

Article

Sensor Fusion and State Estimation of IoT Enabled Wind Energy Conversion System

Md. Noor-A-Rahim ^{1,2,*}  and M. O. Khyam ³, Xinde Li ⁴  and Dirk Pesch ¹

¹ School of Computer Science and IT, University College Cork, T12 K8AF Cork, Ireland; d.pesch@cs.ucc.ie

² Nimbus Centre, Cork Institute of Technology, T12 P928 Cork, Ireland

³ Department of Mechanical Engineering, Virginia Tech, Blacksburg, VA 24061, USA; mok@vt.edu

⁴ School of Automation, Southeast University, Nanjing 210096, China; xindeli@seu.edu.cn

* Correspondence: m.rahim@cs.ucc.ie or Mohammad.Rahim@cit.ie; Tel.: +353-89-9673-240

Received: 31 January 2019; Accepted: 26 March 2019; Published: 1 April 2019

Abstract: The use of renewable energy has increased dramatically over the past couple of decades. Wind farms, consisting of wind turbines, play a vital role in the generation of renewable energy. For monitoring and maintenance purposes, a wind turbine has a variety of sensors to measure the state of the turbine. Sensor measurements are transmitted to a control center, which is located away from the wind farm, for monitoring and maintenance purposes. It is therefore desirable to ensure reliable wireless communication between the wind turbines and the control center while integrating the observations from different sensors. In this paper, we propose an IoT based communication framework for the purpose of reliable communication between wind turbines and control center. The communication framework is based on repeat-accumulate coded communication to enhance reliability. A fusion algorithm is proposed to exploit the observations from multiple sensors while taking into consideration the unpredictable nature of the wireless channel. The numerical results show that the proposed scheme can closely predict the state of a wind turbine. We also show that the proposed scheme significantly outperforms traditional estimation schemes.

Keywords: Wind energy; Sensor fusion; State estimation; Internet of Things (IoT); Renewable energy

1. Introduction

Demand for renewable energy is rapidly increasing in order to counteract progress of global warming and diminishing natural resources. Wind turbines play a significant role in renewable energy generation. It is predicted that wind farms, consisting of many wind turbines, will supply about 10% of total electricity by the year 2020 [1,2]. Usually, wind turbines are placed in remote areas where plenty of wind is available. Examples of such places are off or near the seashore or on mountains. For maintenance and monitoring purposes, various sensors are embedded into a wind turbine to measure the state of the turbine. However, convenient monitoring of the state requires remote communication connectivity to the wind turbines. A viable solution to this problem can be an Internet of Things (IoT) based network [3–5]. Integration of the wind turbines into an IoT network can enable a real-time monitoring system to monitor turbine performance and detect faults so that technicians can take necessary actions promptly. As communication links between the wind turbines and the Internet will likely take place over wireless communication channels, a reliable communication framework is required between the wind turbines and wireless base stations or access points connected to the Internet. On top of that, to process and combine the multiple sensor readings, a fusion algorithm needs to be developed for accurate state estimation while taking into account the uncertainty of the wireless channel.

In previous works, various algorithms and models have been developed to estimate the state of the wind turbine and examples of such works are [6–16] (see Table 1). A sensor selection procedure for

estimation of wind turbine rotor state and wind-induced load state was presented in [6]. Considering nonlinear observers, a wind turbine dynamic state and parameter estimation technique was described in [8]. In [9], a semi-empirical method for wind turbine state estimation was proposed. In conjunction with the Newton-Raphson method, a Kalman filter based estimation technique was described in [10] to estimate the aerodynamic torque acting on the rotor of the wind turbine. To overcome drawbacks due to the decoupling effects, an extended state estimation method was proposed in [11], where the models for the different wind turbine technologies were incorporated. A type of non-linear state estimator called particle filter was utilized in [12] to estimate the wind turbine parameters: tower top displacement, tower stop velocity and the rotor speed. More recently, authors in [13–16] studied dynamic state estimation of doubly fed induction generator (DFIG) based wind turbine and permanent magnet synchronous generator (PMSG) based wind turbine. However, most of the previous work did not take into account the impact of an unreliable wireless channel (between the wind turbines and the remote controller) on the state estimation of the wind turbine. Moreover, IoT based network can play a vital role in enabling a reliable and real-time state estimation and controlling of wind turbines. Although the potentials of IoT were discussed in a few studies [3–5,17], not much is known regarding wireless communication framework of IoT based state estimation of wind turbine. On the other hand, the embedded sensors in a wind turbine measure different components of the turbine independently. Thus, an appropriate sensor fusion algorithm is required to combine the readings of different sensors for accurate state estimation. Note that previous research, including [18–24] considered sensor fusion techniques for generic observable plants. However, to the best of our knowledge, no work has considered sensor fusion techniques in the context of turbine-based wind energy systems.

Table 1. Comparison of different works on state estimation of wind turbine.

Works	Type of Wind Turbine	Filter Type	Sensor Fusion	Impact of Wireless Channel	Error Correction Technique
Berg et al. [6]	Generic	Linear Kalman	No	No	No
Ritter et al. [8]	Generic	Linear Kalman	No	No	No
Petar et al. [9]	Generic	Extended Kalman	No	No	No
Bourlis et al. [10]	Generic	Adaptive Kalman	No	No	No
Blanco et al. [11]	Generic	Extended Kalman	No	No	No
Sudev et al. [12]	Generic	Particle filter	No	No	No
Yu et al. [13]	DFIG	Unscented Kalman	No	No	No
Yu et al. [14]	DFIG	Unscented Kalman	No	No	No
Prajapat et al. [15]	DFIG	Unscented Kalman	No	No	No
Shahriari et al. [16]	PMSG	Extended Kalman	No	No	No
This work	Generic	Linear Kalman	Yes	Yes	Yes

To address these shortcomings, this paper studies sensor fusion and state estimation of wind turbines over a wireless IoT network. We present a communication framework for wind turbines and remote IoT components. To achieve a reliable communication between the wind turbines and the remote observer, we present a repeat-accumulate coded transmission scheme over a wireless channel. To exploit different observations from the sensors, we propose a Kalman filter based sensor fusion technique while taking into consideration the error events caused by the wireless channel. Through the numerical results, we show that the proposed estimation technique can closely predict the state of the wind turbine. We also show that the proposed scheme significantly outperforms traditional state estimation techniques.

The remainder of this paper is organized as follows. In Section 2, we present an IoT enabled wind energy conversion system and linearized state space model of wind turbine's induction motor. The proposed communication framework is presented in Section 3. In Section 4, we propose Kalman filter based sensor fusion technique. We present the simulation results of the proposed fusion scheme in Section 5 along with the improvement over the traditional scheme.

2. IoT Enabled Wind Energy Conversion System and State Space Model

In general, wind farms are situated in remote locations and thus the control centers are typically located several hours away from the wind farms. Remote data communication connectivity through

the Internet in the form of an IoT network can help the control centers to monitor the state of a wind turbine and to control a wind turbine's operation. Due to the remote location of wind turbines, the IoT network connectivity requires wireless networks such as cellular or satellite networks. An integrated scenario comprised of wind turbines and a wireless IoT network [25] is depicted in Figure 1.

