IEICE TRANS. FUNDAMENTALS, VOL.E97–A, NO.9 SEPTEMBER 2014

PAPER

Traffic Pattern Based Data Recovery Scheme for Cyber-Physical Systems

Naushin NOWER[†a)], ***Nonmember***, **Yasuo TAN**[†], ***and*** **Azman Osman LIM**[†], ***Members***

SUMMARY Feedback data loss can severely degrade overall system performance. In addition, it can affect the control and computation of the Cyber-physical Systems (CPS). CPS hold enormous potential for a wide range of emerging applications that include different data traffic patterns. These data traffic patterns have wide varieties of diversities. To recover various traffic patterns we need to know the nature of their underlying property. In this paper, we propose a data recovery framework for different traffic patterns of CPS, which comprises data pre-processing step. In the proposed framework, we designed a Data Pattern Analyzer to classify the different patterns and built a model based on the pattern as a data pre-processing step. Inside the framework, we propose a data recovery scheme, called Efficient Temporal and Spatial Data Recovery (ETSDR) algorithm to recover the incomplete feedback for CPS to maintain real time control. In this algorithm, we utilize the temporal model based on the traffic pattern and consider the spatial correlation of the nearest neighbor sensors. Numerical results reveal that the proposed ETSDR outperforms both the weighted prediction (WP) and the exponentially weighted moving average (EWMA) algorithms regardless of the increment percentage of missing data in terms of the root mean square error, the mean absolute error, and the integral of absolute error.

key words: *data pattern analyzer, stochastic traffic pattern, data recovery, cyber-physical systems*

1. Introduction

Cyber-physical systems (CPS) are a new generation of communication, control and computation that have received a great deal of attention recently [1]. CPS enable the virtual world to interact with the physical world in order to monitor and control the intended parameter in real-time basis. In CPS, technologies such as communication, control, computation, cognition and sensing converge to create new technologies for a smarter society. The area of CPS represents the intersection of several system trends, such as real-time embedded systems, distributed systems, control systems and networked wireless systems.

To facilitate communications between the cyber and the physical world, wireless sensor and actuator network (WSAN) is an essential component of CPS. This is because, the traditional wireless sensor network (WSN) is limited in its ability to monitor the physical world [2]. However, CPS achieves this requirement by facilitating the system to sense, interact and change the physical world in real-time by using feedback control loop. In a typical CPS application, sensor nodes collect information from the physical world as a source of CPS input. Upon receiving the input information, a controller makes a corresponding decision by computing and actuators perform a corresponding action in the physical world through the closed-loop feedback. Thus, the proper timing and accuracy of feedback data is very important for interaction between the cyber and the physical world. Figure 1 shows the general control view of a cyber-physical system.

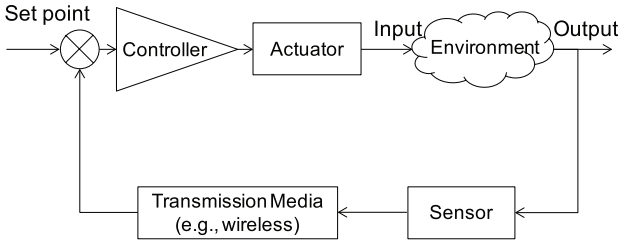


Fig. 1 General control view of CPS.

Since the CPS exploit the physical information collected by WSANs, it also inherit the wireless contention problem of WSAN. This is a challenging issue for control in real-time. Wireless channels have many adverse properties like path loss, fading, adjacent channel interference, node/link failure, etc. Besides these, wireless signal can be easily affected by noise, physical obstacles, node movement, environmental change and so on [3]–[5]. Because of this unpredictable and dynamic nature, sensing data loss is a common phenomenon, which makes hamper in controlling decision. In particular, for time critical applications, feedback data must have to arrive on time, to make decision. In many cases, re-transmission cannot provide appropriate solution because of the unpredictable network behavior, which can cause high delay.

Moreover, the applicability of CPS is found in numerous time-critical applications including smart house to smart grid. Emerging applications of CPS include, medical devices and systems, aerospace systems, transportation vehicles and intelligent highways, defense systems, robotic systems, process control, factory automation, building and environmental control, smart spaces, intelligent home and so on [6]. These systems are equipped with a large network of sensors distributed across different components, which leads to a huge amount of measured data available to the system controller. For example, CPS can be used in the medical health care applications, where various types of sensors

Manuscript received January 9, 2014.
Manuscript revised May 20, 2014.

[†]The authors are with the School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), Nomi-shi, 923-1211 Japan.
a) E-mail: naushin@jaist.ac.jp

DOI: 10.1587/transfun.E97.A.1926

are used to monitor patient's condition and then controller communicates with doctor using the feedback closed control loop system. Thus, the doctor can remotely monitor the patient's physical condition, give suggestions or prescriptions and also do remotely guided robotic microsurgery. Moreover, CPS is planning to use in more complex situation, in particular robot-assisted MRI guided interventions on aortic valve implantation, cardiac surgery, etc. [7]. In these time-critical applications, the accuracy and real-time presence of feedback data is very essential for the controller to make a real-time and highly reliable decision. Since these measurements are collected continuously along the time, they can be regarded as a time series data. In this wide spectrum of CPS applications, different data properties are observed. These time series data also have different traffic patterns in terms of their shape, trend, variation and periodicity. Some series maintain stable stage, some show stochastic behavior and others exhibit repetition in their evolutions. Because of this time series data heterogeneity, one single data recovery algorithm can not provide a solution for all.

To maintain uninterrupted control, we always need to ensure the continuous presence of feedback time series data. Therefore, to handle any uncertainty we need to know the behavior or trend of the data. For time series data, we need a general tool that can analyze and determine the pattern from the data. Thus, it is important to build an effective traffic pattern analyzer to analyse the data, so that we can better understand the underlying properties of collected time series data that control the system operation. Based on the data properties, it is easier to design an effective data recovery algorithm to provide uninterrupted control. Thus, successful determination of traffic pattern ensure efficient data recovery to maintain continuous control. In this paper, we proposed a data recovery framework for the CPS applications. Inside the proposed framework, we design a Data Traffic Pattern Analyzer (DTPA) to deal with different time series data for CPS. The designed analyzer determines whether the time series data exhibits deterministic or stochastic property or have repeated patterns based on the property of the collected data. Based on the pattern, we design a temporal model construction step and propose a model based data recovery algorithm for real-time recovery. That is, from the known nature of deterministic, stochastic and repeated pattern, the analyzer will evaluate the collected data. Whenever, the data series has stochastic or repeated patterns, we build a temporal model and use that model to recover the data in the cased of missing data. The stochastic data series is normally highly auto-correlated and outliers have a different correlation structure than the deterministic data series. Auto Regressive Integrated Moving Average (ARIMA) [8] model is a very powerful model to identify the auto-correlated nature or trend of stochastic data. We identify the stochastic and repeated trend of data using the analyzer and then build a temporal model to recover the data. For data recovery, we propose an Efficient Temporal and Spatial Data Recovery (ETSDR) scheme for stochastic data and repeated traffic of CPS. For deterministic pattern, we incorporate our spatial correlation based data recovery scheme in [9].

