This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

FINNIM: Iterative imputation of missing values in Dissolved Gas Analysis Dataset

Zahriah Sahri, Rubiyah Yusof, and Junzo Watada, *Member, IEEE*.

***Abstract*—Missing values are a common occurrence in a number of real world databases, and statistical methods have been developed to deal with this problem, referred to as missing data imputation. In the detection and prediction of incipient faults in power transformers using Dissolved Gas Analysis (DGA), the problem of missing values is significant and has resulted in inconclusive decision making. This study proposes an efficient non-parametric iterative imputation method, named FINNIM, which comprises of three components : the imputation ordering, the imputation estimator and the iterative imputation. The relationship between gases and faults and the percentage of missing values in an instance are used as a basis for the imputation ordering; whilst the plausible values for the missing values are estimated from k-nearest neighbour instances in the imputation estimator; and the iterative imputation allows complete and incomplete instances in a DGA dataset to be utilized iteratively for imputing all the missing values. Experimental results on both artificially inserted and actual missing values found in a few DGA datasets demonstrate that the proposed method outperforms the existing methods in imputation accuracy, classification performance and convergence criteria at different missing percentages.**

***Index Terms*—Dissolved gas analysis, iterative imputation, imputation ordering, k-nearest-neighbour, missing values, missing data imputation.**

I. Introduction

Power transformers are essential equipments to transmit and distribute electrical energy through interconnected power systems. While in-service, transformers may face electrical or thermal disturbances that cause faults such as arcing, partial discharge, and thermal to surface. These faults will release several gases commonly known as fault gases: hydrogen (H_2), acetylene (C_2H_2), ethylene (C_2H_4), methane (CH_4), ethane (C_2H_6), carbon monoxide (CO) and carbon dioxide (CO_2) that stay dissolved at above threshold values in the insulating oil of a transformer. If left untreated for long, these faults could induce transformer failure and disrupt power supply to industries, businesses, and homes; causing huge financial losses or triggering worst impacts such as explosions, loss of human lives, or environmental disasters. Therefore, to minimize these risks, early detection of incipient faults in a transformer is of vital importance. In industrial practice and for oil-filled transformers, dissolved gas analysis (DGA) is an efficient tool for such purpose since it can give warning about an impending problem, and helps provide an early diagnosis and identify the necessary preventive actions. These are achieved through a) periodic sampling of insulation oil, b) extracting the dissolved gases, calculating and analyzing the concentration of these gases, their gassing rates and the ratios of certain gases and c) finally, the identification of the possible fault types through conventional methods such as IEC ratios, Rogers ratio, Doernenburg ratio and the Duval Triangle.

The above diagnostic DGA ratio methods identify fault types using the ratios of certain fault gases and each ratio is assigned to one or more numerical thresholds. These thresholds are coded and mapped to specific faults. In some cases, measured gas concentrations or ratios may be incomplete and thus do not match any predefined threshold. As a result, fault that occurs inside a transformer may be classified unknown or inconclusive [1]. One of the reasons for incomplete gas concentrations is missing values for some of the fault gases. Missing values in DGA can occur for various reasons, such as acetylene evaporates quickly, the existence of contamination on the surface of the platinum alloy of a gas meter, and some transformer faults generate only a few but not all of the fault gases. Apart from reducing the effectiveness of the DGA ratio methods as stated earlier, missing values can also affect the performance of machine learning algorithms that learns from DGA data to diagnose faults, such as support vector machines (SVM), neural network, and fuzzy logic. As missing values increase in a dataset, the prediction accuracy of the learning algorithms decreases in tandem, as documented in [2] and [3].

This problem can be managed in many different ways from simply deleting the DGA instances containing missing values, although this may substantially reduce the number of available instances, especially if the missing rate is high, to reporting only complete instances (instances without missing values) of DGA, although this is inappropriate because valuable information of the incomplete instances (instances with missing values) is lost. The best solution is to attempt to accurately estimate and fill-in the missing values with available data (”imputation”), but to our knowledge, only [4] estimated the missing values in DGA dataset using support vector machine regression (SVR), which increased the accuracy of their Naive-Bayes classifier. However, their approach requires dispersion of the continuous values of the fault gases before estimation

Manuscript received November 6, 2013; revised April 3, 2014, revised June 17, 2018. Accepted for publication August 8, 2014.

Zahriah Sahri is with the Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia (e-mail: szahriah@utem.edu.my).

Rubiyah Yusof is with the Universiti Teknologi Malaysia, Jalan Semarak, 54100 Kuala Lumpur, Malaysia (phone: +603-26913710; fax: +603-26970815; e-mail: rubiyah@ic.utm.my).

Junzo Watada is with the Graduate School of Information, Production and Systems, Waseda University, Kitakyushu 808-0135, Fukuoka, Japan (e-mail: junzow@osb.att.ne.jp).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

takes place, a pre-processing step that can lead to information loss. In addition, only complete instances were used to estimate plausible values for the missing values, whereas [5], [6] have demonstrated that if the information within incomplete instances is utilized as well then the estimation bias caused by the only few complete instances can be reduced. There is, thus considerable need to develop an efficient method to estimate missing values in a DGA dataset minus the information loss and estimation bias found in [4] and with the missing values being filled in, the reduced performance of a learning algorithm is duly arrested.

As an attempt to realize the above objectives, we introduce an efficient imputation algorithm called FINNIM that iteratively imputes all missing values by utilizing complete and incomplete instances in a DGA dataset. It has three different components mainly 1) imputation ordering, 2) imputation estimator and 3) iterative imputation. The task of imputation ordering is to assign an imputation order to a missing value in a dataset. The assignment is made based on the relationship between gases and faults and the percentage of the missing values in an instance. In the imputation estimator, the plausible values to replace the missing values are estimated using the k nearest-neighbor (kNN) algorithm. This estimation process is done iteratively in the third component, where for the first iteration, the kNN instances are selected from complete instances only, and the next imputations thereafter use all instances (imputed and complete) until convergence is reached. The primary contributions of this paper lies in the following: 1) a unified framework to impute missing values found in a DGA dataset that simultaneously facilitates the principle of DGA method, applies a non-parametric approach to predict the missing values, and utilizes all information contained in a DGA dataset; 2) a unified framework that works collectively to ensure that the data distribution in a dataset is preserved after data imputation; 3) an ordering of missing values which preserves continuous data and the feature-class relationship in a dataset.