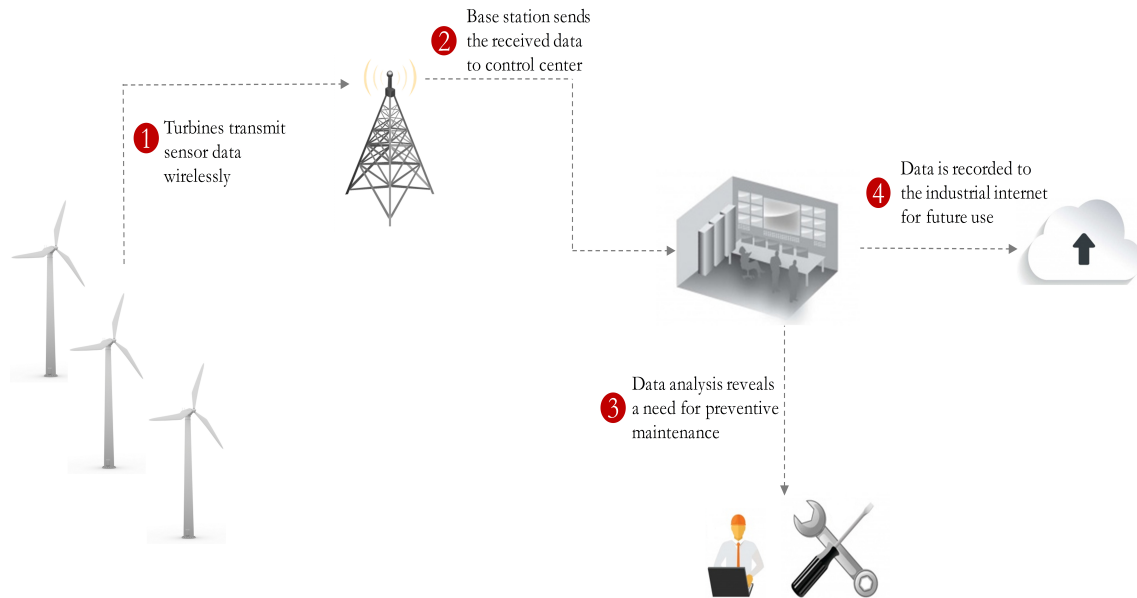


Figure 1. Use of Internet of Things (IoT) network for transmitting data from turbines to control center and technicians.

A wind turbine system consists of induction generator, rotors, gearbox, and matrix converter [26,27]. The matrix converter connects the induction generator with the power grid and controls the output power delivered to the grid. In this paper, we track the output current components of an induction generator in a wind turbine. We adopt the state space model reported in [28], where a fixed-speed wind turbine is considered. Authors in [28] utilized the current model to characterize the induction generator and the following assumptions were considered for the sake of simplicity: (1) the stator current is assumed to be negative when it flows toward the machine; (2) a synchronous reference frame was considered to derive the equations; (3) the d-axis is 90 deg behind the q-axis (Direct axis (d-axis) is the axis of the stator/rotor's salient pole and quadrature axis (q-axis) is the axis in quadrature or perpendicular to the stator/rotor's salient pole.). With the above assumptions, we adopt the linearized state space current model of induction generator, which is described by the following equation:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{U} + \mathbf{\Gamma}\mathcal{N}_p \quad (1)$$

where $\mathbf{X} = [i_{ds} \ i_{qs} \ i_{dr} \ i_{qr}]^T$ is the state of the induction generator, $\mathbf{U} = [v_{ds} \ v_{qs} \ v_{dr} \ v_{qr}]^T$ is control input, $\mathbf{\Gamma}$ is a constant matrix, and \mathcal{N}_p is process noise. We model the process noise as Gaussian noise with zero-mean and covariance matrix \mathbf{Q} . In \mathbf{X} , i_{ds} and i_{qs} are the stator currents in the d and q axes, respectively; i_{dr} and i_{qr} are the rotor currents in the d and q axes, respectively. In \mathbf{U} , v_{ds} and v_{qs} are the stator voltages in the d and q axes, respectively; v_{dr} and v_{qr} are the rotor voltages in the d and q axes, respectively. With the specifications of a single-cage induction generator, \mathbf{A} and \mathbf{B} are characterized by [28,29],

$$\mathbf{A} = \frac{\omega_b}{X_s X_r \gamma} \begin{bmatrix} -R_s X_r & \alpha_1 \omega_s & -R_r X_m & -\beta_r \omega_s \\ -\alpha_1 \omega_s & -R_s X_r & \beta_r \omega_s & -R_r X_m \\ -R_s X_m & \beta_s \omega_s & -R_r X_s & -\alpha_2 \omega_s \\ -\beta_s \omega_s & -R_s X_m & \alpha_2 \omega_s & -R_r X_s \end{bmatrix}$$

$$\mathbf{B} = \frac{\omega_b}{X_s X_r \gamma} \begin{bmatrix} -X_r & 0 & X_m & 0 \\ 0 & -X_r & 0 & X_m \\ -X_m & 0 & X_s & 0 \\ 0 & -X_m & 0 & X_s \end{bmatrix}$$

where ω_b , ω_s , ω_r are base, stator, rotor angular speeds, respectively; s is the slip defined by $s = \frac{\omega_s - \omega_r}{\omega_s}$; R_s , R_r Stator, rotor resistances, respectively; X_s , X_r , X_m stator, rotor, magnetizing reactances, respectively; $\alpha_1 = X_s X_r - s X_m^2$, $\alpha_2 = X_m^2 - s X_s X_r$, $\beta_s = X_m X_s (1 - s)$, $\beta_r = X_m X_r (1 - s)$, $\gamma = 1 - \frac{X_m^2}{X_r X_s}$.

For the sake of simplicity, we discretise Equation (1) in the following form [30]:

$$\mathbf{X}(t+1) = \mathbf{A}_d \mathbf{X}(t) + \mathbf{B}_d \mathbf{U}(t) + \mathbf{\Gamma} \mathcal{N}_p \quad (2)$$

where \mathbf{A}_d and \mathbf{B}_d are obtained by,

$$\begin{aligned} \mathbf{A}_d &= \exp(\mathbf{A}\Delta t) \simeq (\mathbf{I} + \mathbf{A}\Delta t) \\ \mathbf{B}_d &= \int_0^{\Delta t} \exp(\mathbf{A}z) \mathbf{B} dz \simeq \mathbf{B}\Delta t \end{aligned}$$

where Δt is the step size used for discretization.