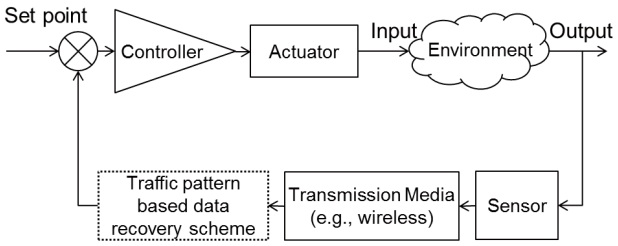


Fig. 2 Proposed data recovery scheme for control view of CPS.

In [9], we proposed a highly Efficient Spatial Data Recovery (ESDR) scheme that deals with deterministic traffic pattern of CPS. This scheme is very efficient for deterministic traffic pattern like temperature, humidity, moisture which is linearly correlated with space. Thus, in our proposed ESDR scheme, we utilized spatial correlation of neighboring sensors by using the Pearson correlation coefficient (PCC). In this paper, we present a data analyzer to categorize different traffic patterns. For stochastic and repeated traffic pattern, we design a data model based on temporal correlation and determine the spatial effect by considering linear or non linear relation exist between the neighbors. The designed model is used to recover data in real-time applications. Using the analyzer we extract the data property and build a model, as a data pre-processing step. Since, deterministic data seldom varies we can utilize our ESDR [9] whenever the analyzer classify the data as a deterministic pattern.

Our first contribution is that, the analyzer can successfully classify the traffic patterns of CPS. Second, the patterns of the time series data determined by the analyzer can be used to build a temporal model. This data model is used to estimate the missing data for uninterrupted control. Third, the data model is determined in the off line analysis stage step, so it can ensure timely data recovery because of minimum computation in real-time. Figure 2 shows the control view of CPS with proposed traffic pattern based data recovery scheme.

The rest of the paper is organized as follows. Section 2 summarizes some state-of-the-art research works that is related to this paper. In Sect. 3, the proposed traffic pattern based data recovery scheme is presented. We describe the simulation scenario and the evaluation parameters in Sect. 4. Simulation results and discussions are presented in Sect. 5. Section 6 concludes with conclusion and future works.

2. Research Background

2.1 Background

CPS are deployed as a sensor and actuator network of interacting elements with a huge amount of data with different patterns. Based on the observed data from different CPS applications, we classify three traffic patterns: deterministic, stochastic and repeated. Theoretically, the pattern that

remains constant with time is known as deterministic pattern. The deterministic traffic pattern always maintains a stable state. For example, in a temperature controlled room, the temperature remains stable or varies to a certain degree within a given range.

On the other hand, any traffic pattern which involves random change and indeterminacy is defined as a stochastic traffic pattern. Unlike a deterministic pattern, a stochastic pattern involves randomness in their nature. If this stochastic data have time depended moments then, it is called stationary stochastic data pattern. That is, the mean and variance of the stationary data remained constant over a fixed time. On the other hand, the mean and variance of non-stationary data changes over time. Stochastic time series data can be presented as a combination of autoregressive (AR), moving average (MA), or seasonal dynamic components.

The time series data, that involves repetition, is known as a repeated/periodic pattern. These traffic patterns can be transmitted by four different traffic types [10]: Streamline traffic of variable rate or fixed rate, periodic, bursty and arbitrary rate. In the periodic traffic types, data is transmitted periodically. In bursty data traffic, the data tends to a bursty nature, from low transmission to sudden increase the rate of transmission. In arbitrary rate, there is no fixed schedule for data transmission. In this research, we design our data recovery scheme for periodic data traffic since, CPS utilize periodic traffic types for control.

2.2 Related Work and Motivation

Data recovery is a part of most research and there exist several methods to handle this. Even-though, there exist several methods, the recovery of data loss for CPS still poses an open problem because of its unique requirement. The whole recovery process for CPS must be held in real-time and invisible to the outside world. In this section, we first focus on the existing data analyzer for CPS. Then, we discuss in details the data recovery procedures for CPS.

In [11], the authors proposed a data analysis technique to extract meaningful information from the large volume of noisy data. Their designed analyzer named Tru-Alarm, is used to recognize trustworthy alarms from the noisy and false alarms. Tru-Alarm estimates the locations of objects causing alarms, then constructs an object-alarm graph and carries out trustworthiness inference based on the graph links. Their study also reveal that the alarm trustworthiness and sensor reliability could be mutually enhanced. This property is used to filter out the alarms generated by unreliable sensors. Moreover, in [12], the authors proposed a method called IntruMine to detect and verify intruders from the untrustworthy data by modeling the relationships between sensor and intruders. The authors discovered the trajectories of intruders from the untrustworthy data by constructing watching network in [13].

In [14] authors discussed about retrieving the atypical events from massive sensor data and analyzing them with spatial, temporal, and other multidimensional information. Whenever a abnormal event happens such as a congestion is detected in traffic system, the sensor will send out as atypical records. They fixed a threshold for normal event and based on that a atypical event is detected and cluster is formed. The basic cluster is designed to summarize an individual event, and the macro-cluster is used to integrate the information from multiple events. The atypical cluster is then used to effective query execution. Each of the existing analyzer is designed for different purposes and objectives. None of this can be used for data traffic pattern analysis for data recovery.

Xia, et al. [15] first proposed a solution for CPS over WSANs to cope with packet loss. They illustrate three prediction algorithms and provide a comparison between them. First algorithm based on the assumption that, the state of the physical system does not change during the last sampling period. So, previous sample is used to replace the missing value. The second algorithm computes a moving average of the previous m samples to restore the lost data. Thus it treats every previous measurement equally. In third algorithm, which is known as weighted prediction (WP), weighted average of all previous samples is taken to replace the missing one. Simulation result shows that third algorithm works well compared with others. All of their procedures are bound for specific situation where current data depends on the previous data or the combination of previous data but not for all conditions.

Choi, et al. [16] exploit an exponentially weighted moving average (EWMA) based value estimation algorithm to reduce the impact of packet loss. When some packets are randomly dropped in wireless network environment, the EWMA algorithm filters an abrupt increase or decrease by exponentially smoothing commands or data based on the past value profile. This method is only suitable, when the data series is an exponentially weighted combination of past data sets. But in real-life there is no guarantee that data will always maintain this combination. Moreover, none of the existing data recovery scheme includes model identification before recover the data. We believe that successful identification of data model can ensure accurate and timely recovery.

In the literature, model based data aggregation scheme exists. In [17], authors proposed an ARIMA based data aggregation method to reduce the energy consumption and number of communication. In this scheme, both the sensor node and the aggregator have the same model for data generation. Sensor node checks whether the data predicted from the model and measured data is same. If the real value and predicted value is within the threshold, then the sensor node will not transmit the data to the aggregator. Otherwise, the sensor will send new data to the aggregator.

The applications of CPS are numerous which involve different data patterns. In the existing literature, there is no direction of data recovery based on data traffic pattern. Thus, the recovery process without considering the nature can not provide a solution for all. To recover data accurately,

we first need to understand the nature of the data and their spatial relationship with others. To achieve our motivation, we propose a data pre-processing stage, where the data analyzer is used to classify the data pattern and based on that property, a model is built for real time recovery process.