Comparative studies between FINNIM and other well-established methods such as single regression (REG), mean/mode (MEAN), expectation-maximization (EM), and multiple imputation (MI) are presented. The first comparison is made to evaluate the convergence ability between our proposed method and the EM method. Only these two methods apply iterative imputation, thus the convergence comparison between the EM method and FINNIM. Next, the accuracy of each imputation method is evaluated using the normalized root mean square error (NRMSE) which calculates the deviation between the estimated and the actual values. Imputation method with the lowest NMRSE produces the most accurate estimates. According to [7], one desirable characteristic of an imputation method is the ability to improve the classification performance of a learning algorithm. Therefore, the "before-and-after" experiment where the accuracy of SVM learned on the original incomplete dataset and that learned on the imputed dataset is compared to validate the effectiveness of each imputation method in meeting the aforementioned characteristic. Experimental results show the robustness and effectiveness of the proposed method.

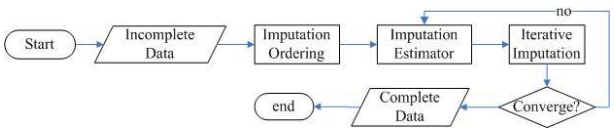

Fig. 1. FINNIM Imputation Accuracy on LITGY dataset

TABLE I
DGA DATASET

Instances Features	f_1	f_2	f_3	f_4	y
X_1	x_{11}	x_{12}	?	x_{14}	1
X_2	x_{21}	x_{22}	x_{23}	x_{24}	2
X_3	?	x_{32}	?	x_{34}	1
X_4	x_{41}	x_{42}	x_{43}	x_{44}	3
X_5	x_{51}	x_{52}	x_{53}	x_{54}	3

The remaining part of the paper consists of the following. Section II details out the proposed method that is FINNIM. The efficiency of the proposed method is demonstrated using three different comparisons and the results are analyzed in Section III. Section IV provides some conclusions of the paper.

II. THE FINNIM ALGORITHM

This section presents the FINNIM method that capitalizes on the characteristics of DGA data and utilizes all available information when estimating missing values in a DGA dataset. It consists of three processes; first is the imputation ordering; second is the imputation estimator, and third is the iterative imputation process as shown in Fig. 1.

In general, a dataset D is represented as a set of data points with label, $\{X_i, y_i\}_{(i=1)}^{n}, y_i \in \{1, \cdots, c\}$ as illustrated in Table I where n is the number of instances. $X_i = \{f_j\}_{j=1}^{m}$ is an instance with m number of features, and the symbol ? represents a missing value. An incomplete instance is an instance that has one or more missing values such as X_1 and X_3, whilst a complete instance contains no missing values such as X_2, X_4 and X_5. Let us call D_t the subset of D that contains only complete instances, D_u is the subset of D that contains only incomplete instances. Here, D represents a DGA dataset with n samples X_i where each sample is labeled with a fault type y_i and contains m dissolved gases (f_j).

A. Imputation Ordering

Table I has three missing values (denoted as X_1f_3, X_3f_1, and X_3f_3) in different features and in different instances. This raises the question whether imputing X_1f_3, X_3f_1 and X_3f_3 arbitrarily makes no difference to the estimation performance, or giving a preference to one (e.g X_3f_3) to be imputed first will produce better estimation values for all. References ([8], [9], [10]) proved that the latter approach was better. Because a DGA dataset may contain instances with multiple missing values as in Table I, this study proposes that all missing values in the DGA dataset are ordered so as to provide hierarchy of imputation for each missing value.

DGA is a method that relies on the strong presence of a few dissolved gases (features) to determine a fault type (class). Thus, the stronger the relationship between a feature and the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

class is, the more worthwhile it is to impute the missing values of that feature. Other imputation ordering techniques [9], [11] did not take into consideration the relationship of features and classes. For example, [9] proposed lexicographic ordering in which the missing values were re-arranged based on the number of missing values in columns as the first criterion, and the number of missing values in rows as the second criterion. This rearrangement could distort the data distribution of a dataset whereas several authors [10], [11] have argued that it is more important that such imputations produce a workable estimate that least distorts the values that are actually present.

To preserve both the data distribution and the continuous values in a DGA dataset after the imputation process, this study exploits the strong relationship between features and classes (also known as feature-class relationship) as the main criterion to produce the imputation order for each missing value in a DGA dataset. In order to represent the degree of feature-class relationship, each feature is given a weight I. Thus a method that calculates I without having to discretize the continuous values into discrete values is preferred. One of the main issue of the proposed imputation ordering method is to measure the weight I which is actually the distance measure between features and classes. Fisher score, one of the popular methods for determining the most relevant features for classification, is chosen for measuring I since it works with multiclass dataset and continuous data. In addition, to the best of our knowledge, Fisher score has not been used as a criterion for generating imputation order in the literature.

Let n_p denote the number of data points X_i in class p, $p = 1, \cdots, c$, $p \in y_i$. Let $\mu_p$and σ_p be the mean and variance of the class p, corresponding to the jth feature f_j. Let μ and σ^2 denote the mean and variance of the whole data set. Then, the weight I for the feature f_j is as below:

$$I_j = F(f_j) \tag{1}$$

where $F(f_j)$ is the Fisher score of the jth feature and is defined as follows:

$$F(f_j) = \frac{\sum_{p=1}^{c} n_p(\mu_p - \mu)^2}{\sum_{p=1}^{c} n_p \sigma_p^2} \tag{2}$$

Because a higher I indicates a stronger relationship, thereby an ordered list of features based on descending I is established where missing values in features with higher I are imputed earlier.

However, an I-ordered feature may contain more than one missing value as in the features of Table I. This gives rise to the question of which missing value in an I-ordered feature should be imputed first. The fact that an imputation algorithm is an instance-based learning algorithm has compelled us to consider the missing rate of the instances as the secondary criterion, where instances with lower missing rates get higher imputation priority. Let num_{miss} denotes the number of missing values in X_i, and $num_{feature}$ denotes the number of features in X_i. The missing rate of an instance (denoted as R_i, where i is the index of the instance) is defined as

$$R_i = \frac{num_{miss}}{num_{feature}} \tag{3}$$

Using (3), all missing values in an I-ordered feature are sorted in ascending R. A missing value where its instance has the lowest R gets imputed first.

