varying from 1 to 6. As occurred in the hard scenario, the RMSE values do not change much with more than 3 neighboring soft sensors. Considering the trade-off between accuracy and running time, we chose 3 as the maximum number of neighboring sensors in all our experiments in both the hard and soft scenario.

BME has only one tuning parameter (number of neighbors), which does not affect BME estimation performance; however, PMF has many parameters such as the number of clusters, the number of iterations or the RMSE threshold, the number of latent feature vectors K, step value α, and the mean and standard deviation for the random matrix R that need to be tuned in order to attain a desirable PMF performance. This shows that BME is a more robust framework for the imputation of missing data in a real-time IoT scenario.

TABLE II
COMPARISON OF RMSE VALUES AND RUNNING TIME (IN SECONDS) OF HARD BME. THE EVALUATION INCLUDED VARYING THE MAXIMUM NUMBER OF HARD NEIGHBORS WITH 10 % OF MISSING DATA.

		Maximum number of hard neighbors					
Metric	**Dataset**	**1**	**2**	**3**	**4**	**5**	**6**
RMSE	IBRL	1.11	0.87	0.81	0.78	0.77	0.77
	Docklands	5.35	4.64	4.41	4.41	-	-
	Beach	2.98	2.56	2.33	2.31	2.29	2.29
Running time (in seconds)	IBRL	2.39	2.81	2.84	2.9	2.87	2.88
	Docklands	0.19	0.23	0.2	0.21	-	-
	Beach	0.28	0.38	0.59	0.33	0.34	0.31

TABLE III
COMPARISON OF RMSE VALUES AND RUNNING TIME (IN SECONDS) OF SOFT BME. THE EVALUATION INCLUDED VARYING THE MAXIMUM NUMBER OF SOFT NEIGHBORS WITH 10 % OF MISSING DATA AND MAXIMUM NUMBER OF HARD NEIGHBORS FIXED AT 3.

		Maximum number of soft neighbors					
Metric	**Dataset**	**1**	**2**	**3**	**4**	**5**	**6**
RMSE	IBRL	0.97	0.87	0.85	0.84	0.83	0.83
	Docklands	6.68	5.85	5.54	5.54	-	-
	Beach	2.74	2.40	2.37	2.33	2.33	2.33
Runing time (in seconds)	IBRL	10.2	10.8	11.11	11.68	13.08	15.7
	Docklands	0.58	0.56	0.57	0.59	-	-
	Beach	1.02	1.16	1.11	1.19	1.24	1.24

In our experiments, we used the BME framework as a centralized approach in which all the missing values were estimated simultaneously on a machine using neighboring sensors and their measurements at corresponding instants. In our prior work [14], we implemented a distributed approach BME algorithm in which each sensor could estimate its missing value by using neighborhood sensors iteratively in the time-domain. Moreover, the distributed approach can also be implemented on the sensor board itself for online, in-network processing. We are, therefore, of the firm opinion that the BME-based framework could be very useful for online missing data imputation in IoT scenarios. In this distributed approach, once the variograms have been computed, BME requires only the instant readings of neighboring sensors in order to estimate the missing values. The distributed approach of BME can also be implemented on a sensor board for in-network processing.

Our future work will include an in-depth study of how the variogram fit would influence the estimations. We also aim to analytically study the effect of the experimental imitation of the missing values in real-time systems.

VI. CONCLUSION

Missing value imputation is a well-known issue that has been studied in the past. However, the continuously evolving IoT applications pose new challenges, which limit the performance of existing techniques, such as Probability Matrix Factorization (PMF) and Kriging. In this article, we presented a new framework with which to impute the missing values of sensors using the Bayesian Maximum Entropy (BME) technique. The BME technique handles data uncertainty very well in its estimation and also requires less computational time and a single parameter as input for the model, thus making the BME framework well suited to real IoT deployments. We have compared the performance of our BME-based framework with a state-of-the-art method (*i.e.*, the PMF approach) for missing data imputation in IoT applications and we have found that the BME approach outperforms it as regards: (1) estimating the missing values, (2) requiring significantly less computational time, (3) managing data variance robustly, and (4) managing estimation with a single model parameter. Our results are based on our research and the experiments that we conducted with three real datasets that have different IoT contexts and origins (indoor and outdoor applied to a wide range of applications). The results attained after experimenting with these datasets demonstrate the superiority of the BME framework as regards accuracy, running time, and robustness. Our framework can also be extended to distributed IoT nodes for the online imputation of missing values.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] IHS Markit, "Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions)," 2016, last accessed 18 July 2019. [Online]. Available: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/

[3] K. Kambatla, G. Kollias, V. Kumar, and A. Grama, "Trends in big data analytics," *Journal of Parallel and Distributed Computing*, vol. 74, no. 7, pp. 2561–2573, 2014.

