

Article

Optimal Sensor and Relay Nodes Power Scheduling for Remote State Estimation with Energy Constraint

Yufei Han ¹ , Mengqi Cui ² and Shaojun Liu ^{2,*} 

¹ Department of Automation, University of Science and Technology of China, Auhui 230027, Hefei, China; hanyufei@mail.ustc.edu.cn

² School of Energy and Power, Jiangsu University of Science and Technology, Zhenjiang 212003, Jiangsu, China; cuimengqijust@163.com

* Correspondence: phoenix205@zju.edu.cn

Received: 12 January 2020; Accepted: 14 February 2020; Published: 16 February 2020



Abstract: We study the sensor and relay nodes' power scheduling problem for the remote state estimation in a Wireless Sensor Network (WSN) with relay nodes over a finite period of time given limited communication energy. We also explain why the optimal infinite time and energy case does not exist. Previous work applied a predefined threshold for the error covariance gap of two contiguous nodes in the WSN to adjust the trade-off between energy consumption and estimation accuracy. However, instead of adjusting the trade-off, we employ an algorithm to find the optimal sensor and relay nodes' scheduling strategy that achieves the smallest estimation error within the given energy limit under our model assumptions. Our core idea is to unify the sensor-to-relay-node way of error covariance update with the relay-node-to-relay-node way by converting the former way of the update into the latter, which enables us to compare the average error covariances of different scheduling sequences with analytical methods and thus finding the strategy with the minimal estimation error. Examples are utilized to demonstrate the feasibility of converting. Meanwhile, we prove the optimality of our scheduling algorithm. Finally, we use MATLAB to run our algorithm and compute the average estimation error covariance of the optimal strategy. By comparing the average error covariance of our strategy with other strategies, we find that the performance of our strategy is better than the others in the simulation.

Keywords: state estimation; relay nodes; wireless sensor networks

1. Introduction

Recent years have witnessed a boom of WSNs [1]. In the application of WSNs, sensors' remote state estimation plays a key role. To process the estimation, the sensors in WSNs collect the information of physical phenomena such as temperature and humidity, record them as the "state", run a preprocess for the state, and then send the state information as data packets to a remote estimator via a wireless network. Each procedure of this process has received extensive research from different perspectives [1–14].

In this work, we study the optimal sending strategy for the sensor and the relay nodes in a WSN. Different from most previous works that designed sending strategies for the WSN with only a single sensor [2–8], in this work, the strategy is designed for a WSN with one sensor, one remote estimator, and extra relay nodes in between. The signal sent from the sensor no longer reaches the estimator directly, but is passed on by each relay node before finally reaching the estimator. Therefore, we assume that the sensor and the relay nodes all had the same energy storage for sending signals. The wireless channels links the sensor, the relay nodes, and the estimator. While being passed on, the signals sent by the high power are passed on by the relay nodes successfully, whereas the signals sent by the low

power suffer packet drops. Then, the optimal strategy minimizes the error covariance of the remote estimator given the fixed amount of energy. Before demonstrating the strategy, we introduce the most recent works that are related to similar topics and explain the reasons we study this model.

1.1. Motivation and Background

To stop a high packet drop rate from deteriorating the performance of the remote estimator, relay nodes are added between the sensor and the remote estimator [11–22]. Relay nodes help pass on signals from the sensor to the final estimator and thus break the long wireless communication path into shorter sections that have a smaller packet drop rate. This assertion was motivated by the indication of M.Holland et al. [23] that the larger the distance between the sensor and the estimator, the larger the packet drop rate would be. A recent study regarding state estimation with relay nodes [24] emphasized the stability of the system, whereas we focused on finding the optimal strategy.

Our model extended Shi's model [2]. Shi et al. gave the optimal sending strategy for the WSN that transmitted signals from the sensor to the estimator directly. In [2], the authors revealed that for a finite-time and finite-energy sensor data scheduling problem, sending every signal with a high power that ensures a successful transmission to the remote estimator does not make the best use of the energy. Under the assumption that sending low power signals will bring packet drops, they deduced that the optimal choice was to send high power signals mixed with low power signals and make them distribute as uniformly as possible. In this way, the sending strategy minimizes the average estimation error covariance of the remote estimator.