3. Proposed Communication Framework

In a wind turbine, different types of sensors are embedded to measure the components of the wind turbine, for example, generated current, voltage, rotor speed, etc. Let $\mathcal{O}_i(t)$, $i = 1, 2, \dots, N$, be the measured state by the i^{th} sensor of the wind turbine. We define $\mathcal{O}_i(t)$ by,

$$\mathcal{O}_i(t) = \mathbf{C}_i \mathbf{X}(t) + \mathcal{N}_{m_i},$$

where \mathbf{C}_i is the measurement/sensing matrix of sensor i and \mathcal{N}_{m_i} is the observed noise during the measurement at sensor i . Similar to the process noise, we model the measurement noise as Gaussian noise with zero-mean and co-variance \mathbf{R}_i . The measured state is sent periodically to the control center for the appropriate action to be taken. Due to the remote placement of wind turbines there often is no direct communication link between a wind turbine and the control center. In most cases, a transmitter in the wind turbine communicates with a nearby base station, from where the message is relayed to the control center. We assume that the communication link between the base station and the control center is reliable as it is part of a fixed backbone network. However, due to the wireless communication channel between the wind turbine and the base station it is challenging to maintain reliable data communication, although reliable communication is highly desirable for accurate state estimation and control applications. To achieve reliable communication between the wind turbine and the base station we propose the following communication strategy. We define the observed state as $\mathcal{O}_i(t) = [o_{i1}, o_{i2}, \dots, o_{ip}]$, where $o_{ij}(t)$ is the j^{th} component of \mathbf{X} , which is measured by the i^{th} sensor. Each component of $\mathcal{O}_i(t)$ is mapped and quantized into K bits. The bit block corresponding to component j is represented by $\mathbf{b}_{ij}(t) \in \{0, 1\}^K$. A repeat-accumulate code is then applied over $\mathbf{b}_{ij}(t)$ to generate a code word $\mathbf{c}_{ij}(t) \in \{0, 1\}^{\frac{K}{\eta}}$, where η is the rate of the code. All code words are arranged serially to form $\mathbf{m}_i(t) = [c_{i1}(t), c_{i2}(t), \dots, c_{ip}(t)]$. After modulating the $\mathbf{m}_i(t)$ onto the wireless carrier signal, the resultant carrier signal $\mathbf{s}_i(t)$ is then transmitted from the wind turbine to the base station. Let $\hat{\mathbf{s}}_i(t)$ be the received signal at the base station, which is defined by,

$$\hat{\mathbf{s}}_i(t) = \mathbf{s}_i(t) + \mathcal{N}_w,$$

where \mathcal{N}_w is the additive white Gaussian noise (AWGN) with zero mean and standard deviation σ_w . Upon receiving $\hat{\mathbf{s}}_i(t)$, the receiver performs the reverse process (i.e., demodulation, decoding,

demapping, etc.) in order to construct the observed state. Let $\hat{\mathcal{O}}_i(t)$ be the reconstructed observed state which corresponds to $\mathcal{O}_i(t)$. The reconstructed observed state $\hat{\mathcal{O}}_i(t)$ is then fed to our proposed fusion algorithm (discussed in Section 4) to track the original state of the wind turbine. The overall communication framework is depicted in Figure 2. In the following, we briefly describe the repeat-accumulate code along with its encoding and decoding procedures.

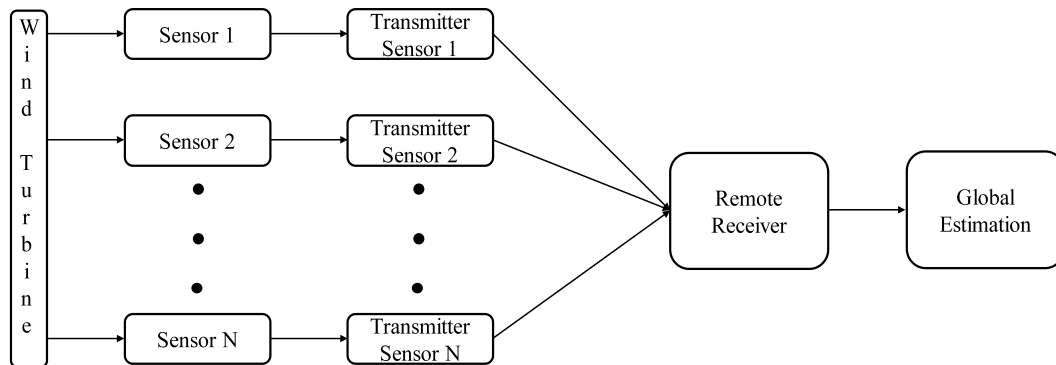


Figure 2. Communication framework of multi-sensor wind turbine.

Repeat Accumulate (RA) Codes

Low-density parity-check (LDPC) codes are best known for their capacity approaching performance and low complexity decoding property [31,32]. Repeat accumulate (RA) codes are a special type of LDPC codes which inherit the above properties while allowing low encoding complexity [33–35]. Similar to the LDPC codes, RA codes can be represented by a bipartite graph with K information variable nodes, M parity variable nodes, and M check nodes. Each information node represents an information bit, while each check node represents a check equation satisfying the condition that the modulo 2 sum of all the connected nodes will be zero. The value of each parity node, which represents a redundant/parity bit, is generated such that the check equation satisfies the above condition. In an RA code, each information node is connected to more than one check node, while each parity bit node has connection with exactly two check nodes (except the last parity node, which connects to only one check node). We connect the parity bit nodes and the check nodes such that the i^{th} parity bit node always connects with the check nodes at position i^{th} and $(i - 1)^{th}$. The parity/redundant bits or the value of the parity nodes are generated in the following manner. The i^{th} parity bit node's value is calculated by performing modulo-2 sum of the information bit nodes that have connection with the i^{th} check node and the $(i - 1)^{th}$ parity bit node. A RA code is referred to as (q, a) -regular code, when each of the information bit node is connected with exactly q check nodes and each of the check nodes is connected with exactly a information bit nodes. A Tanner graph representation of a $(q, a) = (3, 3)$ regular RA code is depicted in Figure 3. Note that a Tanner graph is equivalent to a binary matrix \mathbf{H} of size $M \times (K + M)$, where each column and row represent variable and check nodes, respectively and each non zero entry represents a connection between the corresponding variable and check nodes.

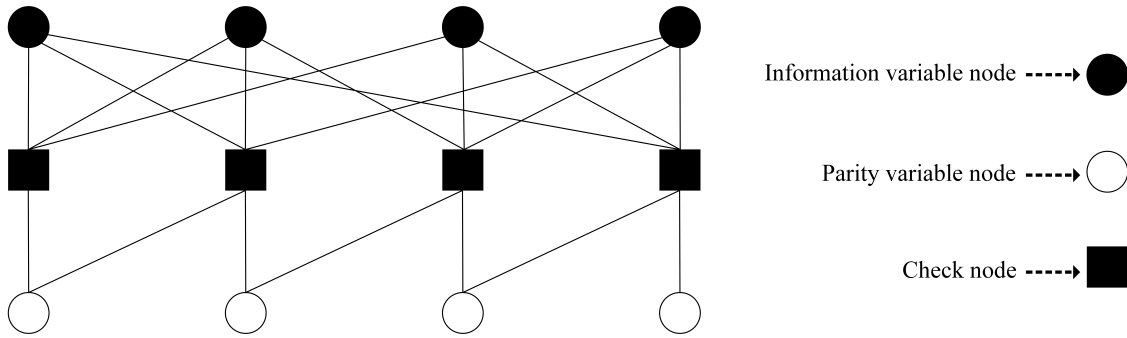


Figure 3. A Tanner graph representation of a repeat accumulate code with $(q, a) = (3, 3)$. In the graph, filled and unfilled circle nodes represent the information and parity bits, respectively, while rectangular nodes represent check equations.