[4] T. C. Havens, J. C. Bezdek, C. Leckie, L. O. Hall, and M. Palaniswami, "Fuzzy c-means algorithms for very large data," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 6, pp. 1130–1146, 2012.

[5] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a smart city through internet of things," *IEEE Internet of Things journal*, vol. 1, no. 2, pp. 112–121, 2014.

[6] A. González-Vidal, F. Jiménez, and A. F. Gómez-Skarmeta, "A methodology for energy multivariate time series forecasting in smart buildings based on feature selection," *Energy and Buildings*, vol. 196, pp. 71–82, 2019.

[7] A. González-Vidal, A. P. Ramallo-González, F. Terroso-Sáenz, and A. Skarmeta, "Data driven modeling for energy consumption prediction in smart buildings," in *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017, pp. 4562–4569.

[8] A. Tzounis, N. Katsoulas, T. Bartzanas, and C. Kittas, "Internet of things in agriculture, recent advances and future challenges," *Biosystems Engineering*, vol. 164, pp. 31–48, 2017.

[9] B. V. Philip, T. Alpcan, J. Jin, and M. Palaniswami, "Distributed real-time iot for autonomous vehicles," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1131–1140, 2018.

[10] A. Vafaeinejad, "Dynamic guidance of an autonomous vehicle with spatio-temporal gis," in *International Conference on Computational Science and Its Applications*. Springer, 2017, pp. 502–511.

[11] A. S. Rao, S. Marshall, J. Gubbi, M. Palaniswami, R. Sinnott, and V. Pettigrovet, "Design of low-cost autonomous water quality monitoring system," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2013, pp. 14–19.

[12] A. González-Vidal, J. Cuenca-Jara, and A. F. Skarmeta, "Iot for water management: Towards intelligent anomaly detection," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE, 2019, pp. 858–863.

[13] A. S. Rao, J. Gubbi, T. Ngo, P. Mendis, and M. Palaniswami, "Internet of things for structural health monitoring," in *Structural Health Monitoring Technologies and Next-Generation Smart Composite Structures*. CRC Press, 2016, pp. 89–120.

[14] P. Rathore, D. Kumar, S. Rajasegarar, and M. Palaniswami, "Maximum entropy-based auto drift correction using high-and low-precision sensors," *ACM Transactions on Sensor Networks (TOSN)*, vol. 13, no. 3, p. 24, 2017.

[15] P. J. García-Laencina, J.-L. Sancho-Gómez, and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review," *Neural Computing and Applications*, vol. 19, no. 2, pp. 263–282, 2010.

[16] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

[17] T. M. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[18] L. Pan and J. Li, "K-nearest neighbor based missing data estimation algorithm in wireless sensor networks," *Wireless Sensor Network*, vol. 2, no. 02, p. 115, 2010.

[19] D. L. Donoho *et al.*, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[20] L. Kong, M. Xia, X.-Y. Liu, M.-Y. Wu, and X. Liu, "Data loss and reconstruction in sensor networks," in *2013 Proceedings IEEE INFOCOM*. IEEE, 2013, pp. 1654–1662.

[21] P. Vamplew and A. Adams, "Missing values in a backpropagation neural net," in *Proceedings of the 3rd. Australian Conference on Neural Networks (ACNN), I*, 1992, pp. 64–66.

[22] J. Yoon, J. Jordon, and M. Van Der Schaar, "Gain: Missing data imputation using generative adversarial nets," *arXiv preprint arXiv:1806.02920*, 2018.

[23] L. Z. Wong, H. Chen, S. Lin, and D. C. Chen, "Imputing missing values in sensor networks using sparse data representations," in *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*. ACM, 2014, pp. 227–230.

[24] A. Ramallo-González, "New method to reconstruct building environmental data," in *Buildign Simulation International Conference BS2015, University of Bath*, 2015.

[25] C.-Y. Li, W.-L. Su, T. G. McKenzie, F.-C. Hsu, S.-D. Lin, J. Y.-j. Hsu, and P. B. Gibbons, "Recommending missing sensor values," in *2015 IEEE International Conference on Big Data (Big Data)*. IEEE, 2015, pp. 381–390.

[26] J. A. Manrique, J. S. Rueda-Rueda, and J. M. Portocarrero, "Contrasting internet of things and wireless sensor network from a conceptual overview," in *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2016, pp. 252–257.

[27] X. Yan, W. Xiong, L. Hu, F. Wang, and K. Zhao, "Missing value imputation based on gaussian mixture model for the internet of things," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[28] I. P. S. Mary and L. Arockiam, "Imputing the missing data in iot based on the spatial and temporal correlation," in *Current Trends in Advanced Computing (ICCTAC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1–4.

[29] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: an open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. ACM, 1994, pp. 175–186.

[30] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 43–52.

[31] D. Billsus and M. J. Pazzani, "Learning collaborative information filters." in *Icml*, vol. 98, 1998, pp. 46–54.