Using Table I as an example and assuming that each feature f_j has a weight I_j such that $I_4 \geq I_3 \geq I_2 \geq I_1$. Only f_1 and f_3 contain missing values, and since $I_3 \geq I_1$, all missing values in f_3 are to be imputed first than the missing values in f_1. f_3 has two missing values at instance X_1 and X_3, respectively. Therefore, the secondary criterion is applied and has resulted in $R_1 \geq R_3$. Thus, $X_1 f_3$ is the first to be imputed, followed by $X_3 f_3$. Because f_1 has only one missing value $X_3 f_1$, than R become insignificant. Therefore, $X_3 f_1$ is the last to be imputed.

The proposed ordering in this study has two benefits:

a) it utilizes the relationship between features and classes which helps preserve the correlation between features and classes.
b) it avoids discretization of continuous values which prevents loss of useful information.

B. Imputation Estimator

The next step after obtaining the ranked-list of missing values from the imputation ordering component is to fill-in the missing values with estimated values from the pool of complete instances. Before doing so, this study adopts a few considerations as follows.

a) the estimated values should be as close as possible to the original (unobserved) values so that the covariance or correlation to other variables are preserved.
b) no pre-processing of continuous values so that loss of useful information is avoided.
c) non-parametric method is preferred since parametric ones are based upon certain assumptions such as the population of data values and the prior distribution for the model parameters. These assumptions are difficult to realize in real-world applications.

As one of the popular non-parametric imputation methods, k-nearest neighbours (kNN) algorithm fulfills with all of the above considerations as evident in [2] and [7], thus becomes the proper choice for the estimation task. This method searches the kNN of the instances with missing value(s) and replaces the missing value(s) by the mean or mode value of the corresponding feature values of the kNN. The quality of the estimated values obtained from the kNN approach largely depends on two important parameters: the choice of (k), the number of neighbors used and the appropriate distance metric. Simulation results have demonstrated that for small datasets, k = 10 is the best choice (Acuna et al., [12]), while Troyanskaya et al. ([13]) observed that kNN is insensitive to values of k in the range of $10-20$. Therefore, this study replaces the missing values with estimated values from $1-10$ nearest neighbors depending on the size of datasets. To get the best k, a simulation is performed, by randomly changing the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

observed values into missing values, estimating these missing values based on different choices of k, and measuring the error between the imputed and the actual observed values. The k that produces the smallest error can be considered as the best. In this study, the Manhattan distance is used to measure similarity between instances because of its simplicity in calculation and easy decomposition into contributions made by each variable (for the Euclidean distance, we would need to decompose the squared distance). Most importantly, Manhattan distance is more robust (since the distances are not squared) to the influence of outliers compared to higher order distance metrics including Euclidean distance and Mahalanobis distance.

The steps of kNN estimation are as follows:

1) choose k, the number of nearest neighbours to be selected.
2) using the Manhattan distance metric, calculate the distance between the instance with the to-be-imputed missing value with another instance. Let $X_i = \{x_{i1}, \cdots, x_{im}\}$ denotes the instance with the to-be-imputed missing value, and $X_q = \{x_{q1}, \cdots, x_{qm}\}$ be the other instance. The Manhattan distance between X_i and X_q is calculated using (4)

$$dist(X_i, X_q) = \sum_{j=1}^{m} |x_{ij} - x_{qj}| \tag{4}$$

where m is the number of features in X_i and X_q, and x_{ij} is the jth feature of instance X_i and x_{qj} is the jth feature of instance X_q.
3) Repeat step 2 to compute the distance between X_i with each remaining instance in the dataset.
4) sort in ascending order (based on the calculated Manhattan distance values) all X_q excludes X_i.
5) select the top k instances from the sorted list as the k-nearest neighbours to X_i. These k-nearest neighbours are $X_{kNN} = \{X_1, X_2, \cdots, X_k\}$.
6) Let x_{ij} be the to-be-imputed missing value in X_i. Then the estimated value is obtained from (5)

$$x_i j = \frac{\sum_{l=1}^{k} x_{lj}}{k} \tag{5}$$

where k is the number of nearest neighbours, x_{lj} is the jth feature of instance X_l, and $X_l \in X_{kNN}$.

C. Iterative Imputation

If the proposed imputation stops at the second component of FINNIM, then all missing values are imputed only once. This single-imputation approach, however, tends to overstate precision because it omits the between imputation component of variability [14]. Multiple imputation, where several likelihood choices for imputing the missing values are computed, incorporates data variability by replacing a missing datum with two or more values representing a distribution of likely values. As such, this study adds iterative imputation as the third component, where each missing datum is imputed using the imputation estimator iteratively until convergence is reached. This study adopts the definition found in [8] which concludes that when the change in estimated values in successive iterations is zero or trends to a value which trends fast and stably to zero means that the method has converged. This component is divided into first iteration and successive iterations as elaborated below.

1) First Iteration: For the first iteration, many studies [5], [6] used the MEAN method to fill in the missing values but this method can distort the original data distribution since it causes the missing values to be artificially close to each other. This motivates us to use our imputation estimator as the imputation method in the first iteration since it takes into account the data correlation. In this iteration, a missing value x_{ij} in D_u is selected for imputation from the ordered list of missing values. Then, the imputation estimator is executed. Here, the candidates for the k-nearest neighbours are selected from instances in D_t that have the same label as the label of X_i. These steps are repeated to each missing value in D_u. The final outcome of the first iteration is a filled and complete dataset of D_u. Let $D_{u,1}$ be the filled and complete dataset of D_u and D_u retains its original instances.

2) Successive Iterations: For the subsequent iterations, the same steps in the first iteration are executed. However, the candidates for the k-nearest neighbours of x_{ij} in D_u are selected from instances in $(D_t + D_{(u,s-1)})$ that have the same label as the label of X_i. If s is the current number of iteration, then $D_{u,s-1}$ is the outcome of the previous iteration. Each iteration s will produce $D_{u,s}$ dataset. With the inclusion of the imputed instances, all information are now used to estimate the missing values in a DGA dataset. This phase stops when the change in the estimated values drop to zero or does not drop all the way to zero and only trends to a value which trends fast and stably to zero in non-parametric models.