3. Proposed Traffic Pattern Based Data Recovery Scheme

In this section, we propose data recovery framework for time series data. We design a data pattern analyzer to classify the traffic pattern for model generation as a data pre-processing step. Before doing this, we identified the basic properties of each traffic pattern. The designed data recovery framework contains two steps: i) Off line data pattern analysis and temporal model construction ii) On line recovery of data. Figure 3 shows the block diagram of our proposed traffic pattern based data recovery scheme.

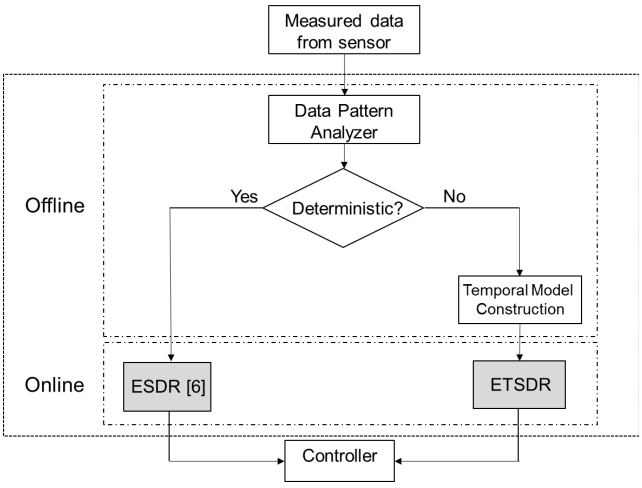


Fig. 3 Block diagram framework of traffic pattern based data recovery scheme.

3.1 Offline Data Pattern Analyzer

The aim of this step is to classify the data using the analyzer, and build a model based on the property present in the data. According to the classifications, we include three pattern checkers in the data analyzer: deterministic pattern checker, stochastic pattern checker and repeated pattern checker. The block diagram of proposed data analyzer is shown in Fig. 4. The following assumptions have been considered. First, n observed sensor data is available for analysis and model generation. Second, the group of time series data for a applications follow a specific data property. Third, the maximum tolerable variation (α) of two consecutive deterministic data pattern is known. In addition, some real-life known repeated patterns are available for matching. Forth, the maximum number of attempts (C) to generate the model is fixed at initialization stage. The parameter C is also used to make the decision that, the model cannot be generated from the available data.

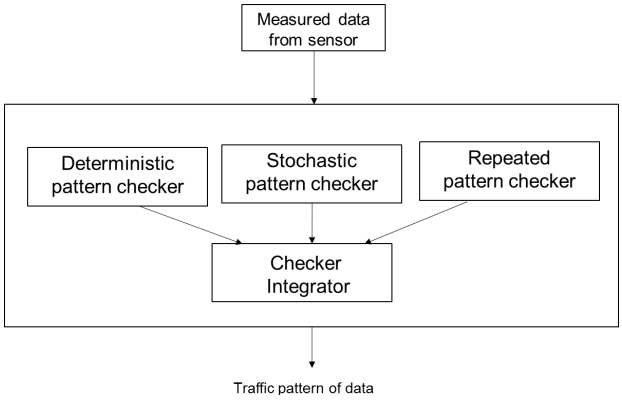


Fig. 4 Block diagram of data pattern analyzer.

Each pattern checker identifies the time series data based on the property and calculates the percent of the data maintains that property. The checker integrator combines the result from the all checkers and makes decision.

Deterministic pattern checker focuses on detecting the stable behavior in time series data. The main feature of deterministic pattern is that, they have almost constant value or have a very small variation. Thus, checker identifies whether the consecutive measurement (d_{s1}, d_{s2}) of time series data is less then α, where α is the maximum tolerable difference between two consecutive measurements.

The stochastic pattern checker is designed to detect the stochastic behavior in time series data. The stochastic data series can be further categorized into two types: stationary and non-stationary. Stationarity, is defined as a quality of a time series data in which the statistical parameters (mean and variance) of the series do not change with time. The stationarity of time series data can be determined by examining the auto-correlation coefficient function (ACF) and partial correlation coefficient function (PACF). The ACF is a set of correlation coefficients between the series and lags of itself over time [18]. The k-order auto-correlation coefficient of a data series $d_{s1}, d_{s2}, \ldots, d_{sn}$ of sensor s is defined as

$$r_k = \frac{\sum_{i=1}^{n-k} (d_{si} - \bar{d_{si}})(d_{s(i+k)} - \bar{d_{si}})}{\sum_{i=1}^{n} (d_{si} - \bar{d_{si}})^2} \tag{1}$$

where, r_k is the k-lag sample auto-correlation and $\bar{d_{si}}$ is the average of n observations. The PACF is the partial correlation coefficients between the series and lags of itself over time. The k-order partial auto-correlation coefficient of a data series is defined as

$$\phi_{11} = r_1 \tag{2}$$

$$\phi_{22} = (r_2 - {r_1}^2)(1 - {r_1}^2) \tag{3}$$

$$\phi_{kj} = \phi_{(k-1)j} - \phi_{kk}\phi_{(k-1)(k-j)} \tag{4}$$

$$\phi_{kk} = r_k - \sum_{j=1}^{k-1} \phi_{(k-1)} r_{k-j} \Big/ 1 - \sum_{j=1}^{k-1} \phi_{(k-1)} r_j \tag{5}$$

For the stationary time series, the ACF and PACF trend

to zero gradually (die out). On the other hand, for non-stationary data series, the value of ACF and PACF remain for a long time. The analyzer uses this property to determine the type of stochastic data.

The repeated pattern checker compares the time series data with the stored real life periodic pattern. In the case of successful matching with any known pattern, we specify the number of distinguish patterns, their duration, interval and other properties. From the observed properties, we build a model and verify the model by comparing the model generated data and with the real data.

3.2 Temporal Model Construction

For the stochastic data pattern, we deploy a temporal model construction step. We assume that, the error offset, the maximum difference between the model computed data and the measured data is fixed at initilization step for verifications. We analyze the stochastic data series trend by modelling it into ARIMA series. The Autoregressive Integrated Moving Average (ARIMA) models, or Box-Jenkins methodology, are a class of linear models that is capable of representing stationary as well as non-stationary time series.

ARIMA model is a very powerful tool that uses historical data to predict future data values. Any type of stochastic data series can be identified by this model [18]. The ARIMA model, also called Box-Jenkins model, can be divided into three components: auto-regressive (AR), moving-average (MA), and one-step differencing. The AR component estimates the current sample as a linear-weighted sum of previous samples; the MA component captures relationship between prediction errors; and the one-step differencing component captures relationship between adjacent samples. In ARIMA, the AR component captures the temporal correlation in the time series by modeling a future value as a function of a number of past values. The MA component is modeled as a zero-mean, uncorrelated Gaussian random variable [20].