As was proposed in Asshad's survey [22], in the WSNs with relay nodes, nodes with limited energy need to have a power optimization mechanism. To find the mechanism, in recent years, many papers considered methods concerning state estimation and sensor scheduling. Y. Yao et al. [25] studied the method for sensors' position estimation in an ad hoc WSN with large numbers of sensor nodes and addressed an energy-efficient method. The authors made full use of distance estimation. P.M. Daflapurkar et al. [26] came up with an energy-saving algorithm for cluster routing in the WSN. It focused on designing routing paths between nodes rather than node-to-node sensor scheduling strategies. P. Cheng et al. [24] built three node-to-node data forwarding strategies for state estimation in the WSN with relay nodes. To adjust the trade-off between the energy consumption and the state estimation accuracy, the authors designed an online strategy that compared the error covariance gap between two adjacent nodes to a predefined threshold. R. Zhu et al. [27] used a Markov chain to characterize the delay of the relay nodes of a multi-hop WSN and obtained the necessary and sufficient condition under which the estimation error covariance was stable. L. Yao et al. [28] investigated optimal scheduling for transmission in the WSN with one relay node that was placed in the feedback loop. Apart from the theoretical analysis, F.A. Aderohunmu et al. [14], M. Maggiorotti et al. [13], and M.Rossi et al. [12] also developed the real case applications for the power scheduling problem.

Compared to previous works, our model adds relay nodes in the model of [2], and the way the relay nodes update their state is the same as the local processing and forwarding strategy in [24]. However, different from the work of P. Cheng et al. in [24], we design an offline strategy that controls whether to send a signal or not for each relay node at each time step. Hence instead of a predefined threshold that adjusts the trade-off of energy and state estimation performance, we find an optimal offline sensor scheduling algorithm that achieves the smallest average estimation error covariance with limited energy. The detailed introduction of the problem we solve and the assumptions we adopt for our model are demonstrated as follows.

We considered the sensor power scheduling problem based on the relay system shown in Figure 1. The state information was preprocessed in the dashed line box, and the preprocessed data was sent to Relay Node 1 via a wireless channel that suffered packet drops. Then, the data packets were passed on from one relay node to the next through the wireless channel until they finally reached the remote estimator. For simplicity, we first analyzed the case when there were only two relay nodes in the system. Each relay node's behavior was a combination of the sensor and the remote estimator.

It received data packets as the estimator did, and sent the packets to the next node as the sensor does. As assumed previously, we only considered the energy distributed to send data and did not address the energy requirements for other functions such as detecting signals in the air, receiving the signals, and preprocessing them. Furthermore, the total amount of energy every relay node used for sending signals was equivalent to that of the sensor. Our goal was to find the optimal sending strategy to minimize the accumulation of the average estimation error of the remote estimator. To achieve this goal with the model and the assumptions we made above, we developed a scheduling strategy called the “converted table” method. The challenge was that the error covariance of the remote estimator was the result of two kinds of updates: the sensor-to-relay-node kind and the relay-node-to-relay-node kind. We were able to find the optimal scheduling methods for both kinds of updates, respectively, but when they were combined to produce one error covariance for the estimator, it would be hard to compare two scheduling strategies by subtracting one’s covariance formula representation from another as L. Shi et al. [2] did. Therefore, we used the “converted table” to change the sensor-to-relay-node update to the relay-node-to-relay-node update. Then, with the majorization theory [29], we were able to prove that the overall optimal scheduling strategy for the sensor and relay nodes was the combination of the optimal sensor-to-relay-node scheduling with the optimal relay-node-to-relay-node scheduling. We also discussed the case when the total amount of energy and the time horizon were not infinite and explained that there would be no way for the states or the error estimations of the nodes to obtain an optimal strategy. Therefore, we only focused on the finite-time and finite-energy case and developed algorithms within the scope of our assumptions.

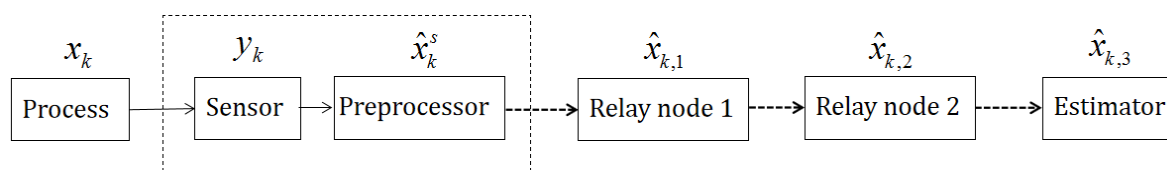


Figure 1. System structure diagram. The dashed-line arrows mark a network with packet drops.