D. The FINNIM Algorithm

Below is the summarized algorithm for FINNIM.

input:
$D = \{X_i, y_i\}_{i=1}^{n}$ //an incomplete dataset with n instances
output: D_{comp} //complete and imputed dataset of D
begin
Imputation Ordering:
  for each feature f_j in X_i
    I_j = Fisher-score of f_j;
  end for
  for each instance X_i in D
    R_i = missing rate of X_i;
  end for
  for each missing value x_{ij} in f_j
    if f_j has 1 x_{ij}
      $O(i,j) = I_j$; // $O(i,j)$ is the imputation order of x_{ij}
    else
      $O(i,j) = I_j + R_i$;
  end for
  D_{rank} = dataset that contains only missing values ranked by $O(i,j)$
Imputation estimator
  k = an integer;
  x_{ij} = a missing value;
  X_i = the instance that contains x_{ij};
  for each instance X_q in D
    calculate the Manhattan distance between X_i and X_q using (4);
  end for
  $D_{neighbours}$ = all X_q sorted in ascending Manhattan distance;
  $D_{kNN} = X_1, X_2, \cdots, X_k$ //the top k X_q from $D_{neighbours}$;
  calculate the estimated value for x_{ij} using (5);
First Iteration:
  D_t = subset of D that contains complete instances only;
  D_u = subset of D that contains incomplete instances only;
  $s = 1$;
  for each missing value x_{ij} in $(D_{rank} \cap D_u)$
    class = y_i of X_i;
    D_{class} = subset of D_t that contains instances having label ==class;
    execute the Imputation estimator using D_{class};
  end for
  $D_{(u,s)}$ = imputed and complete dataset of D_u at iteration $s = 1$;
Successive Iterations:
  $D = D_t + D_{(u,s-1)}$
  repeat
    for each missing value x_{ij} in $(D_{rank} \times D_u)$
      class = y_i of X_i;

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

TABLE II
CHARACTERISTICS OF DGA DATASETS USED IN THIS STUDY

	LITZW	LITZM	LITGY	IEC10DB	MAL
A	30	30	50	167	1228
B	5	5	5	7	9
C	5	5	5	6	6
D (%)	0	0	0	27.54	76.07
E (%)	0	0	0	7.96	14.21

A=number of samples, B=number of dissolved gases, C=number of fault types, D=percent of instances with missing values, E=total number of missing values

D_{class} = subset of D that contains instances having label ==class;
execute the Imputation estimator using D_{class};
end for
$s = s + 1$;
$D(u, s)$ = imputed dataset of D_u at iteration s
until convergence
$D_{comp} = D_t + D_{u,s}$ //complete and imputed dataset of D
end

III. EXPERIMENTAL DESIGN AND RESULTS

A. Experimental Design

In this study, three DGA datasets named LITZW, LITZM, and LITGY that contained no missing values were downloaded from [15], [16], and [17] respectively. They were deliberately chosen to help validate the accuracy of FINNIM, EM, MI, REG and MEAN. Randomly simulated missing values were then inserted to each dataset, and the missing rates were fixed at 3%, 6%, and 9%. Missing rate is the total number of missing values over the total number of values in a dataset. Next, each method independently imputed each simulated dataset. However, experiments performed on datasets with randomly inserted missing values may not truly reflect the nature of actual DGA data missing values. All the five imputation methods were, therefore, tested on the IEC10DB and MAL datasets which contain actual missing values. IEC10DB is a benchmark database that contains actual missing values, whilst the MAL dataset was obtained from a local utility company in Malaysia that maintained power transformers all over Malaysia. The characteristic of the five datasets are shown in Table II.

For the implementation of FINNIM, the number of nearest neighbors were selected as k=1,2,3,4,5 for the LITGY, LITZM, and LITZW datasets as most classes have less than ten instances. For the IEC10DB and MAL datasets, nearest neighbors were restricted to k=1,3,5,7,10. For the implementation of MI, the number of repetition was set to M=5 because according to [18], the MI method does not need a large number of repetition for precise estimates. The number of iterations for the EM method was manually selected for each dataset.

B. Evaluation on Convergence

Since FINNIM and EM are iterative imputation methods, it is important to determine at which point additional iterations have no meaningful effect on the imputed values, i.e. how to evaluate the convergence of the two algorithms. For datasets with actual missing values, Table III shows the convergence results where each cell represents the number of iterations needed to converge. For MAL dataset, FINNIM with k=1 and k=5 required less number of iterations than EM. However, the other three k needed more iterations than EM. For the IEC10DB dataset, all k of FINNIM needed less number of iterations to converge than EM.

TABLE III
COMPARISON OF CONVERGENCE BETWEEN FINNIM AND EM USING DATASETS WITH ACTUAL MISSING VALUES

Dataset	FINNIM					EM
	k = 1	k = 3	k = 5	k = 7	k = 10	
MAL	6	49	11	148	77	23
IEC10DB	4	7	23	19	14	38

TABLE IV
COMPARISON OF CONVERGENCE BETWEEN FINNIM AND EM USING DATASETS WITH SIMULATED MISSING VALUES

Dataset		FINNIM					EM
		k = 1	k = 2	k = 3	k = 4	k = 5	
LITGY	3%	4	4	4	4	4	3
	6%	4	4	4	4	5	7
	9%	4	5	5	5	6	10
LITZM	3%	4	4	4	4	4	7
	6%	4	4	4	4	4	6
	9%	4	5	4	5	5	5
LITZW	3%	4	4	4	4	4	6
	6%	4	4	4	7	6	8
	9%	4	4	5	4	4	6

For datasets with artificially inserted missing values, Table IV shows the comparison of performances between FINNIM and EM on these datasets. Here, both methods have almost similar convergence performances where the number of iterations needed is mostly small for all values of k. Nevertheless, FINNIM converged faster than EM in most experimental settings. Only once did EM perform better than FINNIM on the LITGY dataset for all values of k and the missing rate was 3%. However, for the other two datasets, FINNIM for all values of k required a few iterations lesser than EM to converge. Overall, the results on all datasets showed that both methods converge but FINNIM converged faster than EM in most experimental settings. We note that, despite the big difference in the number of iterations needed to converge and the various sizes of the datasets, both of the methods require very minimal time to converge (seconds only) for all values of k and for all missing rates on each datasets. It can be said that, because time is insignificant, convergence rate is not so important as what is more important is the accuracy of the imputed values - how different are they compared to the actual observed values. The next section III-C evaluates the accuracy of the imputation methods mentioned in this study in estimating missing values in different datasets.

