Data Availability Optimization for Cyber-Physical Systems

Liying Li[†], Peijin Cong[†], Junlong Zhou[†‡], Zonghua Gu[¶], Keqin Li[§]
[†]School of Computer Science and Engineering, Nanjing University of Science and Technology, China
[‡]State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, China
[¶]Department of Applied Physics and Electronics, Umeå University, Sweden
[§]Department of Computer Science, State University of New York, USA

***Abstract*—As the backbone of Industry 4.0, Cyber-Physical Systems (CPSs) have attracted extensive attention from industry, academia, and government. Missing data is a common problem in CPS data processing and may cause incorrect results and eventually serious malfunction. Existing data availability optimization methods either rely on a large amount of complete training data or suffer from poor performance. To solve these problems, this paper proposes an iterative data availability optimization method for CPSs. Specifically, the proposed method first pre-processes the raw dataset by using a Singular Value Decomposition-based feature selection approach to identify crucial features and reduce computation overheads. It then makes an initial guess for missing values via a designed K-Means-based imputation approach. The appropriate initial estimation decreases the probability of the proposed method falling into the local optimum. Finally, the proposed method iteratively estimates missing data based on the Orthogonal Matching Pursuit algorithm. The proposed method optimizes data availability by accurately imputing missing values. Simulation results on two datasets demonstrate that compared to multiple state-of-the-art approaches, the proposed data availability optimization method can reduce imputation error by up to 99.65%.**

***Index Terms*—Cyber-Physical System, Data Availability Optimization, Missing Data Imputation.**

I. Introduction

A. Motivation

As a new generation of digital system of Industry 4.0, Cyber-Physical System (CPS) has attracted more and more attention due to its significant impacts on society, environment, and economy [1], [2]. Typically, a CPS tightly integrates the cyber world of computing and communication with the physical world of sensing and actuation to improve the availability and functionality of various applications [3]–[5]. In CPS, cyber components like cloud servers compute results and make decisions according to data collected or generated by physical components such as sensors. If the availability/quality of these data is insufficient, more than 41% of relevant CPS applications will generate unexpected intermediate results and eventually result in severe failures ("garbage in, garbage out") [6]. Therefore, data availability/quality is crucial for process control and decision-making activities in CPSs.

In general, four measurements are considered important factors impacting data availability: completeness, accuracy, validity, and timeliness [7], [8]. Among various potential problems in data processing, missing data is one of the most important factors that have a significant impact on completeness, accuracy, and validity simultaneously [9]. In reality, raw data in CPSs are collected and/or generated from various CPS sensors which are often tiny and fragile and are usually deployed in complicated and even harsh environments [10]. Hence, missing data are prevalent in CPS due to sensor malfunction. Moreover, some sensors are powered by unstable and uncertain renewable energy. Power-off and other energy supply-related factors may also lead to data missing [11]. Further, when transmitting data to the cyber world, due to the volatility and instability of wireless transmission protocols, data will also be missing during transmission [12].

Three potential problems may be induced by missing data: waste of data resources, meaningless computation results, and misleading computation results [9], [13], [14]. Specifically, i) most applications handle missing data by automatically discarding samples containing missing values, which may lead to insufficient data volumes for analysis. In addition, deleting the whole sample is essentially a waste of valuable data resources, and applications may eventually malfunction because of the shortage of important information. ii) Meaningless computation results may be derived due to insufficient data especially when the missing rate is high. iii) When data missing occurs in important features, misleading computation results are possibly generated and may disturb the process control and decision-making in CPSs, affecting the quality of service of CPSs.

B. Related Work

Existing methodologies for handling missing data can be in general divided into ignoring, discard, and imputation [15]. Ignoring methods simply neglect missing values and directly process raw datasets containing missing data for computation. Discard methods delete the whole sample with one or more missing values and continue the data analysis. Imputation methods replace missing values with estimated values to improve data availability. Among the three options, ignoring and discard methods essentially optimize data processing efficiency by reducing the amount of data. For instance, Wang *et al.* [16] designed a novel tensor-based Long Short-Term

Junlong Zhou is the corresponding author. E-mail: jlzhou@njust.edu.cn.
This work was supported in part by the National Natural Science Foundation of China under Grants 62172224 and 61802185, in part by the China Postdoctoral Science Foundation under Grants BX2021128, 2021T140327, 2020M680068, and in part by the Open Research Fund of the State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences under Grant CARCHA202105, in part by the Future Network Scientific Research Fund Project under Grant FNSRFP-2021-YB-6, and in part by the Open Research Fund of Engineering Research Center of Software/Hardware Co-Design Technology and Application, Ministry of Education (East China Normal University) under Grant OP202203.

Memory with edge plane and cloud plane to process big data with high-efficiency via reducing parameters and distributed computing strategy. Ignoring and discard methods are simple but may lead to insignificant even wrong computation results if the data missing rate is high, and waste data resources [17]. In practice, high data missing rate would cause deviation in data analysis results. Therefore, it is more practical to optimize data availability using imputation methods.

Recently, numerous imputation methods are developed to optimize data availability. Shrestha *et al.* [18] proposed an imputation method to enhance the data availability for sepsis prediction. This method first combines information gain and the Chi-square to select appropriate features in raw datasets. It then estimates missing values in the raw dataset by using the mean value of the non-missing parts. Liao *et al.* [19] introduced a slide window technology into the data availability optimization procedure. They designed a fuzzy K-Means imputation algorithm over a sliding window to estimate missing values. In [20], the authors combined Bayesian kernelized matrix factorization with Markov chain Monte Carlo sampling to estimate missing values. The aforementioned methods are simple and easy to implement. However, they suffer from poor performance especially when the data missing rate is high.

To improve the imputation performance, more and more research efforts are devoted to estimating missing values via learning-based methods. Gong *et al.* [21] designed a learning model consisting of multiple spatial kernels for missing data imputation. This learning-based method considers regional similarities, positions, and correlations among multiple views to impute missing values. Khan *et al.* [22] combined three machine learning algorithms (i.e., extreme gradient boosting, categorical boosting, and random forest) and proposed a data imputation approach for energy data estimation. In [23], the authors utilized the long short-term memory network to recover missing data in time series sensor datasets in CPSs. The above-mentioned learning-based imputation methods can overcome the shortcoming of poor imputation performance. However, they usually need a large amount of complete data to train an accurate learning algorithm. Besides, different application scenarios and/or datasets usually need different learning models. This would bring a dramatic computation overhead inevitably.