Belief Propagation Decoding

A belief propagation (BP) decoding algorithms, which belong to the category of message passing algorithms, is among the best known decoding algorithms over binary input AWGN channels [36]. In a BP decoding algorithm, messages that passed between nodes are represented by log likelihood ratios (LLRs) (see Equations (3)–(5) in the following). Let x_m be the m^{th} bit of a codeword (i.e., m^{th} variable node in the Tanner graph) and y_m be the corresponding channel output. For variable node m , the initial (channel) LLR is given by ([37] [Chapter 2]),

$$r_m = \log \frac{P_{\text{ch}}(y_m | x_m = 0)}{P_{\text{ch}}(y_m | x_m = 1)}. \quad (3)$$

For simplicity, we use the following notations to describe the BP decoding algorithm:

- $\mathbb{S}_v(m) \rightarrow$ set of variable nodes that have connection/edge with the m^{th} check node.
- $\mathbb{S}_c(m) \rightarrow$ set of check nodes that have connection/edge with the m^{th} variable node.
- $\mathcal{V}_{n,m}^{(\ell)} \rightarrow$ LLR message sent from variable node m to check node n at iteration ℓ .
- $\mathcal{C}_{n,m}^{(\ell)} \rightarrow$ LLR message sent from check node n to variable node m at iteration ℓ .

Message from check node:

$$\mathcal{C}_{n,m}^{(\ell)} = 2 \tanh^{-1} \left(\prod_{m' \in \mathbb{S}_v(n), m' \neq m} \tanh \left(\frac{\mathcal{V}_{n,m'}^{(\ell)}}{2} \right) \right). \quad (4)$$

Message from variable node:

$$\mathcal{V}_{n,m}^{(\ell+1)} = \sum_{n' \in \mathbb{S}_c(m), n' \neq n} \mathcal{C}_{n',m}^{(\ell)} + r_m. \quad (5)$$

The above decoding process is initialized by sending the channel LLRs from each variable node to the connected check nodes. After maximum iteration, the ℓ_{max} , m^{th} decoded bit is given by

$$\hat{x}_m = \begin{cases} 0 & \text{if } \sum_{n \in \mathbb{S}_c(m)} \mathcal{C}_{n,m}^{(\ell_{\text{max}})} + r_m \geq 0 \\ 1 & \text{if } \sum_{n \in \mathbb{S}_c(m)} \mathcal{C}_{n,m}^{(\ell_{\text{max}})} + r_m < 0 \end{cases} \quad (6)$$

A successful decoding event can be indicated by summed syndrome ψ , which is defined by $\psi = \sum \text{mod}_2(\mathbf{H} * \hat{\mathbf{x}}^T)$, where $\text{mod}_2(\cdot)$ is a modulo-2 operation and $\hat{\mathbf{x}} = \hat{x}_1, \hat{x}_2, \dots, \hat{x}_{K+M}$. $\psi = 0$ indicates a successful decoding event, while $\psi > 0$ indicates an unsuccessful decoding event.

4. Proposed Sensor Fusion Technique

We now present our state estimation with sensor fusion based on a Kalman filter algorithm [38,39]. We illustrate the proposed scheme in the following four steps:

Step 1—Prediction: Let $\hat{\mathbf{X}}^-(t)$ and $\hat{\mathbf{P}}^-(t)$ be the predicted state and co-variance matrix, respectively. According to the Kalman filter algorithm, we calculate $\hat{\mathbf{X}}^-(t)$ and $\hat{\mathbf{Y}}^-(t)$ by

$$\begin{aligned}\hat{\mathbf{X}}_i^-(t) &= \mathbf{A}_d \hat{\mathbf{X}}_i(t-1) + \mathbf{B}_d \mathbf{U}(t-1) \\ \mathbf{Y}_i^-(t) &= \mathbf{A}_d \mathbf{Y}_i(t-1) \mathbf{A}_d^T + \mathbf{Q}\end{aligned}$$

We denote $\mathbf{Y}_i(t)$ as the i^{th} local estimator's updated co-variance matrix at time step t . Initially we set $\mathbf{Y}_i(0)$ to a predefined value \mathbf{Y}_0 .

Step 2—Modification: We find the summed syndrome value of each component of $\hat{\mathbf{O}}_i(t)$. We remove the erroneous component from $\hat{\mathbf{O}}_i(t)$ and modify \mathbf{C}_i and \mathbf{R}_i accordingly. Let $\bar{\mathbf{O}}_i(t)$, $\bar{\mathbf{C}}_i(t)$, and $\bar{\mathbf{R}}_i(t)$ be the modified version of $\hat{\mathbf{O}}_i(t)$, \mathbf{C}_i and \mathbf{R}_i , respectively.

Step 3—Update: Let $\epsilon_i(t)$ and $\mathcal{G}_i(t)$ be the measurement pre-fit residual and Kalman gain, respectively. We calculate $\epsilon_i(t)$ and $\mathcal{G}_i(t)$ by

$$\begin{aligned}\epsilon_i(t) &= \bar{\mathbf{O}}_i(t) - \bar{\mathbf{C}}_i(t) \hat{\mathbf{X}}_i^-(t) \\ \mathcal{G}_i(t) &= \frac{\mathbf{Y}_i^-(t) \bar{\mathbf{C}}_i(t)^T}{\bar{\mathbf{C}}_i(t) \mathbf{Y}_i^-(t) \bar{\mathbf{C}}_i(t)^T + \mathbf{Q}}\end{aligned}$$

At time step k , co-variance matrix estimate and the updated state estimate are calculated by,

$$\begin{aligned}\mathbf{Y}(t) &= \mathbf{Y}^-(t) - \mathcal{G}(t) \bar{\mathbf{C}}_i(t) \mathbf{Y}_i^-(t) \\ \hat{\mathbf{X}}_i(t) &= \hat{\mathbf{X}}_i^-(t) + \mathcal{G}_i(t) \bar{\mathbf{R}}_i(t)\end{aligned}$$