3.2.1 Auto-Regressive Model of Order p

An auto-regressive (AR) model is a simplified version of ARIMA model which describes random time-varying process. The AR model specifies that the output variable depends linearly on its own previous values [8]. The AR model of sensor s data series $d_{s1}, d_{s2}, \ldots, d_{sn}$ with order p is defined as follows

$$d_{sn} = c + \sum_{i=1}^{p} \varphi_i d_{s(n-1)} + \varepsilon_n \tag{6}$$

where p is the order of auto-regressive terms, $\varphi_1, \varphi_2, \ldots \varphi_p$ are the parameter of the model, c is a constant and ε_n is white noise. This can be equivalently written using the back-shift operator B as

$$d_{sn} = c + \sum_{i=1}^{p} \varphi_i B^i d_{sn} + \varepsilon_n \tag{7}$$

3.2.2 Moving Average Model of Order q

A moving-average (MA) model is a linear regression of the current and previous error of a random series. The MA model of sensor s data series $d_{s1}, d_{s2}, \ldots, d_{sn}$ with order q is defined as follows

$$d_{sn} = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{n-1} \tag{8}$$

where q is the number of moving average terms, μ is the mean of the series, $\theta_1, \theta_2, \ldots \theta_q$ are the parameter of the series, and ε_n is the error. This can be written using back shift operator B as

$$d_{sn} = \mu + \sum_{i=1}^{q} \theta_i B^i \varepsilon_n \tag{9}$$

3.2.3 ARIMA Model

An ARIMA model predicts future values of a sensor s data series by a linear combination of its auto-regressive past values, integrated, and moving average of errors. The model is generally referred to as an ARIMA(p, d, q) model where parameters p, d, and q are non-negative integers that refer to the order of the auto-regressive, the amount of differencing, and moving average parts of the model respectively. ARIMA is used for non-stationary data time series modelling. If any of p, d, or q are zero, the corresponding letters are often dropped. For example, if p and d are zero, then model would be denoted MA(q).

$$\theta_p(B) \triangle^d d_{s(t)} = \Theta_q(B) \varepsilon_n \tag{10}$$

where B is the backward shift operator, $\triangle$is the backward difference, d is the order of differencing and θ_p and Θ_q are the polynomial of order p and q respectively. In addition, $Bd_{sn} = d_{s(n-1)}$ and $\triangle = 1 - B$. ARIMA(p, d, q) model is the product of an AR part AR(p):

$$\theta_p = 1 - \varphi_1 B - \varphi_2 B^2 - \ldots - \varphi_p B^p, \tag{11}$$

an integrating part:

$$I(d) = \triangle^{-d} \tag{12}$$

and a MA part MA(q):

$$\Theta_q = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q. \tag{13}$$

The flowchart for temporal model identification for stochastic data is shown is Fig. 5.

Step 1: Temporal Model Identification

The aim of this step is to determine whether the series is stationary or not. If the series is not stationary, it is converted to a stationary series by differencing: the original series is replaced by a series of differences and an ARMA model is then specified for the differenced series. Differencing can help stabilize the mean of a time series by removing changes

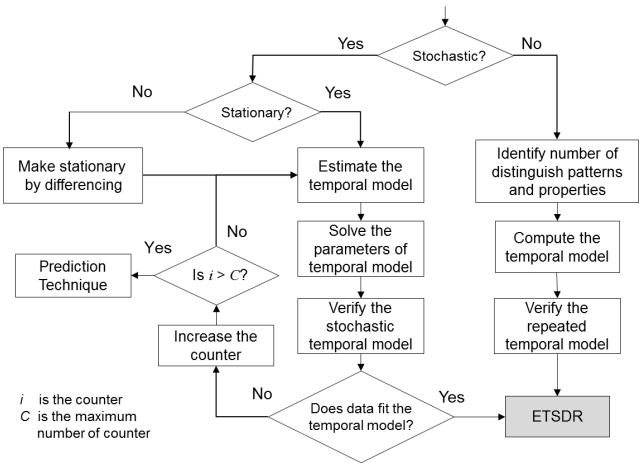


Fig. 5 Flowchart of temporal model construction.

Table 1 Properties of ACF and PACF.

	ACF	PACF
AR(p)	Tails off (trend to zero gradually)	Cuts off after lag p
MA(q)	Cuts off after lag q (disappear or zero)	Tails off
ARMA(p,q)	Tails off after lag ($q-p$)	Tails off after lag ($q-p$)

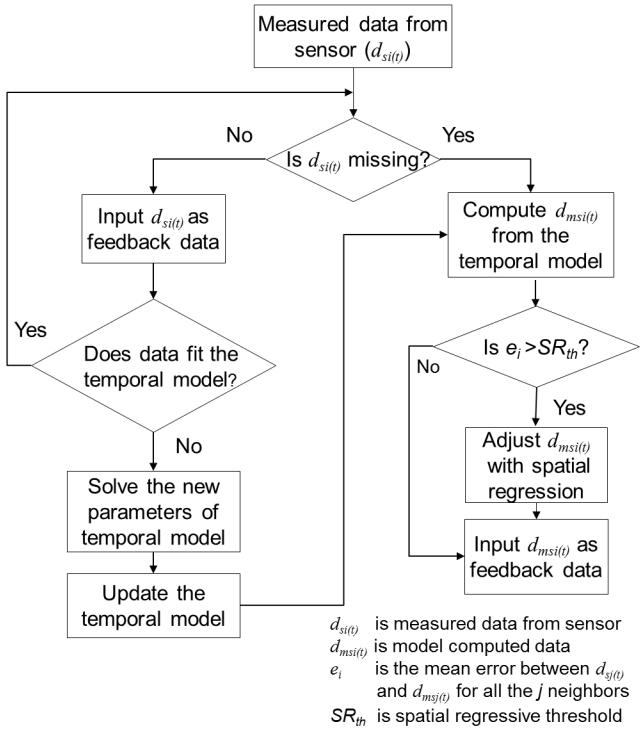


Fig. 6 Flowchart of ETSDR scheme.

in the level of a time series, and thus eliminates trend and seasonality. The differenced series is the change between consecutive observations in the original series, and can be written as $d^1_{st} = d_{st} - d_{s(t-1)}$. Whenever the differenced data is not stationary yet then, it is necessary to difference the data in second times to obtain a stationary series: $d^2_{st} = (d_{st} - d_{s(t-1)}) - (d_{s(t-1)} - d_{s(t-2)})$. In practice, it is almost never necessary to go beyond second-order differences.

Step 2: Estimate the Temporal Model

In this step, we determined the order of p and q from the observed series and determine the model by comparing the sample ACF and PACF with the theoretical pattern of known model. The properties of ACF and PACF for AR(p), MA(q) and ARMA(p, q) is listed in Table 1. From the ACF and PACF, the ARMA model that closely fit to the data can be identified.

Step 3: Solve the Parameters of Temporal Model

In this step, we calculate the parameters of the identified model using method of moments and Yule-Walker equations [19].

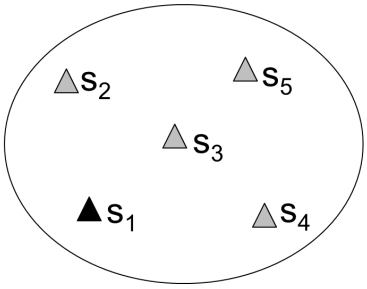
Step 4: Verify the Temporal Model

To verify the model, we compare the model generated data with the e observed sensor data. If the verification fails, we continue to estimate the model until the maximum counter C is reached. In the case of successful verifications, we use that model to generate the data.