C. Contribution

In this paper, we propose an effective data availability optimization method for CPSs. Our key idea is to replace missing values in raw datasets by using accurate estimated values via incomplete datasets without high computation overheads. The major contributions of this paper are summarized as follows.

- We first propose a feature selection method to preprocess the raw dataset based on the Singular Value Decomposition (SVD) algorithm. Based on the selected features, we make an initial guess for missing data via K-Means algorithm to improve the data availability.
- We further design an iterative data availability optimization method based on Orthogonal Matching Pursuit (OMP) algorithm. The proposed method iteratively adjusts the accuracy of the initial guess to improve imputation performance.
- We carry out extensive simulation experiments on two datasets. Results show that the imputation error can be decreased by up to 99.65% by using our proposed data availability optimization method, compared to several state-of-the-art (SOTA) imputation approaches.

The rest of the paper is organized as follows. Section II gives the definition of the problem to be tackled and an overview of the proposed approach. Section III introduces the details of the proposed iterative data availability optimization method. The effectiveness of the proposed approach is shown by extensive experiments in Section IV. Section V concludes the paper.

II. Problem Definition and Methodology Overview

This section defines the studied problem and gives an overview of our data availability optimization approach.

A. Problem Definition

In the scenario of our interests, multiple CPS sensors collect and generate data simultaneously. Ideally, these sensor data are complete and accurate. However, due to reasons such as sensor malfunction and transmission error, the generated raw dataset always contains missing values. We assume that there are N CPS sensors and collect M samples. Let an $M \times N$ matrix $\mathbf{X}_R$ be the raw data collected by the N sensors. $\mathbf{X}_R$ can be represented as

$$\mathbf{X}_R = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1j} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2j} & \cdots & x_{2N} \\ \vdots & \vdots & & \vdots & & \vdots \\ x_{i1} & x_{i2} & \cdots & x_{ij} & \cdots & x_{iN} \\ \vdots & \vdots & & \vdots & & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{Mj} & \cdots & x_{MN} \end{bmatrix}. \quad (1)$$

The ith ($1 \le i \le M$) sample in $\mathbf{X}_R$ is denoted by $\mathbf{x}_i$ where x_{ij} is the value in ith row jth ($1 \le j \le N$) column of $\mathbf{X}_R$.

Our goal is to accurately estimate missing values according to the observed parts of $\mathbf{X}_R$. We assume the data of one sensor can be approximately derived by using data generated from the rest sensors. Let $\mathbf{W}$ denote an $N \times N$ coefficient parameter matrix and w_{ij} ($1 \le i, j \le N$) denote the ith row jth column of $\mathbf{W}$, x_{ij} can be represented as

$$x_{ij} = \sum\nolimits_{n=1}^{N} x_{in} \cdot w_{nj}. \quad (2)$$

Let $\mathbf{w}_n$ ($1 \le n \le N$) be the nth column of $\mathbf{W}$, then Eq. (2) is rewritten as

$$x_{ij} = \mathbf{x}_i \cdot \mathbf{w}_j. \quad (3)$$

Let $\hat{\mathbf{X}}$, $\hat{\mathbf{x}}_\mathbf{i}$, and $\hat{x}_{ij}$ ($1 \le i \le M, 1 \le j \le N$) represent the estimation for $\mathbf{X}_R$, $\mathbf{x}_i$, and x_{ij}, respectively. We use Mean Squared Error (MSE) to measure the performance of the imputation. In application of our interests, the MSE is defined as

$$\text{MSE} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} \left(x_{ij} - \hat{x}_{ij}\right)^2}{M \cdot N}. \quad (4)$$

To reduce the imputation impact on the non-missing parts in $\mathbf{X}_R$, we define a mask matrix $\mathbf{Z}$ of M rows and N columns. Assume the ith row jth column value in $\mathbf{Z}$ is denoted by z_{ij} $(1 \leq i \leq M, 1 \leq j \leq N)$:

$$z_{ij} = \begin{cases} 1, & x_{ij} \text{ is not missing} \\ 0, & x_{ij} \text{ is missing} \end{cases}. \tag{5}$$

The problem to be tackled can be formulated as

$$\begin{aligned} \text{Minimize:} \quad & \text{MSE}, \\ \text{Subject to:} \quad & Diag(\mathbf{W}) = 0, \\ & z_{ij} \cdot (x_{ij} - \hat{x}_{ij}) = 0, \end{aligned} \tag{6}$$

where $Diag(\bullet)$ represents the diagonal elements of matrix $\bullet$. The first constraint of Eq. (6) guarantees that each diagonal element in the coefficient parameter matrix $\mathbf{W}$ is 0. This is because it is meaningless to estimate a value by itself. The second constraint ensures that the imputation only changes the values of missing parts. That is, if the value is non-missing (i.e., $z_{ij} = 1$), $x_{ij} = \hat{x}_{ij}$ holds.

B. Overview

We consider the general architecture of CPS that consists of a cyber and a physical world, and uses control, communication, and computation to achieve various functionalities. In the physical world, devices such as sensors collect and generate raw CPS data. These data are sent to the cyber world for further processing, and usually contain missing values. The cyber world usually contains components like cloud servers. These cyber components compute results and make decisions according to the data from the physical world. Cyber components return information such as computation results and control commands to physical components.

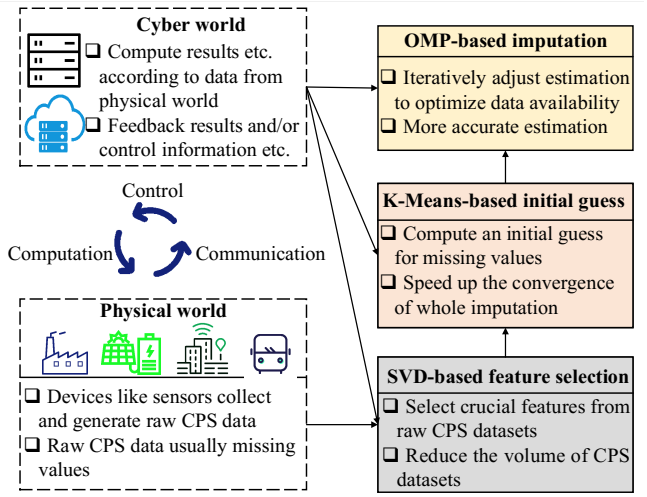

Figure 1: Methodology overview.