C. Evaluation on Imputation Accuracy

The accuracy of a set of imputation methods were compared by computing a statistic quantifying the deviation between the estimated and the true values for each imputation method. These were done using normalized root mean squared error (NRMSE) as recommended by [19] for datasets with continuous variables. The imputation method achieving the smallest NRMSE gives the most correct picture of the complete data

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

TABLE V
NRMSE FOR FINNIM USING DIFFERENT VALUES OF k

Dataset		NRMSE				
		k = 1	k = 2	k = 3	k = 4	k = 5
LITGY	3%	0.74	0.66	**0.52**	0.58	0.62
	6%	0.70	0.66	0.60	0.58	0.60
	9%	0.66	0.65	0.59	0.56	0.59
LITZM	3%	0.74	0.67	1.10	1.43	1.41
	6%	0.69	**0.57**	0.62	0.59	0.64
	9%	0.62	0.61	0.70	0.80	0.86
LITZW	3%	0.58	0.51	**0.44**	0.46	0.56
	6%	0.66	0.59	0.61	0.62	0.74
	9%	0.64	0.62	0.59	0.60	0.73

when estimated values were included. NRMSE is defined by

$$NRMSE = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left[\tilde{e}_{ij} - e_{ij}\right]^2}{\sum_{i=1}^{n}\sum_{j=1}^{m}\left[e_{ij}\right]^2}} \quad (6)$$

where e_{ij} is the original value; $\tilde{e}_{ij}$ is the estimated value, n and m are the total number of rows and columns, respectively. The more NRMSE is, the less is the prediction accuracy. These evaluations were done on the imputed LITGY, LITZM and LITZW datasets.

Because FINNIM used kNN algorithm to estimate the missing values, the effect of the value of k on the NRMSE was evaluated and the best k for each dataset was identified. Table V shows the calculated NRMSE for each k over three datasets. It can be seen that for the three imputed datasets, lower and higher values of k produced higher NRMSE. With lower k, only a small set of correlated instances are used to estimate the missing values while other highly correlated instances are ignored. When k is high, instances which have either very low or no correlation with the instance having missing values will be included in the estimation process. Both scenarios decrease the performance of FINNIM. For the LITGY dataset, k = 3 produced the least NRMSE, while k = 2 was FINNIM the most accurate on the LITZM dataset, and FINNIM required k = 3 to produce the best estimates on the LITZM dataset.

Next, we evaluated the efficiency of FINNIM and the four established methods (EM, MI, REG and MEAN). These comparisons were done using FINNIM at its best performances as identified above. Fig.2, Fig.3 and Fig.4 illustrate the performance of each method for each dataset. It is clear that FINNIM surpassed the four methods for most of the experimental settings. FINNIM was the most accurate at all missing rates for the LITZM dataset. The efficiency of EM was comparable to FINNNIM for all of the datasets. In fact, at 6% missing rate in the LITGY and LITZW datasets, EM produced the least NRMSE values. The MI method was the least efficient of all for all of the datasets, followed by the REG and the MEAN methods. It can be seen that the NRMSE values of FINNIM were the most stable over the whole range of tested missing rates in all of the datasets, a testimony of the robustness of FINNIM. The EM and the MEAN methods were comparably robust to FINNIM. However, the REG and MI methods experienced high fluctuations of NRMSE values over different missing rates, especially at the 9% missing rate. Again, the MI was the least robust of all.

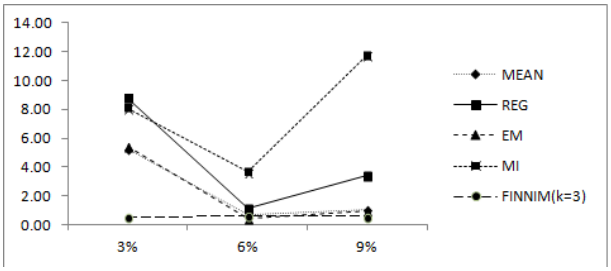

Fig. 2. Imputation Accuracy of FINNIM for the LITGY dataset

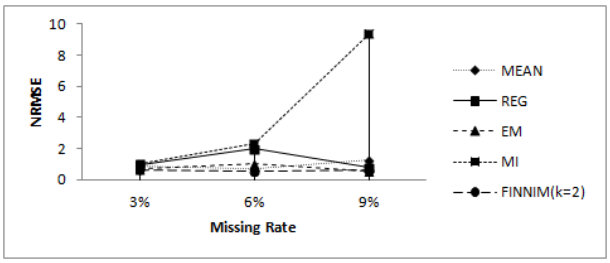

Fig. 3. Imputation Accuracy of FINNIM for the LITZM dataset

For more careful estimation of imputation efficiency, we examined the structure of data after imputation. We calculated the Pearson correlation coefficients for each feature between original data and imputed data. The larger the correlation coefficient is, the better the relationship between original complete data and imputed data is preserved in a feature. Due to lack of space, the correlation coefficients are shown only for each imputation method and for each dataset at 9% missing rate. Table VI, VII, and VIII show that the FINNIM method preserved the structure of the original data set better than the other four methods for many features of the three datasets. In fact, for all features of the LITGY dataset, FINNIM was the most efficient. Interestingly, the MI method was the worst method, congruent with the NRMSE analysis. This column-wise comparison gives us more specific information on the efficiency of imputation method.