3.3 Proposed ETSDR Scheme

To deploy our proposed stochastic data recovery scheme, we propose a flowchart of ETSDR scheme as depicted in Fig. 6. Here, the proposed ETSDR scheme will compute the model estimated data when there is an input measured data from the sensors. If there is no missing data, then the measured data is used as a feedback data. At the same time the difference between the measured data and model computed data is computed for model verifications. If the verifications fails, model is updated by computing new parameters.

On the other hand, when there is a missing data, the model estimated data is utilized. At the same time, neighbor's model estimated data and neighbor's measured data is compared. Whenever the difference between two data crosses the spatial regressive threshold (SR_{th}), the spatial regression is considered. SR_{th} is the maximum tolerable error value as a threshold indicator to determine the spatial regression to be applied or not in the ETSDR scheme. At the initialization step, SR_{th} is a predefined constant value in order to cope with the dynamic environmental changes (i.e., the disturbance effects). Since the temporal model is based only on the property of data series itself, but in real life, the sensor measurement can be effected by the surrounding environment factors. In the case of a missing data of a sensor, we utilize the temporal model to estimate the model computed data and at the same time we check all the one-hop neighbors' measurements to determine whether we should consider the spatial regression or not. To handle the spatial regression, we compare the neighbor's measured data and the neighbor's model computed data. This difference value is defined as model generated error. In this paper, we define that e_i is the average error between all the one-hop neighbors' sensor of the measured data and the model computed



Time	s_1		s_2		s_3		s_4		s_5	
t-1	$d_{s1(t-1)}$	$d_{ms1(t-1)}$	$d_{s2(t-1)}$	$d_{ms2(t-1)}$	$d_{s3(t-1)}$	$d_{ms3(t-1)}$	$d_{s4(t-1)}$	$d_{ms4(t-1)}$	$d_{s5(t-1)}$	$d_{ms5(t-1)}$
t	?	$d_{ms1(t)}$	$d_{s2(t)}$	$d_{ms2(t)}$	$d_{s3(t)}$	$d_{ms3(t)}$	$d_{s4(t)}$	$d_{ms4(t)}$	$d_{s5(t)}$	$d_{ms5(t)}$
...	...	...	...	...	...	...	...	...	...	...

Fig. 7 An example of 5-sensor scenario.

data. If this e_i is greater the SR_{th}, the spatial regression is added to the model computed data. Otherwise, the only the model computed data is used as a feedback data.

As far as we are concerned, most of the spatial correlation regression measures the linear correlation between the nearest neighbors. If an environment is highly correlated in space, then the spatial information can be used to estimate missing data and the estimation function can achieve a high accuracy. Pearson Correlation Coefficient (PCC) is a common measure of the linear correlation between two random variables i and j. It reflects the degree of association between two variables. Therefore, the coefficient correlation degree of PCC (ρ_{ij}) in between two random variables i and j in specified window size (W) can be computed as follows

$$\rho_{i,j} = \frac{\sum_{w=1}^{W} (i(w)-\bar{i})(j(w)-\bar{j})}{\sqrt{(\sum_{w=1}^{W} (i(w)-\bar{i})^2)} \times \sqrt{(\sum_{w=1}^{W} (j(w)-\bar{j})^2)}} \quad (14)$$

But in real-life environment, the neighbor sensors can be correlated non-linearly with their neighbors also. We consider this phenomenon and calculate the spatial regression based on the applications.

Suppose five sensors are placed randomly with one-hop neighbor to each other. They are denoted as s_1, s_2, s_3, s_4, and s_5 as illustrated in Fig. 7. We assume that the measured data of s1 ($d_{s1(t)}$) is lost at time t. In the offline steps of data pattern analyzer and temporal model construction, we construct the model for each sensor which is denoted as m_{s1}, m_{s2}, m_{s3}, m_{s4}, m_{s5} and their model computed data as $d_{ms1(t)}$, $d_{ms2(t)}$, $d_{ms3(t)}$, $d_{ms4(t)}$, and $d_{ms5(t)}$, respectively at the time t.

To perform the pseudocode of the ETSDR scheme, we first set the spatial regressive threshold (SR_{th}) is 1.5 at the initialization stage. Suppose the measured data from the sensor 1 (s_1) at t time is missing, i.e., $d_{s1(t)}$. The temporal model is utilized to compute the model computed data ($d_{ms1(t)}$) of s_1. At the same time, the temporal model will also compute and also compare the measured data of other sensors. Then, the average error, e_1 is computed and compare with the SR_{th}. If the average error is more than the SR_{th}, the spatial regression is taken into account. Otherwise the model computed data only being used as a feedback data.

Algorithm: Efficient Temporal and Spatial Data Recovery

```
if d_si(t) = available then
  for each d_si(t) from the sensor s_i do
    Compute d_msi(t) from the temporal model
      if abs(d_si(t) − d_msi(t)) > error offset then
        Update the model by calculating new parameters
      end if
  end for
else
  for all one-hop neighbors, j of sensor s_i do
    if avg(abs(d_sj(t) − d_msj(t))) > SR_th then
      d_e(t) ⟵ d_si(t) = d_msi(t) + spatial regression
    else
      d_e(t) ⟵ d_si(t) = d_msi(t)
    end if
  end for
end for
```

Fig. 8 Pseudocode for efficient temporal and spatial data recovery algorithm.

Figure 8 describes the proposed ETSDR algorithm, which is used to produce an estimated data from time to time.

4. Numerical Simulations

In this section, we conduct the simulation studies to evaluate our proposed ETSDR scheme compared to the WP algorithm [15] and the EWMA algorithm [16]. Before doing this, we determine the data traffic pattern using the proposed analyzer. We create a small scenario for simulation that can resemble to smart grid applications for energy consumption control in smart community. We assume a community with five houses, where each sensor (e.g., smart meter) in a house measures the energy consumption and communications with the controller that placed in a cloud for computing the energy demand and supply in real-time manner. The energy value that produces by this smart meter is stochastic and depends on its usage profile of consumer on the home appliances in a house. Moreover, the value of created energies (e.g., solar panel, fuel cell, or electric vehicle, wind energy, etc) from different houses may or may not linearly correlate with other houses as a spatial correlation. In this paper, we consider this kind of scenario for our simulation. In our simulation environment, five sensors and one controller are considered. We generate random (stochastic) data series using MATLAB simulator and assign it to the five sensors. We assume that the distance between the sensors is non-linearly. Moreover, to make the scenario more realistic we add some disturbance effects at the certain period of time. Then, we determine the pattern of the generated data by using the analyzer. The analyzer identify the series as a stationary scholastic data since, variance is stable. In the next step, we construct the temporal model from the generated data by observing the ACF and PACF. We identify possible value of p and q and find $p = 2$ and $q = 0$ for sensor 1. Then, we solve the parameters using Yule-Walker

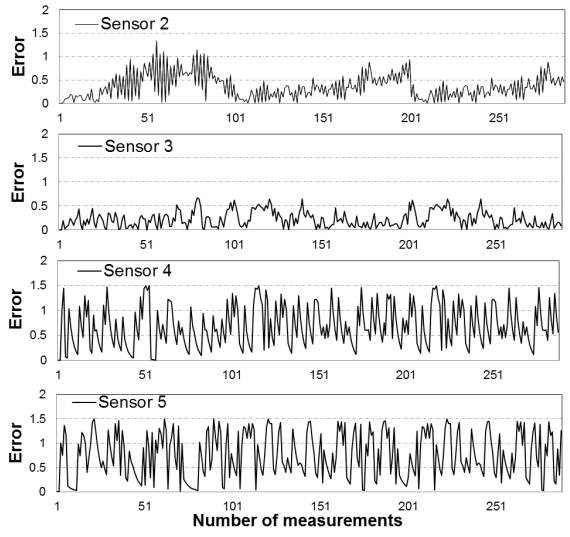


Fig. 9 Error of the measured data from each sensor and the corresponding model computed data to determine the spatial regressive threshold.