1.2. Main Contributions

1. We found an optimal scheduling algorithm for our model. We proposed an original method called the “converted table” method to obtain the optimal sending sequences for the sensor and the relay nodes. This method not only reduced the estimator’s average error covariance effectively, but also spared us from the cumbersome exhaustive computation for searching for the optimal strategy.
2. We analyzed the reason why the infinite-time case did not exist in our model.
3. We ran simulations for our algorithm and compared its sending sequences’ average error covariance with error generated by other sending strategies. The result of the comparison was empirical proof of the optimality of our algorithm.

1.3. Organization

The remainder of the paper is organized as follows. First, Section 2 introduces the optimal sensor and relay nodes’ data scheduling scheme for the finite-time and limited energy case. Furthermore, it gives the proof for the scheme’s optimality. Then, Section 3 explains the non-existence of the scheduling for the infinite-time and energy case. Next, Section 4 shows the simulation result, which compares our optimal strategy with other strategies. Conclusions and consideration for future work are added in Section 5.

1.4. Notations

\mathbb{R} is the set of real numbers. \mathbb{R}^n is the set of n -dimensional vectors. \mathbb{Z}_+ is the set of non-negative integers. $k \in \mathbb{Z}_+$ is the time index. \mathbb{S}^n is the set of n by n matrices. For $X \in \mathbb{S}^n$, $X > 0$ (and $X \geq 0$)

implies that X is positive definite (and positive semi-definite), and it is represented as $X \in \mathbb{S}_{++}^n$ (and $X \in \mathbb{S}_+^n$). X' indicates the transposition of the matrix X . $I = X^0$ implies the identity matrix. $\mathbb{E}[X]$ is the expectation of a random variable X , and $\mathbb{E}[X|\mathcal{Y}]$ denotes the conditional expectation of X given the event \mathcal{Y} . $Pr[\mathcal{Y}]$ indicates the probability of the event \mathcal{Y} . $Tr(X)$ indicates the trace of the matrix X . For functions $h : \mathbb{S}_+^n \rightarrow \mathbb{S}_+^n$, we let $h^k(X) = h(h^{k-1}(X))$ with $h^0(X) = X$.

2. Optimal Sensor and Relay Nodes' Power Scheduling

Our goal was to find the optimal sensor and relay nodes' sending strategy for the system shown in Figure 1. To follow the precedent, we set the state information as x_k and the preprocessed data \hat{x}_k^s . The dashed arrows imply the wireless communication paths the signals take. We will first demonstrate the sending strategy that optimizes the scheduling for the model without the relay nodes and then generalize this strategy to find the optimal scheduling for the model with the relay nodes.

2.1. The No-Relay-Node Case

This case is equivalent to the case that analyzes the transmission between the sensor and Relay Node 1. Assume that this model is built as a dynamic discrete linear time-invariant system shown as follows:

$$\begin{cases} x_{k+1} = Ax_k + w_k \\ y_k = Cx_k + v_k \end{cases} \quad (1)$$

where $k \in \mathbb{Z}_+$. $x_k \in \mathbb{R}^n$ is the system state at time k and $y_k \in \mathbb{R}^m$ is the sensor's measurement of the system state. $w_k \in \mathbb{R}^n$ and $v_k \in \mathbb{R}^m$ are zero-mean i.i.d. Gaussian noises with covariances $Q \geq 0$ and $R > 0$, respectively. Here, we assume that the pair (A, C) is observable and (A, \sqrt{Q}) is controllable.