Fig. 1 shows an overview of the proposed CPS data availability optimization method which is implemented at three steps: feature selection, initial guess, and iterative imputation. In the feature selection step, we design a SVD algorithm based approach that selects crucial features from raw CPS datasets from the physical world to reduce the volume of the datasets and computation overheads. In the initial guess step, we develop a K-Means algorithm based imputation method that makes an initial estimation for missing values in selected datasets from the first step. An appropriate initial guess will speed up the convergence of the whole iterative imputation process and reduce the probability of the proposed method falling into the local optimum. In the last step, we propose an OMP-based imputation approach that iteratively adjusts estimation to further optimize data availability.

III. The Proposed Methodology

In this section, we introduce the proposed data availability optimization methodology in detail.

A. SVD-based Feature Selection

To reduce the volume of the CPS raw dataset and computation overheads, we propose a feature selection method based on SVD algorithm. Specifically, an $M \times N$ CPS raw dataset $\mathbf{X}_R$ is decomposed by

$$\mathbf{X}_R = U \Sigma V^T, \tag{7}$$

where U is an $M \times M$ unitary matrix, Σ is an $M \times N$ matrix whose diagonal elements are all positive, and V^T is the transposition of an $N \times N$ unitary matrix V. Note that the conjugate transpose of a unitary matrix is in fact the inverse of the matrix. That is, in Eq. (7), $U^T U = E$ and $V^T V = E$ holds where E is a identity matrix.

$\mathbf{X}_R$ can be calculated by $\mathbf{X}_R = \mathbf{X}_S + \mathbf{X}_N$ where $\mathbf{X}_S$ and $\mathbf{X}_N$ represent the set of selected and unselected features, respectively. We can reform $\mathbf{X}_R$ as

$$\begin{aligned} \mathbf{X}_R &= [U_S, U_N] \begin{bmatrix} \Sigma_S & 0 \\ 0 & \Sigma_N \end{bmatrix} \begin{bmatrix} V_S \\ V_N \end{bmatrix}, \\ &= U_S \Sigma_S V_S^T + U_N \Sigma_N V_N^T. \end{aligned} \tag{8}$$

$\mathbf{X}_S$ and $\mathbf{X}_N$ contain S selected and $M-S$ unselected features in $\mathbf{X}_R$, respectively. Σ_S and Σ_N are the matrix containing S and $M - S$ selected and rest singular values of Σ derived in Eq. (7), respectively. The diagonal elements in Σ are actually the singular values of matrix $\mathbf{X}_R$, which measures the importance of each feature. Note that the selected values are actually the largest S singular values in Σ. U_S, U_N, V_S, and V_N can be similarly defined.

After decomposing the raw dataset, we select crucial features according to corresponding singular values:

$$\mathbf{X}_S = U_S \Sigma_S V_S^T. \tag{9}$$

Algorithm 1: SVD-based Feature Selection

Input: The raw CPS dataset $\mathbf{X}_R$, the number of selected features S.
Output: The selected dataset $\mathbf{X}_S$.
1 Decompose the raw dataset $\mathbf{X}_R$ by using Eq. (7);
2 Select the largest S singular values in Σ to form Σ_S;
3 Obtain the corresponding U_S and V_S;
4 Reform $\mathbf{X}_R$ by using Eq. (8);
5 Calculate the selected dataset $\mathbf{X}_S$ by using Eq. (9);
6 **return** $\mathbf{X}_S$;

The pseudo-code of the proposed feature selection method is given in Alg. 1. It takes the raw CPS dataset $\mathbf{X}_R$ and the number of selected features S as inputs, and outputs the selected dataset $\mathbf{X}_S$. The algorithm first decomposes the raw CPS dataset $\mathbf{X}_R$ and obtains matrix U, Σ, and V by using Eq. (7) (line 1). It then selects the largest S singular values in Σ from Σ_S (line 2). The algorithm obtains the corresponding U_S and V_S according to Σ_S and reforms $\mathbf{X}_R$ by using Eq. (8) (lines 3-4). It finally calculates the selected dataset $\mathbf{X}_S$ by using Eq. (9) (line 5).

B. K-Means-based Initial Guess

To enhance the utility of the proposed data availability optimization approach, it is important to make a proper initial guess for each missing value. Otherwise, the convergence speed of the proposed iterative imputation method may be degraded. Besides, inappropriate initial may stick the proposed approach into a locally optimal solution.

We design an initial imputation based on K-Means algorithm. Specifically, we pre-process the selected dataset $\mathbf{X}_S$ by using the normalization below:

$$x_{ij}^* = \frac{z_{ij}}{\sigma_j}\left(x_{ij} - \mu_j\right), \tag{10}$$

where μ_j and σ_j ($i \leq j \leq S$) are the mean and standard deviation value of the jth column in $\mathbf{X}_S$, respectively. z_{ij} denotes the mask value defined in Eq. (5). The normalized ith ($1 \leq i \leq M$) sample and selected dataset are represented by $\mathbf{x}_i^*$ and $\mathbf{X}_S^*$, respectively.

Assume there are M_{COM} samples in $\mathbf{X}_S^*$. The M_{COM} samples are complete while other $M - M_{COM}$ samples contain missing values. These M_{COM} complete samples form dataset $\mathbf{X}_{COM}^*$ while other incomplete samples form dataset $\mathbf{X}_{IN}^*$. We first randomly select K samples from $\mathbf{X}_{COM}^*$ as clustering centers, denoted by $\mathbf{c}_1^*, \mathbf{c}_2^*, \cdots, \mathbf{c}_K^*$ where $\mathbf{c}_k^*$ ($1 \leq k \leq K$) is the center of cluster C_k. For the rest samples in $\mathbf{X}_{COM}^*$, we try to find which cluster they belong to. Specifically, for a rest sample $\mathbf{x}_i^* \in \mathbf{X}_{COM}^*$, we calculate the Euclidean distance between $\mathbf{x}_i^*$ and each clustering center C_k by using

$$d_{ik} = \sqrt{\sum\nolimits_{s=1}^{S}(x_{is}^* - x_{ks}^*)^2}, \tag{11}$$

where S is the number of selected features. According to the distance calculated above, $\mathbf{x}_i^*$ is clustered into the class with the smallest distance from the clustering center.