[32] H. Ma, I. King, and M. R. Lyu, "Effective missing data prediction for collaborative filtering," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2007, pp. 39–46.

[33] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Advances in neural information processing systems*, 2008, pp. 1257–1264.

[34] B. Fekade, T. Maksymyuk, M. Kyryk, and M. Jo, "Probabilistic recovery of incomplete sensed data in iot," *IEEE Internet of Things Journal*, 2017.

[35] R. A. Olea, *Geostatistics for engineers and earth scientists*. Springer Science & Business Media, 2012.

[36] H. Yang, J. Yang, L. D. Han, X. Liu, L. Pu, S.-m. Chin, and H.-l. Hwang, "A kriging based spatiotemporal approach for traffic volume data imputation," *PloS one*, vol. 13, no. 4, p. e0195957, 2018.

[37] M. Ardakani, A. Shokry, G. Saki, G. Escudero, M. Graells, and A. Espuña, "Imputation of missing data with ordinary kriging for enhancing fault detection and diagnosis," in *Computer Aided Chemical Engineering*. Elsevier, 2016, vol. 38, pp. 1377–1382.

[38] G. Christakos and X. Li, "Bayesian maximum entropy analysis and mapping: a farewell to kriging estimators?" *Mathematical Geology*, vol. 30, no. 4, pp. 435–462, 1998.

[39] G. Christakos, *Modern spatiotemporal geostatistics*. Oxford University Press, 2000, vol. 6.

[40] J. He and A. Kolovos, “Bayesian maximum entropy approach and its applications: a review,” *Stochastic Environmental Research and Risk Assessment*, vol. 32, no. 4, pp. 859–877, 2018.

[41] A. d. Nazelle, S. Arunachalam, and M. L. Serre, “Bayesian maximum entropy integration of ozone observations and model predictions: an application for attainment demonstration in north carolina,” *Environmental science & technology*, vol. 44, no. 15, pp. 5707–5713, 2010.

[42] K. Modis, K. Vatalis, and C. Sachanidis, “Spatiotemporal risk assessment of soil pollution in a lignite mining region using a bayesian maximum entropy (bme) approach,” *International Journal of Coal Geology*, vol. 112, pp. 173–179, 2013.

[43] A. Zagouras, A. Kolovos, and C. F. Coimbra, “Objective framework for optimal distribution of solar irradiance monitoring networks,” *Renewable Energy*, vol. 80, pp. 153–165, 2015.

[44] Y. Akita, J.-C. Chen, and M. L. Serre, “The moving-window bayesian maximum entropy framework: estimation of pm 2.5 yearly average concentration across the contiguous united states,” *Journal of Exposure Science and Environmental Epidemiology*, vol. 22, no. 5, p. 496, 2012.

[45] M. L. Serre and G. Christakos, “Modern geostatistics: computational bme analysis in the light of uncertain physical knowledge–the equus beds study,” *Stochastic Environmental Research and Risk Assessment*, vol. 13, no. 1-2, pp. 1–26, 1999.

[46] S. Waldrip and R. Niven, “Comparison between bayesian and maximum entropy analyses of flow networks,” *Entropy*, vol. 19, no. 2, p. 58, 2017.

[47] C. E. Shannon, “A mathematical theory of communication,” *Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[48] N. Cressie, “Statistics for spatial data: Wiley series in probability and statistics wiley-interscience,” *NY*, vol. 15, pp. 105–209, 1993.

[49] I. Clark, *Practical geostatistics*. Applied Science Publishers London, 1979, vol. 3.

[50] P. Rathore, A. S. Rao, S. Rajasegarar, E. Vanz, J. Gubbi, and M. Palaniswami, “Real-time urban microclimate analysis using internet of things,” *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 500–511, 2018.

[51] N. Cressie, “Fitting variogram models by weighted least squares,” *Journal of the International Association for Mathematical Geology*, vol. 17, no. 5, pp. 563–586, 1985.

[52] B. Shamo, E. Asa, and J. Membah, “Linear spatial interpolation and analysis of annual average daily traffic data,” *Journal of Computing in Civil Engineering*, vol. 29, no. 1, p. 04014022, 2012.

[53] J. Hu, J. Zhou, G. Zhou, Y. Luo, X. Xu, P. Li, and J. Liang, “Improving estimations of spatial distribution of soil respiration using the bayesian maximum entropy algorithm and soil temperature as auxiliary data,” *PloS one*, vol. 11, no. 1, p. e0146589, 2016.

[54] Libelium, “Waspmote,” http://www.libelium.com/products/waspmote/, 2016.

[55] A. Shilton, S. Rajasegarar, C. Leckie, and M. Palaniswami, “Dp1svm: A dynamic planar one-class support vector machine for internet of things environment,” in *2015 International Conference on Recent Advances in Internet of Things (RIoT)*. IEEE, 2015, pp. 1–6.