D. Evaluation on Classification Accuracy

To evaluate the performance of an imputation method that imputes actual missing values in a dataset, a different approach is required instead of the NRMSE. The fact that the actual complete values are not known has made the NRMSE unsuitable for the evaluation task. This scenario motivates us to use the classification accuracy of a machine learning algorithm as the evaluation criteria, because one desirable characteristic of an imputation method is the ability to improve the classification performance of a learning algorithm [7]. The classification accuracy is defined as:

$$Accuracy = \frac{n_c}{n} \times 100\% \quad (7)$$

where n_c is the number of instances whose class labels being correctly predicted and n is the total number of instances in a test set. Recently, a growing number of researchers have applied SVM for the classification of faults in power

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

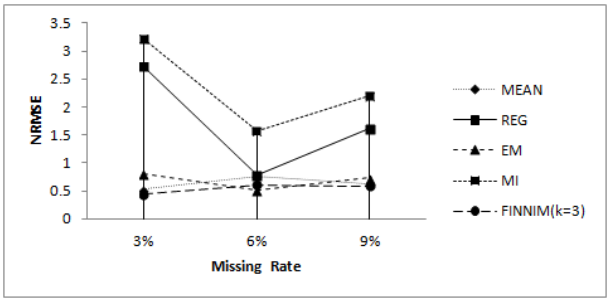

Fig. 4. Imputation Accuracy of FINNIM for the LITZW dataset

TABLE VI
PEARSON CORRELATION COEFFICIENT FOR THE LITGY

	H_2	CH_4	C_2H_6	C_2H_4	C_2H_2
MEAN	0.9907	0.9923	0.9967	0.9930	0.9988
REG	0.9669	0.9655	0.9886	0.9991	0.9873
EM	0.9959	0.9996	0.9897	0.9997	0.9986
MI	0.9101	0.9864	0.9667	0.9603	0.9603
FINNIM	**0.9972**	**0.9999**	**0.9977**	**0.9999**	**0.9999**

transformers and have reported higher classification accuracies with SVM than other widely used learning algorithms, such as the multilayer perceptron neural network, back-propagation neural network and fuzzy logic [20]-[21]. Therefore, this study chose the classification accuracy of SVM to measure the performance of an imputation method on datasets with actual missing values. First, imputed datasets and the original incomplete datasets of IEC10DB and MAL were divided into training and testing subsets where the number of instances for each fault type on both subsets were divided into 70:30 ratio, respectively. Finally, SVM was trained on each training set and its classification accuracy was evaluated by applying the classification learnt model on the corresponding testing set. On each dataset, this experiment was independently run 10 times and the classification accuracy was the average of 10 accuracies.

To identify which k improved the learning task of SVM the most, we compared the performances of SVM using FINNIM-imputed datasets with the original incomplete datasets and the results are shown in Table IX. From it we observe that on the MAL dataset, FINNIM increased the SVM accuracies for all values of k except at $k = 1$, which recorded lower accuracy than the original incomplete dataset. In the case of the IEC10DB dataset, SVM performed better on all value of k than the incomplete dataset. Both datasets reported k = 3 as the best number of neighbor for estimating missing values. It can be said that the increased performance of SVM for majority of k on both of the datasets demonstrates that FINNIM indeed meets the one desirable characteristic mentioned before.

Next, the effectiveness of each imputation method to improve the classification accuracy of SVM was validated using the "before-and-after" experiment where the accuracy of SVM learned on the original incomplete dataset and that learned on the imputed dataset was compared as shown in Table X. We observe that in the case of the MAL dataset, only FINNIM managed to increase the SVM performance, whilst the other

TABLE VII
PEARSON LITZM

	CH_4	C_2H_6	C_2H_4	C_2H_2
MEAN	0.9891	0.9515	0.9853	1.0000
REG	0.9882	0.9976	0.9909	1.0000
EM	**0.9946**	0.9965	0.9962	1.0000
MI	0.8080	0.8287	0.8382	1.0000
FINNIM	0.9896	**0.9986**	**0.9991**	1.0000

TABLE VIII
PEARSON LITZW

	H_2	CH_4	C_2H_6	C_2H_4	C_2H_2
MEAN	0.9913	1.0000	0.9782	0.9978	0.9978
REG	0.9948	1.0000	**0.9908**	0.9525	0.9998
EM	0.9888	1.0000	0.9819	0.9967	0.9810
MI	0.8842	1.0000	0.9464	0.9807	0.9980
FINNIM	**0.9966**	1.0000	0.9727	**0.9984**	**1.0000**

four methods behaved oppositely. For the IEC10DB dataset, only three methods (FINNIM, EM and REG) increased the classification accuracy of SVM compared to the original dataset, with FINNIM scored the highest followed by the EM method and lastly, the REG method.

We analyzed the statistical significance of differences in classification accuracies of SVM on the FINNIM-imputed datasets and on the imputed datasets obtained from the other four methods based on paired t-tests at the 95 percent significance level. The significance is computed for each pair of compared algorithms based on average classification accuracies across the two data sets. The results are presented in Table XI. The results show that on the MAL dataset, FINNIM improves the classification accuracy of SVM significantly compared to the other methods. This indicates FINNIM robustness over higher percentage of missing values as shown on the MAL dataset. For the IEC10DB dataset, FINNIM has significant advantage over the MEAN method in improving SVM classification performance.

E. Analysis

Our proposed method offered better performance in convergence complexity, imputation and classification accuracies than the other four established methods for datasets with artificially inserted missing values as well as for datasets with actual missing values. FINNIM computational complexity was comparable to EM - the other iterative method compared in this study - especially on small datasets of DGA. Convergence iterations differed on individual datasets but all experiments took seconds to complete. Notably, the FINNIM method was robust to different percentages of missing values contained in a dataset. For the REG, MEAN, and MI methods, their efficiency for imputing missing values were not maximized in that they did not efficiently use the information of the instances having missing values. The existence of missing values in an instance limits the use of other observed values of that instance in these well-known imputation methods. In our work, this problem was improved by using the imputed values iteratively for the latter nearest neighbor calculations and imputations.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

TABLE IX
COMPARATIVE PERFORMANCES OF SVM ON THE FINNIM-IMPUTED DATASETS

	Incomplete	FINNIM(k=1)	FINNIM(k=3)	FINNIM(k=5)	FINNIM(k=7)	FINNIM(k=10)
MAL	93.18	92.78	**93.54**	93.48	93.40	93.46
IEC10DB	59.62	61.54	**62.12**	60.39	61.15	61.54