After all samples in $\mathbf{X}_{COM}^*$ are clustered, we update the clustering center of each cluster by using

$$\mathbf{c}_k^* = \frac{1}{|C_k|}\sum_{\mathbf{x}_i^* \in C_k}\sum_{s=1}^{S} x_{is}^*. \tag{12}$$

The process above repeats until the predefined converge criterion of solutions is met.

We obtain an initial guess for each missing value according to the clustered samples. Specifically, we calculate the distance of each sample in dataset $\mathbf{X}_{IN}^*$ and each clustering center by using Eq. (11). Then we use the corresponding values in the clustering center of the nearest cluster to estimate missing values. The normalized estimation for missing value x_{ij}^* is denoted by $\hat{x}_{ij}^*$. Finally, the initial estimation is obtained by using affine transformation to the normalized estimation, i.e.,

$$\hat{x}_{ij} = x_{ij}^* \cdot \sigma_j + \mu_j. \tag{13}$$

Note that the non-missing values whose corresponding mask value is 1, will not be changed.

The pseudo-code of the K-Means-based initial guess is summarized in Alg. 2. The algorithm takes the selected dataset $\mathbf{X}_S$, the number of cluster K as inputs, and outputs the initial estimated dataset, denoted by $\hat{\mathbf{X}}_{ini}$. It first normalizes the selected dataset $\mathbf{X}_S$ by using Eq. (10) (line 1). It then divides the normalized dataset $\mathbf{X}_S^*$ into $\mathbf{X}_{COM}^*$ and $\mathbf{X}_{IN}^*$ which consist of complete and incomplete samples, respectively (line 2). The algorithm randomly selects K samples from $\mathbf{X}_{COM}^*$ as clustering center for K clusters, and initializes the iteration number $iter$ as 1 (lines 3-4). In each iteration, for each sample $\mathbf{x}_i^* \in \mathbf{X}_{COM}^*$ except the selected clustering center, the algorithm first calculates the Euclidean distance between $\mathbf{x}_i^*$ and each clustering center C_k by Eq. (11), and then clusters $\mathbf{x}_i^*$ into the class with the smallest distance from the clustering center (lines 7-10). After all samples in $\mathbf{X}_{COM}^*$ are clustered, the algorithm updates K clustering centers by using Eq. (12) and increment $iter$ (lines 12-15). For each incomplete sample in $\mathbf{X}_{IN}^*$, the algorithm first calculates the distance between $\mathbf{x}_i^*$ and each clustering center by using Eq. (11) (line 18). Then the algorithm uses the corresponding values in the clustering center of the nearest cluster to estimate missing values (line 19). Finally, affine transformation given in Eq. (13) is used to obtain the initial guess (line 20).

Algorithm 2: K-Means-based Initial Guess

Input: The selected dataset $\mathbf{X}_S$, the number of cluster K.
Output: The initial estimated dataset $\hat{\mathbf{X}}_{ini}$.

1 Normalize the selected dataset $\mathbf{X}_S$ by using Eq. (10);
2 Divide the normalized dataset $\mathbf{X}_S^*$ into $\mathbf{X}_{COM}^*$ and $\mathbf{X}_{IN}^*$ according to whether samples containing missing values;
3 Randomly select K samples from $\mathbf{X}_{COM}^*$ as clustering center for K clusters;
4 $iter = 1$;
5 **while** $iter \leq ITER$ **do**
6 **for** *each rest sample* $\mathbf{x}_i^* \in \mathbf{X}_{COM}^*$ **do**
7 **for** *each clustering center* $\mathbf{c}_k^*$ **do**
8 Calculate the Euclidean distance between $\mathbf{x}_i^*$ and each clustering center C_k by Eq. (11);
9 **end**
10 Cluster $\mathbf{x}_i^*$ into the class with the smallest distance from the clustering center;
11 **end**
12 **for** *each cluster* C_k **do**
13 Update the clustering center by using Eq. (12);
14 **end**
15 $iter$++;
16 **end**
17 **for** *each sample* $\mathbf{x}_i^* \in \mathbf{X}_{IN}^*$ **do**
18 Calculate the distance between $\mathbf{x}_i^*$ and each clustering center by using Eq. (11);
19 Use the corresponding values in the clustering center of the nearest cluster to estimate missing values;
20 Use affine transformation in Eq. (13) to obtain the initial guess;
21 **end**
22 **return** $\hat{\mathbf{X}}_{ini}$;

The proposed K-Means-based initial guess is simple yet

effective. However, the performance of the initial imputation highly relies on the selection of K and initial clustering centers. Different K values and initial clustering centers always lead to different imputation results. Therefore, Alg. 2 is only used to obtain the initial guess. We also propose an iterative data availability optimization method in Section III-C to further improve the accuracy of the initial guess.

C. OMP-based Data Availability Optimization

After making an initial guess for each missing value, we obtain an estimated dataset ($\hat{\mathbf{X}}_{ini}$). However, the accuracy of the initial guess can be further improved. In addition, the problem to be tackled in Eq. (6) is NP-hard [24], which is extremely difficult to solve. Below, we introduce a heuristic data availability optimization method based on OMP to tackle this problem.