Step 4—Fusion: Let $\Phi_{pq}(t)$, $p, q \in \{1, 2, \dots, N\}$ be the error cross co-variance between the p^{th} and the q^{th} sensors. $\Phi_{pq}(t)$ is given by

$$\Phi_{pq}(t|t) = \begin{cases} \mathbf{Y}_0 & \text{if } t = 0 \\ \mathbf{Y}_p(t) & \text{if } m = n \\ [\mathbf{I}_\lambda - \mathcal{G}_p(t) \bar{\mathbf{C}}_p(t)] \times \\ \quad [\mathbf{A}_d \Phi_{pq}(t-1|t-1) \mathbf{A}_d^T + \mathbf{I} \mathbf{Q} \mathbf{I}^T] \times \\ \quad [\mathbf{I}_\lambda - \mathcal{G}_q(t) \bar{\mathbf{C}}_q(t)] & \text{Otherwise} \end{cases} \quad (7)$$

where \mathbf{I}_λ is the identity matrix of size $\lambda \times \lambda$ and λ is the number of components in \mathbf{X} . Following [18,40], optimal fusion based on the linear minimum variance can be written as

$$\hat{\mathbf{X}}_g(t) = \mathbf{W}_1(t) \hat{\mathbf{X}}_1(t) + \mathbf{W}_2(t) \hat{\mathbf{X}}_2(t) + \dots + \mathbf{W}_N(t) \hat{\mathbf{X}}_N(t) \quad (8)$$

where the matrix weight $\bar{\mathbf{W}}(t)$ is given by

$$\bar{\mathbf{W}}(t) = \mathbf{\Xi}(t)^{-1} \mathcal{I} \left(\mathcal{I}^T \mathbf{\Xi}(t) \mathcal{I} \right)^{-1} \quad (9)$$

where $\bar{\mathbf{W}}(t) = [\mathbf{W}_1(t), \mathbf{W}_2(t), \dots, \mathbf{W}_N(t)]^T$ is an $\lambda N \times \lambda$ matrix, $\mathbf{\Xi}(t) = (\Phi_{pq}(t))$, $p, q = 1, 2, \dots, N$ is an $\lambda N \times \lambda N$ symmetric positive definite matrix, and $\mathcal{I} = [\mathbf{I}_\lambda, \dots, \mathbf{I}_\lambda]^T$ is an $\lambda N \times \lambda$ matrix. The overall fusion process is shown in Figure 4.

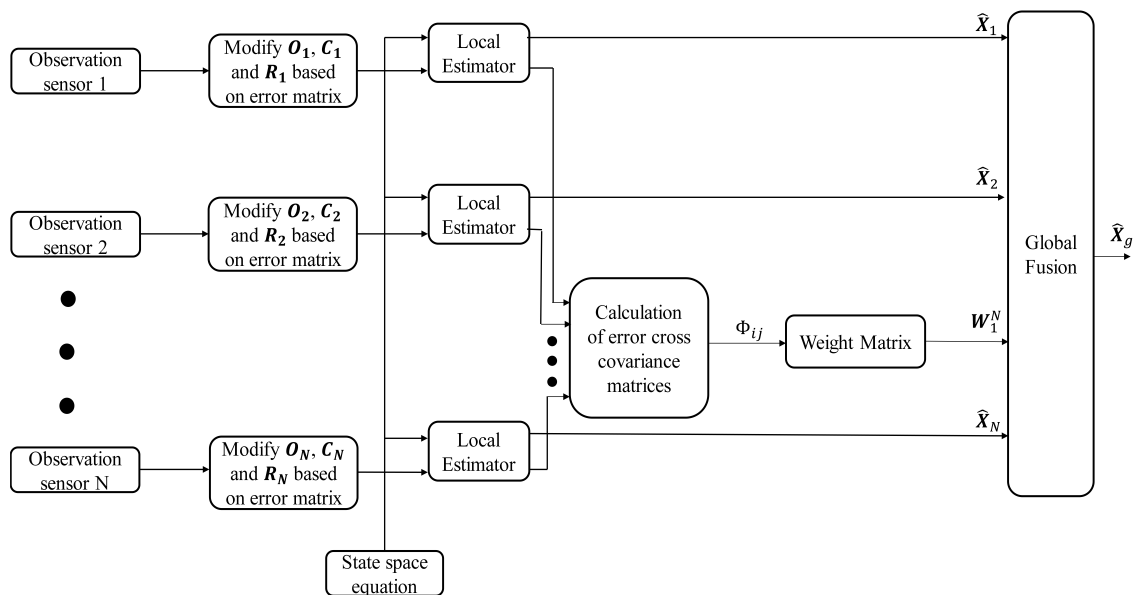


Figure 4. Block diagram of proposed sensor fusion technique.

5. Performance Evaluations

In this section, we present the performance of the proposed communication and fusion technique. We also compare the numerical results with a traditional fusion technique [18]. We built a simulation environment in Matlab to evaluate the tracking performance of the wind turbine's state. The parameters of the induction generator are shown in Table 2. We set the following values for the input parameters: the stator voltage in the d axis is $v_{ds} = 0.04$ Volt, the stator voltage in the q axis is $v_{qs} = 0.99$ Volt, the rotor voltage in the d axis is $v_{dr} = 0.02$ Volt, and the rotor voltage in the q axis is $v_{qr} = 0.206$ Volt. Note that the above parameters are adopted from the experimental setup reported in [28]. The process' noise co-variance is set to $Q = 0.95I_\lambda$ with constant matrix $\Gamma = I_\lambda$. For the following results, simulations were carried out for 150 steps with a step size of $\Delta t = 0.0001$ s.

Table 2. Induction Generator Parameters

Parameter	Value
Base frequency	10 Hz
Stator frequency	15 Hz
Rotor frequency	15 Hz
Resistance of stator	0.004 Ω
Resistance of rotor	0.005 Ω
Reactance of stator	0.09 Ω
Reactance of rotor	0.08 Ω
Magnetizing reactance	3.95 Ω

We present the estimation performance of the proposed scheme in Figures 5–8. Four sensors are used to measure the four current components of the induction generator with the following measurement matrices $C_1 = [1 \ 0 \ 0 \ 0]$, $C_2 = [0 \ 1 \ 0 \ 0]$, $C_3 = [0 \ 0 \ 1 \ 0]$, $C_4 = [0 \ 0 \ 0 \ 1]$. The measurement noise co-variances of the four sensors are set as 1.6, 1.2, 1.4, and 1.6, respectively. We consider a 32-bits uniform quantizer to map each component of the measured state. We use a rate $\frac{1}{2}$ -(q, a) = (4, 4) repeat-accumulate code over each mapped 32 bit frame and hence each component is represented by a codeword of 64 bits. This codeword is then modulated with BPSK and sent over a wireless channel with noise standard deviation 0.7. From the results, we observe that the proposed fusion technique can track the state of the wind turbine very closely. As a benchmark, we also present results obtained from a traditional fusion algorithm [18]. We show that our proposed scheme can significantly outperform

the traditional estimation approach. In case of an error event due to the wireless transmission, it is expected that the measured state received by the remote receiver will be unreliable. While the traditional fusion algorithm treats all the measured states with the same importance, the proposed algorithm ignores the measured state and gives priority to the observation from the system dynamics. We notice that the traditional fusion algorithm fails to track the state (blue spikes in the figures), which are results of erroneous transmissions. We also investigate the impact of the wireless channel on the estimation performance of the proposed scheme. In Figure 9, we show the estimation performance while varying the channel noise standard deviation σ_w . Note that in terms of noise standard deviation, the decoding threshold of the presented (4,4)-RA code is 0.885. For $\sigma_w = 0.6$ and $\sigma_w = 0.7$, we observe similar estimation performance, which indicates that the wireless channel does not influence the estimation performance when the noise standard deviation is well below the threshold. However, when the noise standard deviation is close to the threshold (for $\sigma_w = 0.8$), we observe inaccuracies in the estimation results.