[20] equations for the identified AR(2) model. In the series, the autocorrelation at lag 1 is $r_1 = 0.807$ and autocorrelation at lag 2 is $r_2 = 0.429$. The equations for the estimators of this series are

$$1.000\hat{\varphi_1} + 0.807\hat{\varphi_2} = 0.807 \quad (15)$$

$$0.807\hat{\varphi_1} + 1.000\hat{\varphi_2} = 0.429 \quad (16)$$

which has a solution $\hat{\varphi_1}$ =1.321 and $\hat{\varphi_2} = -0.637$. Since $c = \mu(1 - \varphi_1 - \varphi_2)$, then it can be estimated $c = 46.590(1 - 1.321 - 0.637) = 14.9$. Thus the estimated model is $d_{ms1(t)} = 1.321 \times d_{s1(t-1)} - 0.637 \times d_{s1(t-2)} + 14.9$. We construct the temporal model for the other four sensors by using the same procedure, where all of them are AR(2) model with different parameters.

To determine the value of SR_{th} for stochastic data pattern, we need the history information of all the measured data. Through this information, we can compute the value of SR_{th} before we perform the ETSDR scheme. In other words, the value of SR_{th} is predefined at the initialization stage of the ETSDR scheme. To show how the value of SR_{th} is obtained, we plot the errors of four one-hop neighbors of the sensor 1 without the disturbance effects as shown in Fig. 9. The graphs show that the stochastic data changes very frequently thus, the model computed data has the model generated error. The model computed data are obtained from one whole day with the interval sensing of 5 minutes. Through this graph, we can set the value of SR_{th} is 1.5, which is use for our first simulation. The parameters and values used in first simulation are shown in Table 2.

To evaluate the periodic pattern, we perform evaluation on the Electrocardiography (ECG) data collect from [21]. The analyzer determines that ECG data has a periodic pattern since, it matched with one of its stored data. ECG data has a known pattern, which includes P wave, a QRS complex, and T wave. In addition, there is a interval between the waves, such as PR interval indicates the interval between

Table 2 Parameter settings of first simulation.

Parameter	Value
p : order for AR model	2
q : order for MA model	0
n : no. of data for model identification	100
m : no. of data for verification	80
C : maximum no. of attempts	6
Spatial Regressive Threshold (SR_{th})	1.5

Table 3 Parameter settings of second simulation.

Parameter	Value
PR interval	0.12–0.20 s
QRS duration	0.12 s
QT interval	0.43 s
RR interval	0.60–1.00 s
n : no. of data for model identification	250
m : no. of data for verification	240

'P' wave and 'R' wave, RT interval denotes the interval between 'R' wave and 'T' wave, and RR interval indicates the interval between 'R' wave to next 'R' wave. Moreover, 'Q' to 'R' to 'S' wave, formed the most obvious part of ECG, known as QRS complex, which has a fixed duration. We use these properties to generate a model for ECG in the MATLAB simulator. The ECG model parameters for normal adult people is listed in Table 3. We do our experiment for 3 lead ECG, where lead I, lead II and lead III initially started with 0°, 60° and 120° phase angle respectively. We consider this spatial property for adjustment when lead I's data is lost. We assume that lead I sensor data is missing during the transmission.

For repeated data pattern, the measured data has a fixed periodic pattern. In our second simulation, the three lead sensors have a fixed range for normal patient. In this case, the maximum value is considered as the SR_{th} for that repeated data pattern. For example, PR interval has a range 0.12–0.20 seconds for all lead sensor I, sensor II, and sensor III. When the lead sensor I's PR interval crosses 0.20, the PR intervals of the lead sensor II and lead sensor III are also affected. Thus we need to consider the spatial correlation among them.

Based on the generated data, we investigate the performance of our proposed scheme using a MATLAB. In this simulation, we assume that the single sensor produces a missing sensed data when it transmits its packet to the base station. We randomly delete the data according to the percentage of missing data from the original set and recover them using the aforementioned data recovery algorithms. We use the root mean square error (RMSE), the mean absolute error (MAE) and the integral of absolute error (IAE) to evaluate the performance of the said algorithms.

The RMSE is a frequently used measure of the difference between values estimated by an algorithm and the values actually measured from the real environment. The RMSE of an algorithm estimation with respect to the estimated value, d_e is defined as the square root of the mean squared error as written as

$$RMSE = \sqrt{\frac{\sum_{n=1}^{N} (d_{s1}(n) - d_e(n))^2}{N}} \quad (17)$$

where d_{s1} is original measured value.

The MAE is another statistical measurement that used to measure how close the estimated values are to the measured values. The MAE is given by

$$MAE = \frac{1}{N} \sum_{n=1}^{N} |d_e(n) - d_{s1}(n)| \quad (18)$$

The MAE measures the average magnitude of the errors in a data set, without considering their direction. It is also an average of the absolute error, $e = | d_e - d_{s1} |$. In other words, it measures the accuracy for the continuous variables. The MAE and the RMSE can be used together to analyze the variation in the errors of the data set. The RMSE will always be larger or equal to the MAE. The greater difference between them, the greater the variance in the individual errors in the sample [22]. If the RMSE is equal to the MAE, then all the errors are the same magnitude. In [22], Wilmott, et al. indicate that the MAE is the most natural and unambiguous measure of average error magnitude.

On the other hand, the IAE is a widely used performance metric in control community, which is recorded to measure the performance of the control application. The IAE is calculated as follows

$$IAE = \int_0^t |d_e(t) - d_{s1}(t)| \, dt \quad (19)$$

where, t denotes total simulation time. In general, the larger the IAE values imply the worse the performance of the control algorithm.

5. Simulation Results and Discussion

In this section, we present our simulation results and make some discussions on the performance of algorithms. The aim of this simulation is to examine the potential of the proposed algorithm in coping with the data missing for the CPS application. In our simulation, we investigate the impact of increasing percentage of missing data on the data recovery algorithm performance. The percentage of missing data is varied from 10% to 60% in steps of 10%.

Figure 10 depicts the RMSE comparison among data recovery algorithms for stochastic traffic patterns. As the percentage of data missing increases, the proposed algorithm always shows better performance that is compared to the existing two algorithms. The reason for this improvement is because the proposed scheme estimates the data model then uses that model to generate data. On the other hand, other two algorithm always use the same combinations of previous measurement. In addition, they do not consider the effect from the neighbors. Through this simulation, we can observe that this problem also can be found at the EWMA algorithm. Both WP and EWMA algorithm use the fixed combination of previous measurements only.