OMP is a classic greedy algorithm that can be used to iteratively derive the coefficient parameter matrix $\mathbf{W}$. To intuitively understand the OMP algorithm, we provide an example as follows. Define a residual matrix, denoted by $\mathbf{Res}$. OMP algorithm first initializes the residual matrix $\mathbf{Res}$ to the selected dataset $\mathbf{X}_S$. Then it calculates the inner product values, denoted by ξ_i ($1 \leq i \leq S$), between $\mathbf{Res}$ and $\hat{\mathbf{X}}$ to measure the importance of each column in $\hat{\mathbf{X}}$:

$$\xi_i = \hat{\mathbf{c}}_i^T \mathbf{Res}, \tag{14}$$

where $\hat{\mathbf{c}}_i^T$ is the transposition of the ith column in $\hat{\mathbf{X}}$. Note that in the first iteration, the given estimated dataset $\hat{\mathbf{X}}$ is actually the initial guess from Alg. 2 (i.e., $\hat{\mathbf{X}}_{ini}$). The inner product values derived by Eq. (14) are only used to identify important columns in $\mathbf{X}_S$. Let $\hat{\mathbf{c}}_m$ ($1 \leq m \leq LOOP$) be the column with the largest inner product in the mth iteration, where $LOOP$ denotes the number of iterations. Once $\hat{\mathbf{c}}_m$ is determined in each iteration, it is added to a set named Ω. Then we can estimate $\hat{\mathbf{X}}$ in the direction of the columns already selected:

$$\hat{\mathbf{X}} \approx \sum_{\hat{\mathbf{c}}_m \in \Omega} \hat{\mathbf{c}}_m \mathbf{W}_m. \tag{15}$$

$\mathbf{W}_m$ is the coefficient parameter matrix in the mth iteration, which can be consequently obtained by solving

$$\underset{\mathbf{W}_m}{\text{Minimize:}} \quad (\hat{\mathbf{X}} - \sum_{\hat{\mathbf{c}}_m \in \Omega} \hat{\mathbf{c}}_m \mathbf{W}_m)^2. \tag{16}$$

Then we update $\mathbf{Res}$ for the next iteration by

$$\mathbf{Res} = \hat{\mathbf{X}} - \sum_{\hat{\mathbf{c}}_m \in \Omega} \hat{\mathbf{c}}_m \mathbf{W}_m. \tag{17}$$

After coefficient parameter matrix $\mathbf{W}$ is obtained by using the above process, we use the least-squares fitting to derive the estimated dataset $\hat{\mathbf{X}}$ which contains complete and incomplete samples. Assume $\mathbf{X}_{COM}$ and $\hat{\mathbf{X}}_{IN}$ represent subsets of $\hat{\mathbf{X}}$ containing all complete and incomplete samples, respectively. Since $\mathbf{W}$ is known and each sample is independent, our problem can be rewritten as

$$\underset{\hat{\mathbf{c}}_i}{\text{Minimize:}} \quad (\hat{\mathbf{X}} - \sum_{\hat{\mathbf{c}}_i \in \hat{\mathbf{X}}_{IN}} \hat{\mathbf{c}}_i \mathbf{w}_i - \sum_{\mathbf{c}_j \in \mathbf{X}_{COM}} \mathbf{c}_j \mathbf{w}_j)^2, \tag{18}$$

where $\mathbf{w}_i$ and $\mathbf{w}_j$ are the column in $\mathbf{W}$ corresponding to $\hat{\mathbf{c}}_i$ and $\mathbf{c}_j$, respectively. Note that $\sum_{\mathbf{c}_j \in \mathbf{X}_{COM}} \mathbf{c}_j \mathbf{w}_j$ is a constant, thus this problem can be solved by using least-squares fitting. The above process repeats until the number of iterations reaches a predefined value $LOOP$.

Algorithm 3: OMP-based Iterative Data Availability Optimization

Input: The selected dataset $\mathbf{X}_S$, the initial estimated dataset $\hat{\mathbf{X}}_{ini}$, the number of iterations $LOOP$.
Output: The estimated dataset $\hat{\mathbf{X}}$, the corresponding coefficient parameter matrix $\mathbf{W}$.

```
// Initialization.
```
1 Set the residual matrix $\mathbf{Res} = \mathbf{X}_S$, $\hat{\mathbf{X}} = \hat{\mathbf{X}}_{ini}$, and $\Omega = \emptyset$;
2 **for** $m = 1; m \leq LOOP; m++$ **do**
```
    // Fix X̂ to calculate W.
```
3 **for** *each* $\hat{\mathbf{c}}_i \in \hat{\mathbf{X}}$ **do**
4 Calculate inner product ξ_i between $\mathbf{Res}$ and $\hat{\mathbf{c}}_i$ by using Eq. (14);
5 **end**
6 Select the column $\hat{\mathbf{c}}_m$ with the largest inner product ξ;
7 Add column $\hat{\mathbf{c}}_m$ into Ω;
8 Estimate $\hat{\mathbf{X}}$ based on Ω by using Eq. (15);
9 Update residual matrix $\mathbf{Res}$ by using Eq. (17);
```
    // Fix W to calculate X̂.
```
10 Calculate $\hat{\mathbf{X}}$ by using least-squares fitting in Eq. (18);
11 **end**
12 **return** $\hat{\mathbf{X}}$ and $\mathbf{W}$;

We summarize the proposed OMP-based imputation algorithm in Alg. 3. It takes the selected dataset $\mathbf{X}_S$, the initial estimated dataset $\hat{\mathbf{X}}_{ini}$, and the number of iterations $LOOP$ as inputs, and outputs the estimated dataset $\hat{\mathbf{X}}$ and the corresponding coefficient parameter matrix $\mathbf{W}$. Alg. 3 first initializes the residual matrix $\mathbf{Res} = \mathbf{X}_S$, $\hat{\mathbf{X}} = \hat{\mathbf{X}}_{ini}$, and $\Omega = \emptyset$ (line 1). Then in each iteration, for each column $\hat{\mathbf{c}}_i$ in $\hat{\mathbf{X}}$, the algorithm calculates inner product ξ_i between $\mathbf{Res}$ and $\hat{\mathbf{c}}_i$ using Eq. (14) (lines 3-5). The column $\hat{\mathbf{c}}_m$ with the largest inner product is selected and added into Ω (lines 6-7). The algorithm estimates $\hat{\mathbf{X}}$ based on Ω using Eq. (15), and updates $\mathbf{Res}$ using Eq. (17) (lines 8-9). Finally, Alg. 3 derives $\hat{\mathbf{X}}$ using the least-squares fitting in Eq. (18) (line 10).

IV. Evaluation

This section verifies the effectiveness of the proposed data availability optimization method using two different datasets. For each dataset, we inject missing values by randomly removing a fixed percentage of data from the complete dataset. For evaluation, we compare and analyze the performance of the proposed method and SOTA approaches in multiple aspects.