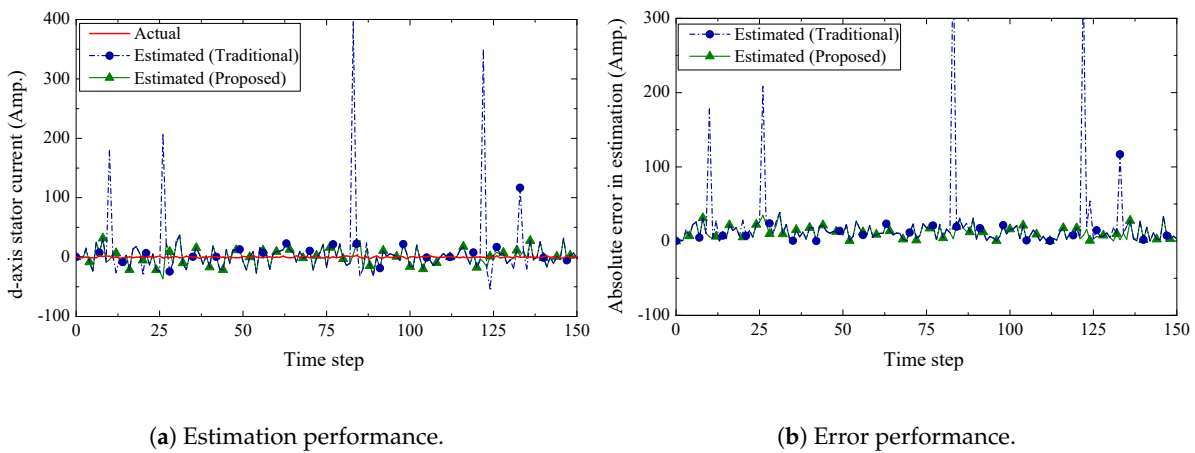


Figure 5. d-axis stator current estimation and error performance.

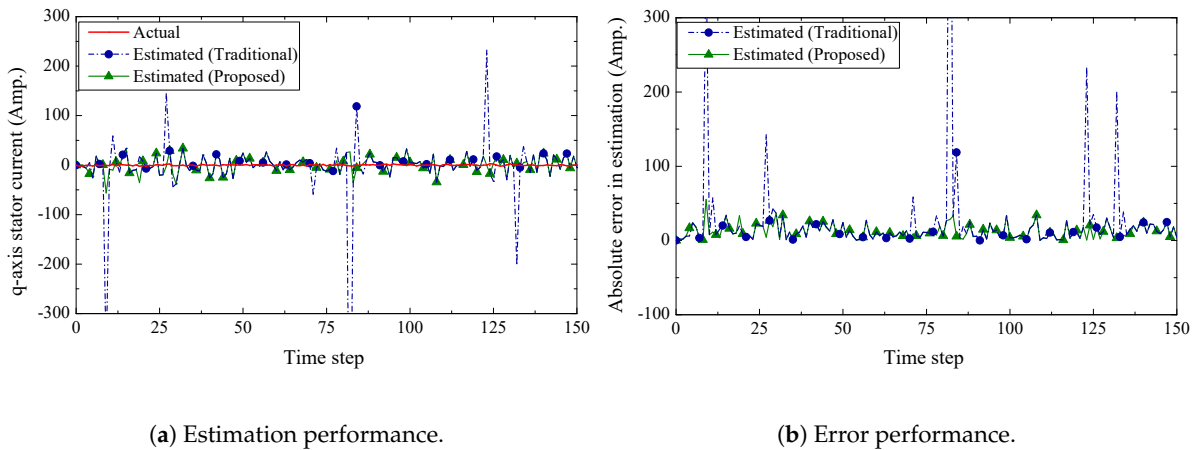
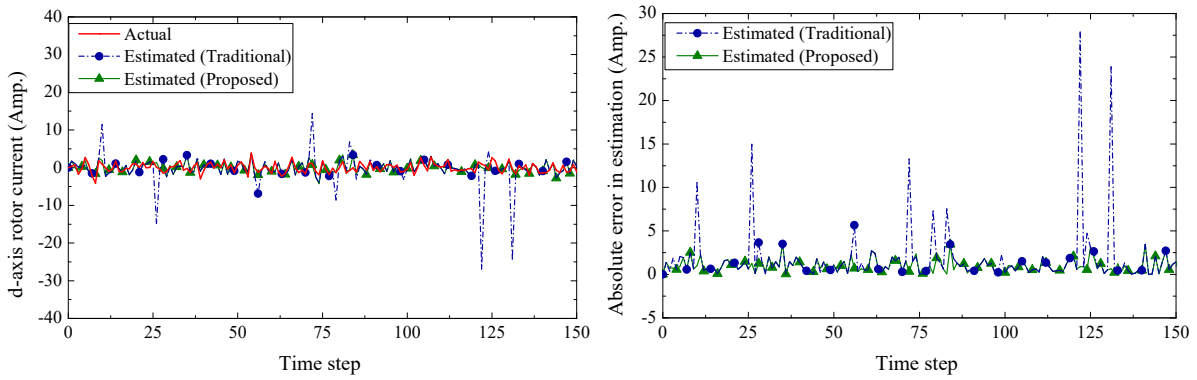


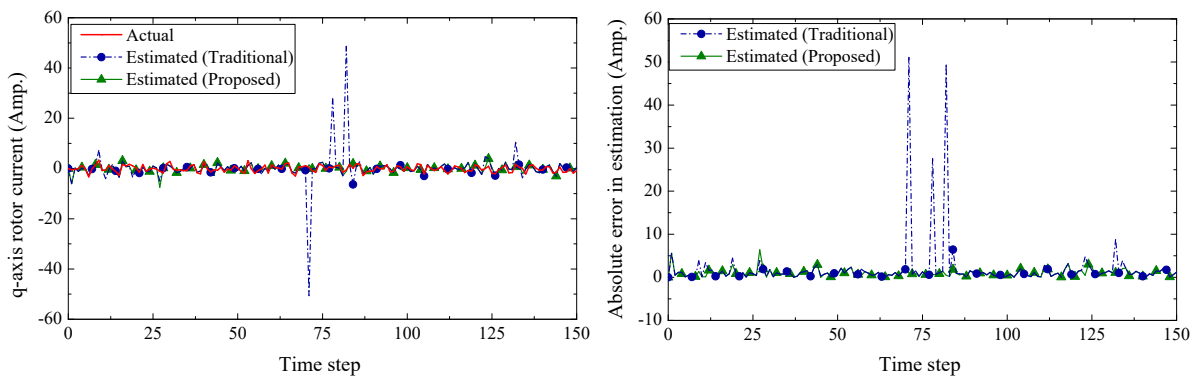
Figure 6. q-axis stator current estimation and error performance.



(a) Estimation performance.