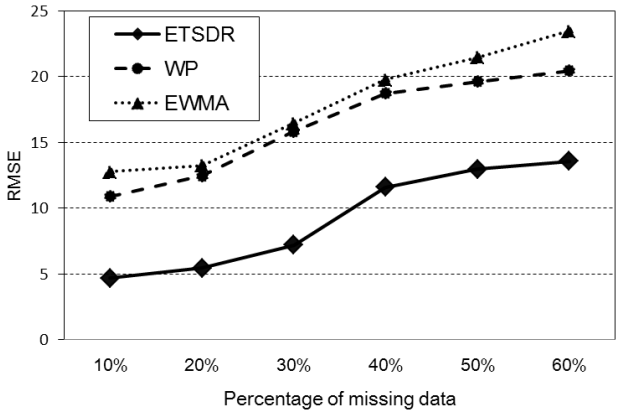


Fig. 10 The comparison of RMSE of stochastic data of all the data recovery algorithms as the percentage of missing data changes from 10% to 60%.

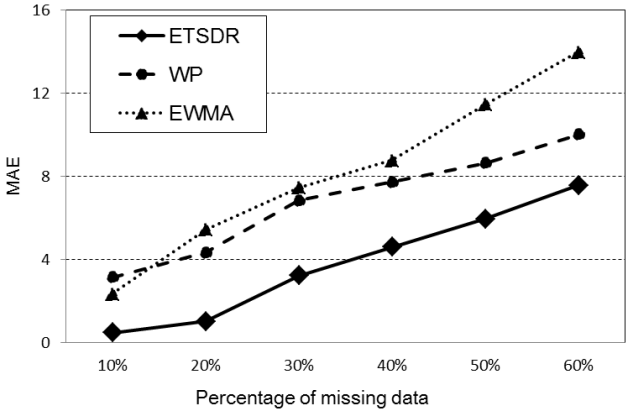


Fig. 11 The comparison of MAE of stochastic data of all the data recovery algorithms as the percentage of missing data changes from 10% to 60%.

Thus, they unable to cope with long consecutive missing and frequent changes in the environment of the conducted experiments.

The MAE comparison for stochastic data traffic among three data recovery algorithms is shown in Fig. 11. We can see that the proposed scheme outperforms the WP algorithm and the EWMA algorithm. Besides that, the proposed scheme can steadily maintain a small value of MAE regardless of the increment of missing data. This also means that the distance between the real measured data and estimated data of the proposed scheme is always stable.

In Fig. 12, the accumulated IAE comparison for stochastic data traffic of all the data recovery algorithms is plotted. The simulation results demonstrate that the proposed scheme outperforms the WP algorithm and the EWMA algorithm. In the 30% data missing the proposed algorithm's IAE is 0.63 on the other hand the IAE of WP and EWMA is 1.93 and 4.02 respectively. At 50% data missing, the proposed scheme's IAE is five times smaller than the EWMA algorithm.

From Fig. 13 to Fig. 15, the RMSE, MAE and IAE comparison for ECG data of all the data recovery algo-

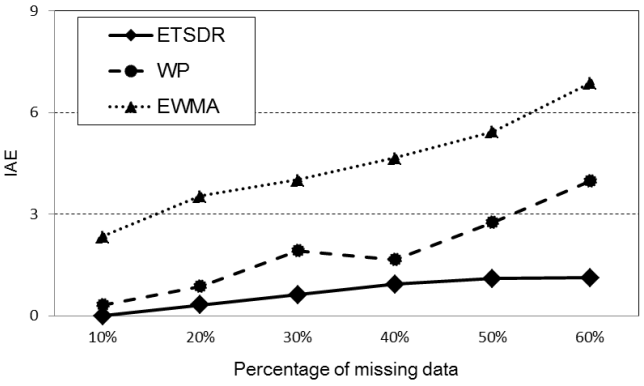


Fig. 12 The comparison of IAE of stochastic data of all the data recovery algorithms as the percentage of missing data changes from 10% to 60%.

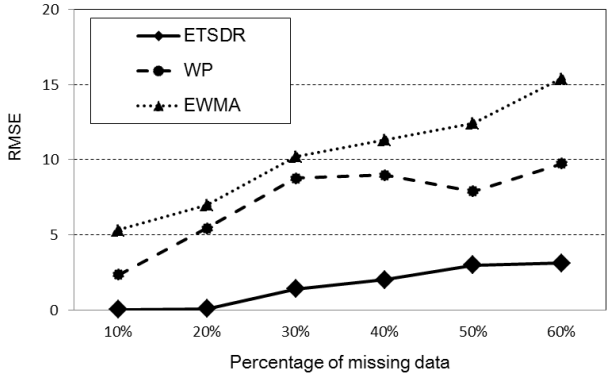


Fig. 13 The comparison of ECG data's RMSE of all the data recovery algorithms as the percentage of missing data changes from 10% to 60%.

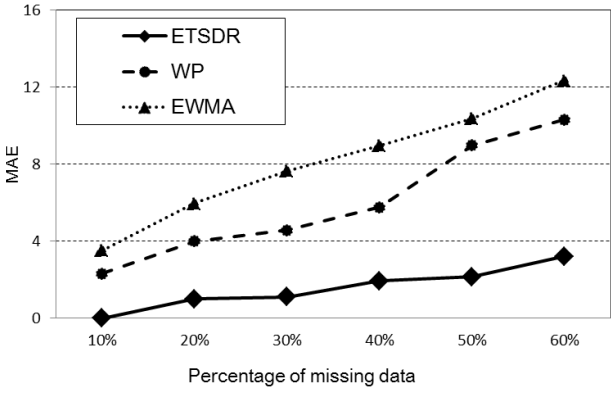


Fig. 14 The comparison of ECG data's MAE of all the data recovery algorithms as the percentage of missing data changes from 10% to 60%.

rithms is plotted. The simulation results demonstrate that the proposed scheme outperforms the other algorithms dramatically. This is because ECG data has its own pattern and we generate the model based on that pattern. Moreover to handle irregular ECG we consider spatial correlation with lead II and lead III sensors.

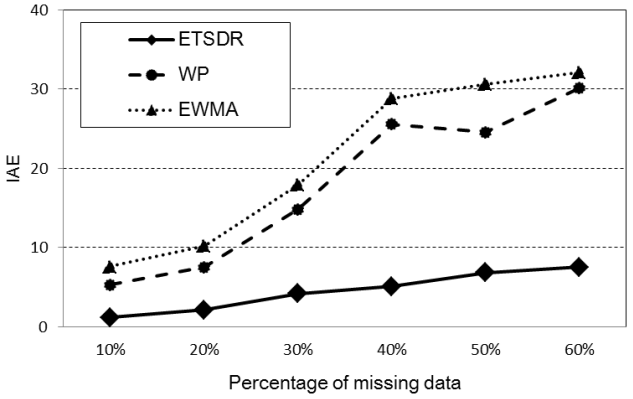


Fig. 15 The comparison of ECG data's IAE of all the data recovery algorithms as the percentage of missing data changes from 10% to 60%.