A. Experimental Settings

We conduct numerical experiments on an NVIDIA Tesla equipped with P100 GPU and 16GB DDR4 memory. The machine runs a Windows version of Matlab x64. We employ two datasets to conduct our experiments. One is the air quality (AQ) dataset which contains data such as PM2.5, PM10, and air pressure collected by a wireless sensor network deployed across Krakow, Poland [25]. The sensor network consists of 56 low-cost sensors whose sampling rates are

all 1 sample/hour. In the applications of our interests, we use normalized air quality data of 720 hours. The other is the environmental sensor telemetry (EST) dataset which contains environment data such as temperature and humidity from multiple identical, custom-built, breadboard-based sensor arrays [26]. EST dataset consists of 405,184 rows and 31 columns of data spanning the period from 07/12/2020 00:00:00 UTC to 07/19/2020 23:59:59 UTC.

In the validation of our scheme, we set K values and other related parameters in the initial guess procedure as in [19]. We set 40 and 20 as the number of selected features for AQ and EST datasets, respectively. The proposed method is compared to a baseline method NI and two SOTA approaches IKNN [27] and ITIM [9] which are described as follows.

- NI (No Imputation) is a baseline method that does not execute any imputation mechanism to improve data availability when data missing occurs.
- IKNN (Iterative K Nearest Neighbors) [27] is an imputation approach that iteratively utilizes K Nearest Neighbors algorithm to estimate missing values and improve data availability.
- ITIM (Iterative Imputation) [9] is an iterative data availability optimization approach that uses least-squares fitting to impute missing values in raw datasets.

We use MSE (given in Eq. (4)) and COST to measure the performance of the proposed scheme and other three comparative algorithms. MSE indicates the accuracy of the imputation, which is actually the difference between the estimated values and the corresponding real values on missing parts. COST is defined as the MSE between the estimated values and the corresponding real values on non-missing parts of the dataset. COST quantifies the impact of the imputation on non-missing parts of the dataset.

B. Results of Air Quality Dataset

Fig. 2 and Fig. 3 plot the MSE and COST of the proposed approach on the AQ dataset when varying the data missing rate, respectively. Note that MSE and COST values in the first iteration are in fact the performance of the initial guess based on K-Means. From the figures, we can obtain some important observations. Firstly, the MSE and COST of the proposed approach are high in the first iteration, then with more iterations, both MSE and COST decrease. This reveals that the imputation accuracy of the initial guess can be further improved by the OMP-based imputation. Secondly, with missing rate increases, both MSE and COST increase, and the number of iterations needed for convergence also increases. This indicates that the imputation becomes harder when the data missing rate increases. Note that as the data missing rate increases, the optimization problem given in Eq. (6) is less constrained, and overfit may occur. Therefore, a small COST is not necessarily equivalent to a more accurate imputation.

Fig. 4 and Fig. 5 compare the proposed approach with a baseline method NI and two SOTA approaches IKNN [27] and ITIM [9] in terms of MSE and COST on the AQ dataset, respectively. Since the method NI does not change any values in datasets with missing values, it has no COST value. It

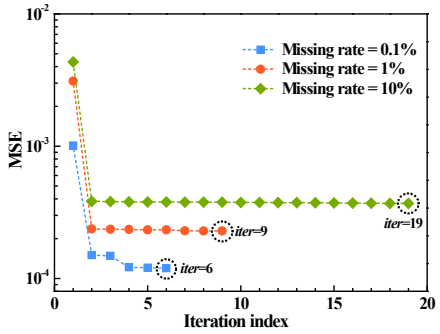

Figure 2: MSE of the proposed approach under varying data missing rates on the AQ dataset.

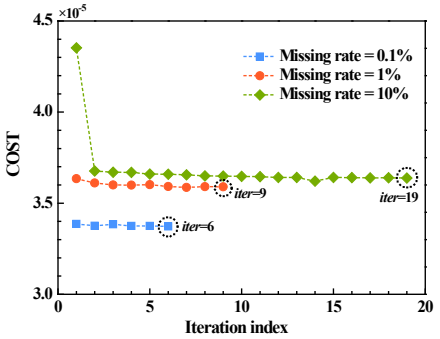

Figure 3: COST of the proposed approach under varying data missing rates on the AQ dataset.

can be seen from the figures that the proposed method can achieve the smallest MSE and COST values under different data missing rates. Compared to NI, IKNN, and ITIM, the proposed data availability optimization approach can reduce MSE by up to 99.59%, 94.4%, and 50.6%, respectively. This is because the proposed approach first uses an initial guess to obtain a preliminary estimation and then utilizes OMP-based imputation to iteratively improve the imputation accuracy.

Specifically, compared to NI, the proposed method uses estimated values to replace missing values such that data availability is increased. Compared to IKNN, the proposed method utilizes the relationship between multiple features in datasets and therefore can estimate missing values more accurately. Compared to ITIM, the proposed method exploits the OMP algorithm to efficiently solve the optimization problem and hence derive accurate estimation.

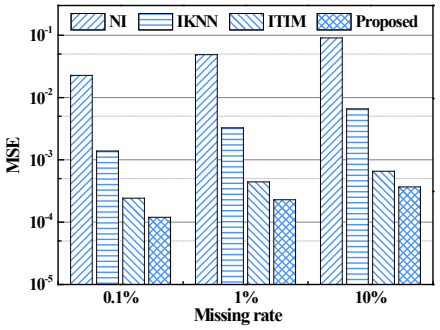

Figure 4: Compare MSE of the proposed approach with NI, IKNN [27] and ITIM [9] on the AQ dataset.

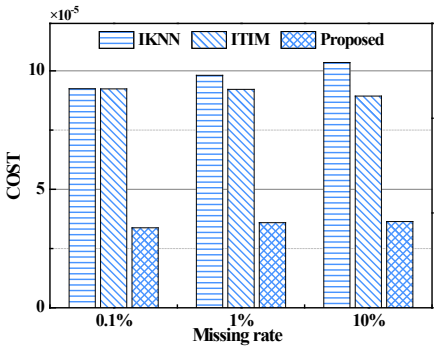

Figure 5: Compare COST of the proposed approach with NI, IKNN [27] and ITIM [9] on the AQ dataset.

Table I further compares the proposed approach with the other 3 benchmarking methods in terms of runtime (s) under different data missing rates on the AQ dataset. It can be seen from the table that the runtime of the NI method is 0. This is because the NI method does not execute any imputation when data missing occurs. The IKNN method is less time-consuming compared to the ITIM and the proposed method. The runtime of the proposed method and the ITIM method is similar. The reason is these two methods need multiple iterations for highly accurate estimation. However, the runtime remains affordable in practice, since in this case, the maximum runtime is 13.21s.