(b) Error performance.

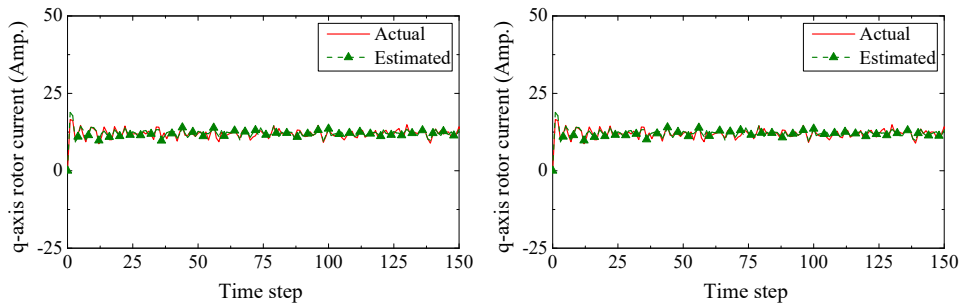
Figure 7. d-axis rotor current estimation and error performance.



(a) Estimation performance.

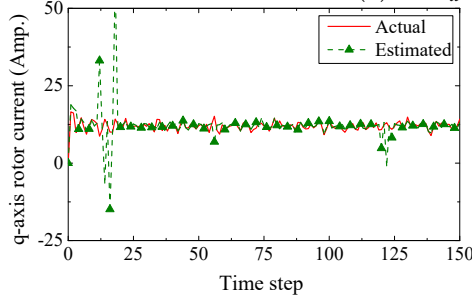
(b) Error performance.

Figure 8. q-axis rotor current estimation and error performance.



(a) For $\sigma_w = 0.6$.

(b) For $\sigma_w = 0.7$.



(c) For $\sigma_w = 0.8$.

Figure 9. Impact of wireless channel quality on estimation performance.

6. Conclusions

The communication link between wind turbines and control center is often wireless to the remote placement of wind farms. In this paper, we have presented the state estimation of a wind turbine while taking into account the uncertainty of the wireless channel. We have shown an IoT inspired wireless communication framework to monitor the states of the wind turbine. A repeat-accumulate coded communication scheme is presented to tackle the noise induced by the unreliable wireless channel. Moreover, we have proposed an effective fusion algorithm to process and combine the multiple readings from different sensors of a wind turbine. Through simulation results, we have shown that our proposed scheme can track the state of a wind turbine accurately. We have also shown that the estimation performance of the proposed scheme outperforms estimation performance of traditional fusion algorithms. In the future we aim to apply our proposed scheme in a practical test-bed to evaluate the performance for real-world use cases.

Author Contributions: M.N.-A.-R. and M.O.K. performed theoretical investigation and wrote the draft; M.N.-A.-R. and X.L. performed the simulation; D.P. supervised the work and revised the manuscript.

Funding: This work has received funding in part, from the European Union’s Horizon 2020 Research and Innovation Programme under the EDGE COFUND Marie Skłodowska Curie grant agreement No. 713567 and from the Science Foundation Ireland under CONNECT Centre grant no. 13/RC/2077.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Moradi, H.; Vossoughi, G. Robust control of the variable speed wind turbines in the presence of uncertainties: A comparison between H² and PID controllers. *Energy* **2015**, *90*, 1508–1521, doi:10.1016/j.energy.2015.06.100. [[CrossRef](#)]
2. Najafi Khoshrodi, M.; Jannati, M.; Sutikno, T. A Review of Wind Speed Estimation for Wind Turbine Systems Based on Kalman Filter Technique. *Int. J. Electr. Comput. Eng.* **2016**, *6*, 1406–1411. [[CrossRef](#)]
3. Xu, L.D.; He, W.; Li, S. Internet of Things in Industries: A Survey. *IEEE Trans. Ind. Inf.* **2014**, *10*, 2233–2243, doi:10.1109/TII.2014.2300753. [[CrossRef](#)]
4. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of Things for Smart Cities. *IEEE Internet Things J.* **2014**, *1*, 22–32, doi:10.1109/JIOT.2014.2306328. [[CrossRef](#)]
5. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376, doi:10.1109/COMST.2015.2444095. [[CrossRef](#)]
6. Berg, J.C.; Miller, K. Sensor Selection for Wind Turbine State Estimation. In Proceedings of the ITEA Live-Virtual-Constructive Conference, Orlando, FL, USA, 1–4 December 2008.
7. Soltani, M.N.; Knudsen, T.; Svenstrup, M.; Wisniewski, R.; Brath, P.; Ortega, R.; Johnson, K. Estimation of Rotor Effective Wind Speed: A Comparison. *IEEE Trans. Control Syst. Technol.* **2013**, *21*, 1155–1167, doi:10.1109/TCST.2013.2260751. [[CrossRef](#)]
8. Ritter, B.; Schild, A.; Feldt, M.; Konigorski, U. The design of nonlinear observers for wind turbine dynamic state and parameter estimation. *J. Phys. Conf. Ser.* **2016**, *753*, 52029. [[CrossRef](#)]
9. Petar, M.; Petrovi, V.; Baotic, M. Dual Kalman Estimation of Wind Turbine States and Parameters. In Proceedings of the International Conference on Process Control, Tatranska Lomnica, Slovakia, 14–17 June 2011; pp. 85–91.
10. Bourlis, D.; Bleijs, J.A.M. A wind speed estimation method using adaptive Kalman filtering for a variable speed stall regulated wind turbine. In Proceedings of the 2010 IEEE 11th International Conference on Probabilistic Methods Applied to Power Systems, Singapore, 14–17 June 2010; pp. 89–94, doi:10.1109/PMAPS.2010.5528980. [[CrossRef](#)]
11. Miranda-Blanco, B.N.; Díaz-Dorado, E.; Carrillo, C.; Cidrás, J. State estimation for wind farms including the wind turbine generator models. *Renew. Energy* **2014**, *71*, 453–465, doi:10.1016/j.renene.2014.05.029. [[CrossRef](#)]