6. Conclusion

In this paper, we have proposed a model based data recovery framework for different traffic patterns of CPS. Based on the analysis, a model based ETSDR scheme for stochastic and repeated is proposed in this paper. Since, stochastic data is more difficult to estimate than the deterministic data, to handle the stochastic data we incorporate the model from that data pattern. Our simulation results reveal that the proposed ETSDR scheme is very beneficial and outperforms the WP and the EWMA algorithms regardless of the increment of missing data because of incorporating model before the recovery.

Moreover, further research is required to improve the analyzer by examining more time-critical traffic patterns. Besides that, a future work will focus on examining the real-time recovery using the proposed ETSDR scheme. In addition, we plan to incorporate prediction analysis to estimate missing data for different traffic pattern.

References

[1] A.L. Edward, "Cyber physical systems: Design challenges," IEEE Symp. on Object Oriented Real-Time Distributed Computing, pp.363–369, 2008.
[2] A.L. Edward, "CPS foundations," ACM/IEEE Design Automation Conf. (DAC), pp.737–742, 2010.
[3] F.J. Wu, Y.F. Kao, and Y.C. Tseng, "From wireless sensor networks towards cyber physical systems," J. Pervasive and Mobile Comp., vol.7, no.4, pp.397–413, 2011.
[4] F.J. Wu, Y.F. Kao, and Y.C. Tseng, "From wireless sensor networks towards cyber physical systems," J. Pervasive and Mobile Comp., vol.7, no.4, pp.397–413, 2011.
[5] F. Martincic and L. Schwiebert, "Introduction to wireless sensor networking," in Handbook of Sensor Networks-Algorithms and Architectures, pp.20–29, John Wiley & Sons, New York, USA, 2005.
[6] A.L. Edward, "Towards a science of cyber-physical system design," ACM/IEEE Conf. on Cyber-physical System, pp.99–108, April 2011.
[7] E. Yeniaras, J. Lamaury, Z. Deng, and N.V. Tsekos, "Towards a new cyber-physical system for MRI-guided and robot-assisted cardiac procedures," IEEE Int. Conf. on Information Technology and Applications in Biomedicine, pp.1–5, Nov. 2010.

[8] G. Box, G. Jenkins, and G. Reinsel, Time Series Analysis: Forecasting and Control, 4th ed., pp.47–92, Wiley, NJ, 2008.
[9] N. Nower, T. Yasuo, and A.O. Lim, "Efficient spatial data recovery scheme for cyber-physical system," IEEE Int. Conf. on Cyber-Physical Systems, Networks and Applications, pp.72–77, 2013.
[10] K. Chen and S. Lien, M2M Communications: Technologies and challenges, Elsevier Ad Hoc Networks, vol.18, pp.3–23, July 2014.
[11] L. Tang, X. Yu, S. Kim, Q. Gu, J. Han, A. Leung, and T.L. Porta, "Trustworthiness analysis of sensor data in cyber-physical systems," J. of Comp. and System Sciences, vol.79, pp.383–401, 2013.
[12] L. Tang, Q. Gu, X. Yu, J. Han, T.L. Porta, A. Leung, T. Abdelzaher, and L. Kaplan, "IntruMine: Mining intruders in untrustworthy data of cyber-physical systems," Int. Conf. on Data Mining (SDM), pp.600–611, 2012.
[13] L. Tang, X. Yu, Q. Gu, J. Han, A. Leung, and T.L. Porta, "Mining lines in the sand: On trajectory discovery from untrustworthy data in cyber-physical system," ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD), pp.410–418, 2013.
[14] L. Tang, Q. Gu, S. Kim, J. Han, W. Peng, Y. Sun, A. Leung, and T.L. Porta, "Multidimensional sensor data analysis in cyber-physical system: An atypical cube approach," Int. J. of Distributed Sensor Network, vol.2012, pp.1–19, 2012.
[15] F. Xia, X. Kong, and Z. Xu, "Cyber-physical control over wireless sensor and actuator networks with packet loss," in Wireless networking based control, pp.85–102, Springer, 2011.
[16] R.H. Choi, S.C. Lee, D.H. Lee, and J. Yoo, "WiP abstract: Packet loss compensation for cyber-physical control systems," IEEE/ACM Int. Conf. on Cyber-Physical Systems (ICCPS), p.205, 2012.
[17] G. Li and Y. Wang, "Automatic ARIMA modeling-based data aggregation scheme in wireless sensor networks," EURASIP J. Wireless Comm. and Networking, vol.2013, no.85, pp.1–13, 2013.
[18] B.L. Bowerman and R.T.O' Connell, Forecasting and Time Series: An Applied Approach, China Machine Press, Beijing, 2003.
[19] G.E. Ljung and G.E.P. Box, On a measure of lack of fit in time series models, Biometrika, 1978.
[20] R.J. Hyndman, "Yule-Walker estimators for continuous-time autoregressive models," J. of Time Series Analysis, vol.14, no.3, pp.281–296, 1993.
[21] G.B. Moody, R.G. Mark, and A.L. Goldberger, "PhysioNet: A Web-based resource for study of physiologic signals," IEEE Trans. Eng. in Medicine and Biology, vol.20, no.3, pp.70–75, 2001.
[22] C.J. Wilmott and K. Matsuura, "Advantages of the mean absolute error (MAE) over root means square (RMSE) in assessing average model performance," Climate Research, vol.30, pp.79–82, 2005.

Naushin Nower received her B.Sc. and M.S. degrees in Computer science and Engineering from the University of Dhaka, Bangladesh, in 2007 and 2009, respectively. Now she is a Ph.D. student in Japan Advanced Institute of Science and Technology and the faculty member of Institute of Information Technology, University of Dhaka, Dhaka 1000, Bangladesh. Her research interests include Cyber physical systems, Data Mining, Mobile Ad hoc Networking, Reversible logic.

Yasuo Tan received his Ph.D. from Tokyo Institute of Technology in 1993. He joined Japan Advanced Institute of Science and Technology (JAIST) as an assistant professor of the School of Information Science in 1993. He has been a professor since 1997. He is interested in Ubiquitous Computing Systems especially Home Networking Systems. He is a leader of Residential ICT SWG of New Generation Network Forum, a chairman of Green Grid Platform at Home Alliance, an advisory fellow of ECHONET Consortium, and a member of IEEE, ACM, IPSJ, IEICE, IEEJ, JSSST, and JNNS.

Azman Osman Lim received the B.Eng. (Hons) and M.Inf. Technology degrees from Universiti Malaysia Sarawak (UNIMAS), Malaysia in 1998 and 2000, respectively. He received the Ph.D. degree in communications and computer engineering from Kyoto University in 2005. He was a visiting researcher at Fudan University in China for two months. During 2005–2009, he was an expert researcher at National Institute of Information and Communications Technology (NICT), Japan. Since 2009, he has been working at Japan Advanced Institute of Science and Technology (JAIST) as an associate professor. His research interests include multihop wireless networks, wireless sensor networks, home networks, wireless mesh networks, heterogeneous wireless networks, network coding, cyber-physical system. He is a member of IEEE, IEICE, and IPSJ.