Table I: Runtime (s) of 4 methods on the AQ dataset when data missing rate is 0.1%, 1%, and 10%, respectively.

Runtime (s)	**NI**	**IKNN** [27]	**ITIM** [9]	**Proposed**
0.10%	0	0.29	4.62	4.83
1%	0	1.93	6.68	6.72
10%	0	2.94	13.08	13.21

C. Results of Environmental Sensor Telemetry Dataset

Fig. 6 and Fig. 7 present the MSE and COST of the proposed approach on the EST dataset when varying the data missing rate, respectively. From these figures, we can see that similar to the situation in the AQ dataset, the MSE and COST of the proposed approach are high in the first iteration. With more iterations, both MSE and COST then decrease. In other words, the proposed method obtains an initial guess and iteratively improves the imputation accuracy by using the OMP-based method.

With missing rate increases, both MSE and COST increase, and the number of iterations needed for convergence also increases. This reveals that the imputation becomes harder when the data missing rate increases. In addition, compared to the number of needed iterations for processing the AQ dataset, that of processing the EST dataset is more. This is because the EST dataset is more complicated and more difficult to impute. Estimating missing values in the EST dataset needs more iterations for computation.

Fig. 8 and Fig. 9 compare the proposed approach with NI, IKNN [27] and ITIM [9] in terms of MSE and COST on the EST dataset, respectively. Since NI does not change any values in datasets with missing values, it has no COST

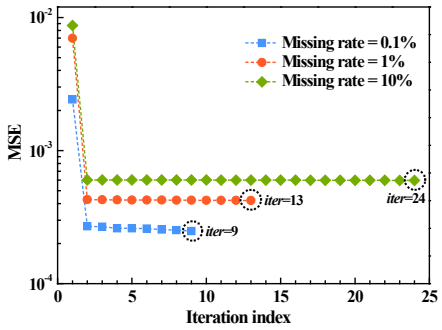

Figure 6: MSE of the proposed approach under varying data missing rates on the EST dataset.

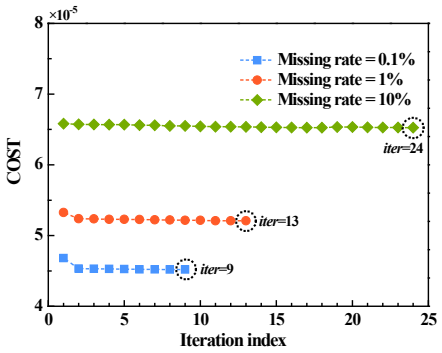

Figure 7: COST of the proposed approach under varying data missing rates on the EST dataset.

value. It can be seen from the figures that similar to the case of the AQ dataset, the proposed method can achieve the smallest MSE and COST values under different data missing rates on the EST dataset. Compared to NI, IKNN, and ITIM, the proposed data availability optimization can reduce MSE by up to 99.65%, 96.73%, and 61.2%, respectively. In addition, compared to NI and IKNN, with the increasing data missing rate, the performance improvement becomes more significant. This is because when the number of missing values increases, there are fewer complete samples for NI and IKNN to utilize. Compared to ITIM, the proposed imputation method obtains an appropriate initial guess and iteratively improves the imputation accuracy using the OMP-based imputation.

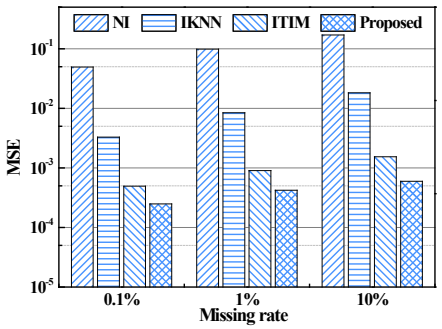

Figure 8: Compare MSE of the proposed approach with NI, IKNN [27] and ITIM [9] on the EST dataset.

Table II lists the runtime (s) of the proposed approach and other three comparative methods under varying data missing rates on the EST dataset. Since NI does not execute any

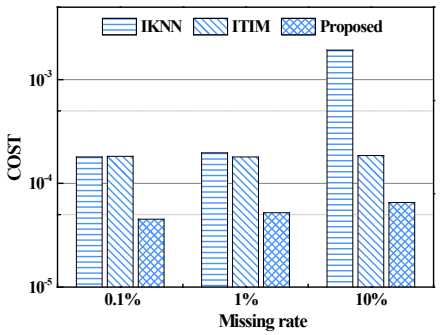

Figure 9: Compare COST of the proposed approach with NI, IKNN [27] and ITIM [9] on the EST dataset.

imputation when data missing occurs, its runtime is 0. The runtimes of the proposed approach and ITIM are higher than that of IKNN since these two methods need multiple iterations for highly accurate estimation. However, the runtime of the proposed method remains affordable in practice, since in this case, the maximum runtime is 17.54s.

Table II: Compare the proposed method with NI, IKNN [27] and ITIM [9] in terms of Runtime (s) on the EST dataset.

Runtime (s)	**NI**	**IKNN** [27]	**ITIM** [9]	**Proposed**
0.10%	0	0.68	9.88	9.79
1%	0	4.82	12.39	12.4
10%	0	8.32	17.58	17.54

V. Conclusion

In this paper, we aim to solve the problem of improving CPS data availability by accurately imputing missing values in raw datasets. Considering that using raw data for computation may bring huge computation overheads, we design an SVD-based approach to select crucial features from raw data. To speed up the convergence and reduce the probability of falling to the local optimum, we make an appropriate initial guess for each missing value by using a K-Means-based imputation approach. We further propose an OMP-based data availability optimization method to iteratively adjust the initial guess and improve the imputation accuracy. Results on two datasets show that compared to three benchmarking methods, the proposed scheme can effectively reduce imputation error without incurring huge computation overheads.

References

[1] F. Farivar, M. S. Haghighi, A. Jolfaei, and M. Alazab, Artificial Intelligence for Detection, Estimation, and Compensation of Malicious Attacks in Nonlinear Cyber-Physical Systems and Industrial IoT, *IEEE Trans. Industrial Informatics*, vol. 16, no. 4, pp. 2716-2725, 2020.