12. Sudev, P.; Anita, J.P.; Sudheesh, P. Nonlinear state estimation of wind turbine. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 13–16 September 2017; pp. 354–358, doi:10.1109/ICACCI.2017.8125866. [CrossRef]
13. Yu, S.; Fernando, T.; Emami, K.; Iu, H.H.C. Dynamic State Estimation Based Control Strategy for DFIG Wind Turbine Connected to Complex Power Systems. *IEEE Trans. Power Syst.* **2016**. [CrossRef]
14. Yu, S.; Emami, K.; Fernando, T.; Iu, H.H.C.; Wong, K.P. State Estimation of Doubly Fed Induction Generator Wind Turbine in Complex Power Systems. *IEEE Trans. Power Syst.* **2016**, *31*, 4935–4944, doi:10.1109/TPWRS.2015.2507620. [CrossRef]
15. Prajapat, G.P.; Bhui, P.; Senroy, N.; Kar, I.N. Modelling and estimation of gear train backlash present in wind turbine driven DFIG system. *IET Gener. Transm. Distrib.* **2018**, *12*, 3527–3535, doi:10.1049/iet-gtd.2017.1377. [CrossRef]
16. Shahriari, S.A.A.; Raoufat, M.; Dehghani, M.; Mohammadi, M.; Saad, M. Dynamic state estimation of a permanent magnet synchronous generator-based wind turbine. *IET Renew. Power Gener.* **2016**, *10*, 1278–1286, doi:10.1049/iet-rpg.2015.0502. [CrossRef]
17. Shahinzadeh, H.; Moradi, J.; Gharehpetian, G.B.; Nafisi, H.; Abedi, M. IoT Architecture for Smart Grids. In Proceedings of the 2019 IEEE International Conference on Protection and Automation of Power System (IPAPS), 8–9 January 2019; pp. 22–30, doi:10.1109/IPAPS.2019.8641944. [CrossRef]
18. Sun, S.L.; Deng, Z.L. Multi-sensor optimal information fusion Kalman filter. *Automatica* **2004**, *40*, 1017–1023, doi:10.1016/j.automatica.2004.01.014. [CrossRef]
19. Deng, Z.L.; Gao, Y.; Mao, L.; Li, Y.; Hao, G. New approach to information fusion steady-state Kalman filtering. *Automatica* **2005**, *41*, 1695–1707, doi:10.1016/J.AUTOMATICA.2005.04.020. [CrossRef]
20. Chen, B.; Hu, G.; Ho, D.W.C.; Zhang, W.A.; Yu, L. Distributed Robust Fusion Estimation with Application to State Monitoring Systems. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *47*, 2994–3005, doi:10.1109/TSMC.2016.2558103. [CrossRef]
21. Tian, T.; Sun, S.; Li, N. Multi-sensor information fusion estimators for stochastic uncertain systems with correlated noises. *Inf. Fusion* **2016**, *27*, 126–137, doi:10.1016/J.INFFUS.2015.06.001. [CrossRef]
22. Chen, B.; Hu, G.; Ho, D.W.; Yu, L. A New Approach to Linear/Nonlinear Distributed Fusion Estimation Problem. *IEEE Trans. Autom. Control* **2019**, *64*, 1301–1308, doi:10.1109/TAC.2018.2849612. [CrossRef]
23. Chen, B.; Ho, D.W.C.; Zhang, W.A.; Yu, L. Networked Fusion Estimation With Bounded Noises. *IEEE Trans. Autom. Control* **2017**, *62*, 5415–5421, doi:10.1109/TAC.2017.2696746. [CrossRef]
24. Ma, J.; Sun, S. Distributed fusion filter for networked stochastic uncertain systems with transmission delays and packet dropouts. *Signal Process.* **2017**, *130*, 268–278, doi:10.1016/J.SIGPRO.2016.07.004. [CrossRef]
25. Froese, M. Talking with Turbines through the Internet of Things. 2016. Available online: <https://www.windpowerengineering.com/wind-farm-networks/talking-turbines-internet-things/> (accessed on 27 March 2019).
26. Masoud Barakati, S. Modeling and Controller Design of a Wind Energy Conversion System Including a Matrix Converter. 2008. p. 264. Available online: <http://hdl.handle.net/10012/3786> (accessed on 27 March 2019).
27. Barakati, S.; Kazerani, M.; Aplevich, J. Maximum Power Tracking Control for a Wind Turbine System Including a Matrix Converter. *IEEE Trans. Energy Convers.* **2009**, *24*, 705–713, doi:10.1109/TEC.2008.2005316. [CrossRef]
28. Ugalde-Loo, C.E.; Ekanayake, J.B.; Jenkins, N. State-Space Modeling of Wind Turbine Generators for Power System Studies. *IEEE Trans. Ind. Appl.* **2013**, *49*, 223–232, doi:10.1109/TIA.2012.2228836. [CrossRef]
29. Ugalde-Loo, C.E.; Ekanayake, J.B. State-space modelling of variable-speed wind turbines: A systematic approach. In Proceedings of the IEEE International Conference on Sustainable Energy Technologies (ICSET), Paris, France, 6–9 December 2010; pp. 1–6, doi:10.1109/ICSET.2010.5684454. [CrossRef]
30. Noor-A-Rahim, M.; Khyam, M.O.; Ali, G.G.M.N.; Liu, Z.; Pesch, D.; Chong, P.H.J. Reliable State Estimation of an Unmanned Aerial Vehicle Over a Distributed Wireless IoT Network. *IEEE Trans. Reliab.* **2019**, 1–9, doi:10.1109/TR.2019.2891994. [CrossRef]
31. Chung, S.-Y.; Forney, G.; Richardson, T.; Urbanke, R. On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit. *IEEE Commun. Lett.* **2001**, *5*, 58–60, doi:10.1109/4234.905935. [CrossRef]
32. Ten Brink, S.; Kramer, G.; Ashikhmin, A. Design of Low-Density Parity-Check Codes for Modulation and Detection. *IEEE Trans. Commun.* **2004**, *52*, 670–678, doi:10.1109/TCOMM.2004.826370. [CrossRef]

33. Ten Brink, S.; Kramer, G. Design of repeat-accumulate codes for iterative detection and decoding. *IEEE Trans. Signal Process.* **2003**, *51*, 2764–2772, doi:10.1109/TSP.2003.818250. [[CrossRef](#)]
34. Noor-A-Rahim, M.; Zhang, N.; Wang, S. Performance estimation of finite-length repeat-accumulate codes. *IET Commun.* **2015**, *9*, 1902–1905, doi:10.1049/iet-com.2014.1193. [[CrossRef](#)]
35. Noor-A-Rahim, M.; Khyam, M.O.; Guan, Y.L.; Ali, G.G.N.; Nguyen, K.D.; Lechner, G. Delay-Universal Channel Coding with Feedback. *IEEE Access* **2018**, doi:10.1109/ACCESS.2018.2853140. [[CrossRef](#)]
36. Chung, S.Y.; Richardson, T.J.; Urbanke, R.L.; Chung, S.Y.; Richardson, T.J.; Urbanke, R.L. Analysis of sum-product decoding of low-density parity-check codes using a {Gaussian} approximation. *IEEE Trans. Inform. Theory* **2001**, *47*, 657–670, doi:10.1109/18.910580. [[CrossRef](#)]
37. Johnson, S.J. *Iterative Error Correction: Turbo, Low-Density Parity-Check and Repeat-Accumulate Codes*; Cambridge University Press: Cambridge, UK, 2010.
38. Simon, D. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control Theory Appl.* **2010**, *4*, 1303–1318, doi:10.1049/iet-cta.2009.0032. [[CrossRef](#)]
39. Faragher, R. Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation [Lecture Notes]. *IEEE Signal Process. Mag.* **2012**, *29*, 128–132, doi:10.1109/MSP.2012.2203621. [[CrossRef](#)]
40. Sun, S.I. Multi-sensor optimal information fusion Kalman filters with applications. *Aerosp. Sci. Technol.* **2004**, *8*, 57–62, doi:10.1016/j.ast.2003.08.003. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).