TABLE X
COMPARATIVE PERFORMANCES OF SVM ON DATASETS IMPUTED BY ALL METHODS

	Incomplete	FINNIM	MEAN	REG	EM	MI
MAL	93.18	**93.54**	75.17	65.08	73.51	76.08
IEC10DB	59.62	**62.12**	57.31	59.81	60.38	56.92

TABLE XI
STATISTICAL DIFFERENCE OF PAIRED METHODS USING PAIRED-T TEST

	p-value	
	MAL	IEC10DB
FINNIM-vs-INCOMPLETE	0.07	0.125
FINNIM-vs-MEAN	0.001	0.049
FINNIM-vs-REG	0.001	0.291
FINNIM-vs-EM	0.001	0.500
FINNIM-vs-MI	0.001	0.104

The iterative reuse of imputed data did not propagate errors of imputation as the missing rate increased which made FINNIM registered the best improvement of accuracy for datasets with high missing rates where the NRMSE results of FINNIM were the lowest.

It can be seen that FINNIM preserved the original distribution of a dataset better than the extant approaches. They neither exploited the relationship between features and classes nor assigned imputation hierarchy when imputing missing values in a DGA dataset. FINNIM, on the hand utilized this feature-class relationship - the basis principle of the DGA method - to determine imputation hierarchies for missing values during the imputation ordering phase. Using the imputation order, features that strongly discriminated the classes were imputed earlier than the less relevant ones, a step which made FINNIM preserve correlations to other features better than the existing approaches as shown in the Pearson coefficient results. Also, by selecting nearest neighbors having the same class label with the instance of interest during the imputation estimator phase further enhanced the preserving ability of FINNIM. Our proposed method also increased the performance of a supervised machine learning algorithm, that is SVM. Over various missing percentages and size of datasets, FINNIM-imputed datasets managed to raise the accuracy of SVM higher than the incomplete datasets. Especially, for the dataset with high number of missing values, FINNIM surpassed the other four methods significantly with p less than 0.001 and was marginally significant to the incomplete one with p value(0.07) close to 0.05. For the small dataset, SVM performance was significantly better using FINNIM-imputed dataset than using MEAN-imputed dataset.

Meanwhile, the performances of the other four methods largely depended on the individual datasets. To the best of our knowledge, all the methods have not been well introduced in the domain of power transformer fault diagnosis using DGA method despite the fact that the EM and MI methods are the state of the art imputation methods. For small datasets such as IEC10DB, LITGY, LITZM, and LITZW, the MI method performed the worst both in imputation and classification accuracies. As stated in [18], the MI method relies on large sample for unbiased estimates, therefore, we can conclude that the MI method is not effective for imputing missing values in small DGA datasets. If a research objective is to improve the learning task of a supervised learning algorithm, the EM method can be an alternative candidate for imputing missing values in small DGA datasets, under consideration of its comparable effectiveness to FINNIM in meeting the said objective, especially for SVM. The EM method was comparable to FINNIM in producing accurate estimates in small DGA datasets. Moreover, the EM method was as robust as FINNIM over various percentages of missing values.

From the results, we can safely said that the compounded effects of the three components of FINNIM helps exert relatively more accurate imputation and classification than the four imputation methods. We want to highlight that for small DGA datasets, methods using incomplete and complete values (FINNIM and EM) achieved even better accuracy than the method using only observed values (REG, MEAN, and MI). For various type of datasets, researchers in [5], [6] also demonstrated the better performance of their proposed methods against compared extant approaches. Using estimated values iteratively was one of the key components for their methods. If iterative use of estimated values is a shared concept in FINNIM, [5], and [6], imputation ordering - a component of FINNIM - is not. The contribution of this component in preserving data distribution has been highlighted earlier.

Although FINNIM is built for the DGA datasets, we suggest that for datasets having similar characteristics as the DGA datasets and if the objective is to improve classification accuracy of a learning algorithm, then imputing missing values using FINNIM can be considered as a pre-processing step to improve data quality before learning process takes place. Through data summarization, researchers in [22] presented robust and efficient rules and matched antecedents to diagnose fault in welding dataset. For an instance with missing values, their method ignored those features with missing values, which led to multiple rules and probabilities being activated and estimated, and the class with the highest probability score is assigned to an unknown data point. Instead of ignoring those features with missing values in the welding dataset, the researchers may apply our proposed method to impute the missing values in the experimented dataset before feature selection and rules extraction processes take place. Imputing with our proposed method may reduce the complexity of their

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2014.2350837, IEEE Transactions on Industrial Informatics

approach - less rules and probabilities activation and estimation - and may increase the performance of their method.

Missing data in the form of packet dropouts is one of the problems commonly faced by networked system [23], [24], [25]. Missing data is also an issue faced by discrete-time systems as mentioned in [26], [27]. Majority of these studies, while incorporating the missing data probability during the design stage of their proposed solution, eluded from estimating the missing data. Researchers in [23] applied corrective sampling-based (CS) algorithm which required no estimation of the missing packets. However, the maximum number of missing data tolerated by their CS depends on the value of sketch length - one of the CS design parameters. This constraint may limit the efficiency of their proposed method because it is difficult to estimate the number of missing values in any dataset beforehand. This possible drawback can be avoided by estimating the missing packets using our proposed method as a pre-processing step to their CS algorithm. To solve H_∞ filtering problem for discrete-time systems, [26] proposed a measurement that compensated the negative influence of missing data by representing the missing data as a stochastic variable that satisfied the Bernoulli binary distribution. Instead of assuming the probability of the missing data occurrence, [26] can apply the FINNIM algorithm to estimate the missing data as alternative approach to handling missing data in discrete-time systems.

IV. Conclusion

This paper has presented an efficient imputation method named FINNIM that estimates missing values in DGA datasets. Experiment results have demonstrated that FINNIM outperforms EM, MI, REG and MEAN, in terms of imputation accuracy, the classification accuracy, and the convergence criteria. In particular, FINNIM lives up to one desirable characteristic for an imputation method that is the missing data estimation improves the classification performance of a learning algorithm, that is SVM. In future, we aim to replace kNN algorithm with SVR as the estimator in the second component of FINNIM. First, set a feature with missing values as the output attribute and the other features as the input attributes. Then, predict all missing values in the output attribute using SVR. Repeat these two steps for the other features. The imputation ordering will consist of only I, which determine the sequence for selecting a feature as the output attribute. The whole process is iteratively executed according to steps in Section II-C.