At time k , the sensor's measurements of the system states are represented as $\mathcal{Y}_k = \{y_1, \dots, y_k\}$. Then, in the preprocessor shown in Fig.1, the sensor with sufficient computation ability uses \mathcal{Y}_k to generate a minimum mean squared error estimate (MMSE) $\hat{x}_k^s = \mathbb{E}[x_k|\mathcal{Y}_k]$ with the corresponding error covariance $P_k^s = \mathbb{E}[(x_k - \hat{x}_k^s)(x_k - \hat{x}_k^s)'|\mathcal{Y}_k]$. By the standard Kalman filtering [17], if we define the function $h: \mathbb{S}_+^n \rightarrow \mathbb{S}_+^n$ as $h(X) = AXA' + Q$ and $\tilde{g}: \mathbb{S}_+^n \rightarrow \mathbb{S}_+^n$ as $\tilde{g}(X) = X - XC'[CXC' + R]^{-1}CX$ just like the definition shown in [3], then the estimation error covariance P_k^s converges to a constant value \bar{P} exponentially fast, and we denote it as $\lim_{k \rightarrow \infty} P_k^s = \bar{P}$, where \bar{P} is the unique positive semi-definite solution of $\tilde{g} \circ h(\bar{X}) = \bar{X}$. Therefore, for simplicity, we assume that for $\forall k \in \mathbb{Z}_+$, $P_k^s = \bar{P}$. To generalize the model without relay nodes into the model with relay nodes in Figure 1, in the following, we mark the sensor as the zeroth node and Relay Node 1 the first node; thus, we use $P_{k,i}$ to represent the error covariance generated at the i^{th} node at time k , and \hat{x}_k^s is also marked as $\hat{x}_{k,0}^s$. In the precedents where there are no relay nodes, the initial error covariance of the remote estimator is the same as the sensor's. Without this initial error covariance, the remote estimator cannot update its estimation of the sensor locally according to its a priori knowledge of the sensor under a packet drop. Therefore, when there are relay nodes, we assume the initial error covariance of the relay nodes is the same as the estimator, which enables local updates, i.e., $P_{0,i} = P_{0,0} = \bar{P}$.

Lemma 1 ([8]). *The function $h(\bar{P})$ defined above has the following property:*

$$\bar{P} \leq h(\bar{P}) \leq h^2(\bar{P}) \leq \dots \leq h^k(\bar{P}) \leq \dots, \forall k \in \mathbb{Z}_+.$$

Note that for $\forall k \in \mathbb{Z}_+$, we have $h^k(\bar{P}) \geq 0$, so $Tr[h^k(\bar{P})] \geq 0$.

To calculate the average error covariance, first we define the scheduling variable of the i^{th} node $\lambda_{k,i}$ as follows:

$$\lambda_{k,i} = \begin{cases} 1, & \text{if a high power is sent at time } k \\ 0, & \text{if a low power is sent at time } k. \end{cases}$$

Let $T \in \mathbb{Z}_+$ be the time horizon, then the i^{th} node's scheduling sequence is $\theta_i = (\lambda_{1,i} \cdots, \lambda_{T,i})$. Besides, as was assumed in [2], we also assume that at time k ,

$$\text{Packet drop rate } \alpha_k = \begin{cases} 0, & \text{if } \lambda_{k,i} = 1 \\ \alpha, & \text{if } \lambda_{k,i} = 0. \end{cases}$$

Therefore at the Relay Node 1, that is the 1th node's state estimate $\hat{x}_{k,1}$ and the error covariance $P_{k,1}$ are updated as follows:

$$(\hat{x}_{k,1}, P_{k,1}) = \begin{cases} (A\hat{x}_{k-1,1}, h(P_{k-1,1})), & \text{if the packet drops} \\ (\hat{x}_{k,1}^s, P_{k,1}), & \text{if the packet arrives.} \end{cases} \tag{2}$$

where $P_{0,1} = P_{0,0} = \bar{P}$.

Thus, the average error covariance of the state updates in the following way [16]:

$$\mathbb{E}[P_{k+1,0}] = (1 - \alpha_k)P_{0,0} + \alpha_k h(\mathbb{E}[P_{k,0}]). \tag{3}$$

Then, we use the trace of the average expected estimation error covariance to evaluate the system performance, just as the authors in [2,3,17,18] did, i.e.,

$$J_{[i]}(\theta) = 1/T \sum_{k=1}^T \mathbb{E}[P_{k,i}(\theta)]. \tag{4}$$

The goal of the optimal sensor scheduling problem is to optimize the following problem:

Problem 1.

$$\begin{aligned} & \min_{\theta} \text{Tr}(J_{[1]}(\theta)) \\ & \text{s.t. } \sum_{k=1}^T \lambda_{k,0} = m \end{aligned}$$

where m indicates the times that the high power is sent. Though our final goal was different, it was still necessary to work out this problem first. Problem 1 can be solved as a dual problem of Problem 3.1 from [18]. In that paper, the problem was proposed from the perspective of an attacker. However, it can be modified into the optimal sensor power scheduling. For clarity, we transformed the format of the solution of Problem 3.1 in [18] into the solution of our power scheduling problem shown as follows.