[2] X. Wang, L. T. Yang, X. Xie, J. Jin, and M. J. Deen, A Cloud-Edge Computing Framework for Cyber-Physical-Social Services, *IEEE Communications Magazine*, vol. 55, no. 11, pp. 80-85, 2017.

[3] Y. Bai, Y. Huang, G. Xie, R. Li, and W. Chang, ASDYS: Dynamic Scheduling Using Active Strategies for Multifunctional Mixed-Criticality Cyber–Physical Systems, *IEEE Trans. Industrial Informatics*, vol. 17, no. 8, pp. 5175-5184, 2021.

[4] L. Ren, Z. Meng, X. Wang, L. Zhang, and L. T. Yang, A Data-Driven Approach of Product Quality Prediction for Complex Production Systems, *IEEE Trans. Industrial Informatics*, vol. 17, no. 9, pp. 6457-6465, 2021.

[5] T. Zhang, Y. Zou, X. Zhang, N. Guo, and W. Wang, Data-Driven Based Cruise Control of Connected and Automated Vehicles Under Cyber-Physical System Framework, *IEEE Trans. Intelligent Transportation Systems*, vol. 22, no. 10, pp. 6307-6319, 2021.

[6] D. Luebbers, U. Grimmer, and M. Jarke, Systematic Development of Data Mining-based Data Quality Tools, *Proceedings of International Conference on Very Large Data Bases*, pp. 548-559, 2003.

[7] S. Tee, P. Bowen, P. Doyle, and F. Rohde, Factors Influencing Organizations to Improve Data Quality in Their Information Systems, *Accounting and Finance*, vol. 47, no. 2, pp. 335-355, 2007.

[8] I. Larsen, M. Småstuen, T. Johannesen, F. Langmark, D. Parkin, F. Bray, and B. Møller, Data Quality at the Cancer Registry of Norway: An Overview of Comparability, Completeness, Validity and Timeliness, *European Journal of Cancer*, vol. 45, no. 7, pp. 1218-1231, 2009.

[9] L. Li, Y. Liu, T. Wei, and X. Li, Exploring Inter-Sensor Correlation for Missing Data Estimation, *The 46th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2108-2114, 2020.

[10] L. Chen, G. Li, G. Huang and P. Shi, A Missing Type-Aware Adaptive Interpolation Framework for Sensor Data, *IEEE Trans. Instrumentation and Measurement*, vol. 70, pp. 1-15, 2021.

[11] M. Weber, M. Turowski, H. K. Çakmak, R. Mikut, U. Kühnapfel, and V. Hagenmeyer, Data-Driven Copy-Paste Imputation for Energy Time Series, *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5409-5419, 2021.

[12] Z. Yao, and C. Zhao, FIGAN: A Missing Industrial Data Imputation Method Customized for Soft Sensor Application, *IEEE Trans. Automation Science and Engineering*, pp. 1-11, 2021.

[13] SPSS, Missing data: The Hidden Problem, *White Paper*, Available: http://www.whitepapercentral.com/browse/marketing/missing-data-the-hidden-problem/ 2009.

[14] R. Razavi-Far, E. Hallaji, M. Farajzadeh-Zanjani, R. Aljoudi, and M. Saif, A Critical Study on the Impact of Missing Data Imputation for Classifying Intrusions in Cyber-Physical Water Systems, *47th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1-6, 2021.

[15] A. Farhangfar, L. A. Kurgan, and W. Pedrycz, A Novel Framework for Imputation of Missing Values in Databases, *IEEE Trans. Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 37, no. 5, pp. 692-709, 2007.

[16] X. Wang, R. Yuan, L. Ren, L. T. Yang, and M. J. Deen, QTT-DLSTM: A Cloud-Edge-Aided Distributed LSTM for IIoT Big Data, *IEEE Trans. Neural Networks and Learning Systems*, DOI: 10.1109/TNNLS.2022.3140238, 2022.

[17] D. Little, and D. Rublin, Statistical Analysis with Missing Data, *New York:Wiley*, pp. 381, 1987.

[18] U. Shrestha, A. Alsadoon, P. Prasad, S. Aloussi, and O. Alsadoon, Supervised Machine Learning for Early Predicting the Sepsis Patient: Modified Mean Imputation and Modified Chi-square Feature Selection, *Multimedia Tools and Applications*, vol. 80, pp. 20477–20500, 2021.

[19] Z. Liao, X. Lu, T. Yang, and H. Wang, Missing Data Imputation: A Fuzzy K-means Clustering Algorithm over Sliding Window, *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 133-137, 2009.

[20] M. Lei, A. Labbe, Y. Wu, and L. Sun, Bayesian Kernelized Matrix Factorization for Spatiotemporal Traffic Data Imputation and Kriging, *IEEE Trans. Intelligent Transportation Systems*, pp. 1-13, 2022.

[21] Y. Gong, Z. Li, J. Zhang, W. Liu, Y. Yin, and Y. Zheng, Missing Value Imputation for Multi-view Urban Statistical Data via Spatial Correlation Learning, *IEEE Trans. Knowledge and Data Engineering*, pp. 1-1, 2021.

[22] P. Khan, Y. Byun, S. Lee, and N. Park, Machine Learning Based Hybrid System for Imputation and Efficient Energy Demand Forecasting, *Energies*, vol. 13, no. 11, pp. 2681, 2020.

[23] Y. Zhang, P. Thorburn, W. Xiang, and P. Fitch, SSIM—A Deep Learning Approach for Recovering Missing Time Series Sensor Data, *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6618-6628, 2019.

[24] X. Li, Finding Deterministic Solution from Underdetermined Equation: Large-scale Performance Variability Modeling of Analog/RF Circuits, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 11, pp. 1661-1668, 2010.

[25] "Air Quality Dataset", [Online]. Available: https://www.kaggle.com/datasets/datascienceairly/air-quality-data-from-extensive-network-of-sensors.

[26] "Environmental Sensor Telemetry Dataset", [Online]. Available: https://www.kaggle.com/datasets/garystafford/environmental-sensor-data-132k.

[27] Z. Sahri, R. Yusof, and J. Watada, FINNIM: Iterative Imputation of Missing Values in Dissolved Gas Analysis Dataset, *IEEE Trans. Industrial Informatics*, vol. 10, no. 4, pp. 2093-2102, 2014.