References

[1] Z. Yang, W. H. Tang, A. Shintemirov, and Q. Wu, "Association rule mining-based dissolved gas analysis for fault diagnosis of power transformers," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 39, pp. 597–610, Nov 2009.

[2] Q. Song, M. Shepperd, X. Chen, and J. Liu, "Can k-nn imputation improve the performance of c4. 5 with small software project data sets? a comparative evaluation," *Journal of Systems and Software*, vol. 81, no. 12, pp. 2361–2370, 2008.

[3] L. Himmelspach and S. Conrad, "Clustering approaches for data with missing values: Comparison and evaluation," in *Digital Information Management (ICDIM), 2010 Fifth International Conference on*, pp. 19–28, July 2010.

[4] Z. Yong-li, W. fang, and G. Lan-qin, "Transformer fault diagnosis based on naive bayesian classifier and svr," in *TENCON 2006. 2006 IEEE Region 10 Conference*, pp. 1–4, Nov 2006.

[5] S. Zhang, Z. Jin, and X. Zhu, "Missing data imputation by utilizing information within incomplete instances," *Journal of Systems and Software*, vol. 84, no. 3, pp. 452 – 459, 2011.

[6] C. Zhang, X. Zhu, J. Zhang, Y. Qin, and S. Zhang, "Gbkii: An imputation method for missing values," in *Advances in Knowledge Discovery and Data Mining*, pp. 1080–1087, Springer, 2007.

[7] P. J. García-Laencina, J.-L. Sancho-Gómez, A. R. Figueiras-Vidal, and M. Verleysen, "K nearest neighbours with mutual information for simultaneous classification and missing data imputation," *Neurocomputing*, vol. 72, no. 7, pp. 1483–1493, 2009.

[8] S. Zhang, X. Wu, and M. Zhu, "Efficient missing data imputation for supervised learning," in *Cognitive Informatics (ICCI), 2010 9th IEEE International Conference on*, pp. 672–679, July 2010.

[9] C. Conversano and R. Siciliano, "Incremental tree-based missing data imputation with lexicographic ordering," *Journal of classification*, vol. 26, no. 3, pp. 361–379, 2009.

[10] E. R. Hruschka Jr, E. R. Hruschka, and N. F. Ebecken, "Bayesian networks for imputation in classification problems," *Journal of Intelligent Information Systems*, vol. 29, no. 3, pp. 231–252, 2007.

[11] M. Di Zio, M. Scanu, L. Coppola, O. Luzi, and A. Ponti, "Bayesian networks for imputation," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 167, no. 2, pp. 309–322, 2004.

[12] E. Acuna and C. Rodriguez, "The treatment of missing values and its effect on classifier accuracy," in *Classification, Clustering, and Data Mining Applications*, pp. 639–647, Springer, 2004.

[13] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman, "Missing value estimation methods for dna microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.

[14] S. Zhang, "Shell-neighbor method and its application in missing data imputation," *Applied Intelligence*, vol. 35, no. 1, pp. 123–133, 2011.

[15] "Fault diagnosis of power transformer based on association rules gained by rough set," in *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on*, vol. 3, pp. 123–126, Feb 2010.

[16] X.-Z. Wang, M.-Z. Lu, and J.-B. Huo, "Fault diagnosis of power transformer based on large margin learning classifier," in *Machine Learning and Cybernetics, 2006 International Conference on*, pp. 2886–2891, Aug 2006.

[17] L. Ganyun, C. Haozhong, Z. Haibao, and D. Lixin, "Fault diagnosis of power transformer based on multi-layer svm classifier," *Electric Power Systems Research*, vol. 74, no. 1, pp. 1–7, 2005.

[18] J. L. Schafer and J. W. Graham, "Missing data: our view of the state of the art.," *Psychological methods*, vol. 7, no. 2, p. 147, 2002.

[19] D. J. Stekhoven and P. Bhlmann, "Missforestnon-parametric missing value imputation for mixed-type data," *Bioinformatics*, vol. 28, no. 1, pp. 112–118, 2012.

[20] K. Bacha, S. Souahlia, and M. Gossa, "Power transformer fault diagnosis based on dissolved gas analysis by support vector machine," *Electric Power Systems Research*, vol. 83, no. 1, pp. 73–79, 2012.

[21] Z. biao Shi and Y. Li, "Fault diagnosis of power transformer using ls-svms with bcc," in *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pp. 417–420, Aug 2009.

[22] R. Gong, S. H. Huang, and T. Chen, "Robust and efficient rule extraction through data summarization and its application in welding fault diagnosis," *Industrial Informatics, IEEE Transactions on*, vol. 4, no. 3, pp. 198–206, 2008.

[23] S. Das and T. Singh Sidhu, "Application of compressive sampling in synchrophasor data communication in wams," *Industrial Informatics, IEEE Transactions on*, vol. 10, pp. 450–460, Feb 2014.

[24] X. He, Z. Wang, Y. Liu, and D. Zhou, "Least-squares fault detection and diagnosis for networked sensing systems using a direct state estimation approach," *Industrial Informatics, IEEE Transactions on*, vol. 9, pp. 1670–1679, Aug 2013.

[25] H. Zhang, Y. Shi, and A. Mehr, "Robust static output feedback control and remote pid design for networked motor systems," *Industrial Electronics, IEEE Transactions on*, vol. 58, pp. 5396–5405, Dec 2011.

[26] P. Shi, X. Luan, and C.-L. Liu, "H_∞ filtering for discrete-time systems with stochastic incomplete measurement and mixed delays," *Industrial Electronics, IEEE Transactions on*, vol. 59, pp. 2732–2739, June 2012.

[27] H. Zhang, Q. Chen, H. Yan, and J. Liu, "Robust h_∞ filtering for switched stochastic system with missing measurements," *Signal Processing, IEEE Transactions on*, vol. 57, pp. 3466–3474, Sept 2009.