Theorem 1. (1) If $m < (1/2)T$, the optimal solution for Problem 1 is the following sequence of θ .

$$\underbrace{(0 \cdots 0)}_l \underbrace{(1 0 \cdots 0)}_p \cdots \underbrace{(1 0 \cdots 0)}_p \underbrace{(1 0 \cdots 0)}_{p+1} \cdots \underbrace{(1 0 \cdots 0)}_{p+1}$$

$$\underbrace{\hspace{10em}}_{mp+m+l-T} \underbrace{\hspace{10em}}_{T-l-mp}$$

where $p = \lceil \frac{T-l-m}{m} \rceil$ and $l = p - 1$ or p ($\lceil x \rceil$ is defined as the smallest integer that is larger than x).

(2) If $m \geq (1/2)T$, an optimal solution for Problem 2.1 is to make sure that the left side and the right side of every zero in θ is one. For instance, if $T = 10, m = 7$, then the sequences in Table 1 are two examples of the optimal scheduling.

Table 1. Two optimal scheduling sequences for Theorem 1.

1	1	0	1	0	1	1	1	0	1
1	0	1	1	1	1	0	1	0	1

Proof of Theorem 1. First, we will show this problem is virtually the dual problem of Problem 3.1 [18] (in [18], the optimal scheduling for the $n > (1/2)T$ case had a minor mistake, and here, we use the corrected format). Problem 3.1 in [18] studied the case when a perfect wireless network suffered from u times of DoS attack over a time window T . It was assumed that if there was no DoS attack, the sensor data packet would arrive at the remote estimator successfully; but if the attacker launched an attack, the packet would drop with probability α^* . Then, the author denoted $\gamma_k = 0$ or 1 as the attacker's decision variable at time k , i.e.,

$$\gamma_k = \begin{cases} 1, & \text{if an attack is launched at time } k \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{Packet drop rate } \alpha_k^* = \begin{cases} 0, & \text{if } \gamma_k = 0 \\ \alpha^*, & \text{if } \gamma_k = 1. \end{cases}$$

By comparing the definitions of γ_k and α_k^* with the definitions of λ_k and α_k , we can see that if we let $m = T - u$ and change $\gamma_k = 1$ into $\lambda_k = 0$, as well as $\gamma_k = 0$ into $\lambda_k = 1$, then the solution to Problem 3.1 in [18] is exactly the solution to Problem 1, which completes the proof. \square

2.2. The Case with Relay Nodes

In this part, we first show how the model in Figure 1 updates its state and error covariance. Next, we introduce the optimal scheduling for this model using the "converted table" method when the packet drop rate is one. Then, we show the way of generalizing this scheduling method for the $\alpha = 1$ case into the $0 < \alpha \leq 1$ case using an algorithm, which generates the optimal scheduling for our model with relay nodes.

First of all, due to the assumption we made in the Introduction part, the way that the relay nodes receive data packets is the same as the estimator does, and the way they send the packets to the next node is the same as the sensor does. Furthermore, whether the node is sending low or high power, the definition of the scheduling variable λ_k and the value of the packet drop rate remain unchanged in the relaying process. Therefore, the only difference between the relay nodes and the sensor is the way they update the estimation. For the relay nodes, the state estimation $\hat{x}_{k,i}$ and the error covariance $P_{k,i}$ are updated as follows [24]:

$$(\hat{x}_{k,i+1}, P_{k,i+1}) = \begin{cases} (\hat{x}_{k,i}^s, P_{k,i}), & \text{if the packet arrives} \\ (A\hat{x}_{k,i}, h(P_{k,i})), & \text{otherwise.} \end{cases} \quad (5)$$

where $P_{0,i} = P_{0,0} = \bar{P}$, for $\forall i \in [2, n - 1]$.

Remark 1. In fact, k refers to the k^{th} signal the sensor has sent, so in this update, k does not change with the increase of i . The reason we call k the "time step" is to follow tradition of previous work such as [18]. Therefore, we can ignore the time delay caused by transmitting data from one node to another and apply (5) to describe the updates between relay nodes.

Besides, according to (5), we obtain the expression of the expectation for the node's error covariance when zero is sent at time k :

$$\mathbb{E}[P_{k,i+1}] = (1 - \alpha)\mathbb{E}[P_{k,i}] + \alpha h(\mathbb{E}[P_{k,i}]). \quad (6)$$

Thus:

$$\mathbb{E}[P_{k,i+1}] - \mathbb{E}[P_{k,i}] = \alpha \{h(\mathbb{E}[P_{k,i}]) - \mathbb{E}[P_{k,i}]\} \geq 0. \quad (7)$$

Hence, we can see that for relay nodes, when the high power is sent, the average error covariance of the following node does not change; and when the low power is sent, the average error covariance increases, as is shown in (7). Now, we consider two cases while searching for the optimal scheduling, that is the $\alpha = 1$ case and the $0 < \alpha < 1$ case, which is an extension of the $\alpha = 1$ case.

2.2.1. Case 1: The Optimal Scheduling When $\alpha = 1$

Now, we show how to minimize the average error covariance of the estimator in Figure 1, that is how to solve the following problem:

Problem 2.

$$\begin{aligned} & \min_{\theta} \text{Tr}(J_{[n]}(\theta)) \\ & \text{s.t. } \sum_{k=1}^T \lambda_{k,i} = m, \quad i = 0, 1, \dots, n-1. \end{aligned}$$

where $\lambda_{k,i}$ refers to the i^{th} node's scheduling variable and $J_{[n]}$ is the estimator's average error covariance when there are $n - 1$ relay nodes functioning between the final estimator and the sensor (for example, in Figure 1, we have $n = 3$) Next, we use a simple example to show the "converted table" method that helps solve this problem.

Example 1. Table 2 lists a strategy that is randomly chosen for each node in Figure 1 at every time step when $T = 6$ and $m = 2$. With the assumption $\alpha = 1$, we can obtain the error covariances accordingly in Table 3.

Table 2. Strategy of each node at each time step.

Time k	1	2	3	4	5	6
θ_0	0	0	1	0	1	0
θ_1	0	0	1	0	1	0
θ_2	0	0	1	0	1	0

Table 3. Error covariance of the strategy in Table 2.

Time k	1	2	3	4	5	6
$P_{k,1}$	$h(\bar{P})$	$h^2(\bar{P})$	\bar{P}	$h(\bar{P})$	\bar{P}	$h(\bar{P})$
$P_{k,2}$	$h^2(\bar{P})$	$h^3(\bar{P})$	\bar{P}	$h^2(\bar{P})$	\bar{P}	$h^2(\bar{P})$
$P_{k,3}$	$h^3(\bar{P})$	$h^4(\bar{P})$	\bar{P}	$h^3(\bar{P})$	\bar{P}	$h^3(\bar{P})$

Now, we can verify that in terms of the states' error covariances, Table 2 equals Table 4 because they have the same $P_{k,i}, i \geq 0$.

Table 4. “Converted table” for Table 2.

Time k	1	2	3	4	5	6
θ_{-2}		1				
θ_{-1}	1	0		1		1
θ_0	0	0	1	0	1	0
θ_1	0	0	1	0	1	0
θ_2	0	0	1	0	1	0

Note that in Table 4, we no longer need to consider the update of (2) in θ_0 's scheduling sequence, because the way of the sensor-to-relay-node update is converted into the way of the relay-node-to-relay-node update. Then, there is only one way of updating the left in this new table, since states updated by (2) are converted to states updated by (5) by replacing the sensor with a different number of imaginary relay nodes that use imaginary strategies θ_{-2} and θ_{-1} . We used these auxiliaries because with only one way left to update the error covariance, we are now able to find the optimal sending strategy for our system. In the following, we give the definition of imaginary relay nodes and imaginary strategies.

Definition 1. *Imaginary relay nodes refer to nodes that do not exist physically, but are come up with to reduce the two ways of updating error covariances of nodes' states into one. Their sending strategies, labeled as $\theta_i, i < 0$, are constructed to make sure after replacing the updating strategy of (2) with that of (5) that the physically-existed nodes hold the unchanged error covariances.*

Our goal is to achieve the minimum $\text{Tr}[J_{[n]}(\theta)]$. To solve this problem, we first assume that in θ_0 , the high power signal ones distributed as uniformly as possible and then find the optimal $\theta_1, \dots, \theta_{n-1}$ under this θ_0 condition. Next, we prove that this $\theta_0, \theta_1, \dots, \theta_{n-1}$ combination is optimal. By Example 1, we have unified the way the error covariance evolves (This method now works when $\alpha = 1$. A more practical way that applies to $0 < \alpha < 1$ will be given later.). $T \times J_{[n]}(\theta)$ is the sum of $P_{k,n}$, and the value of each $P_{k,n}$ only depends on the total number of zeros in the k^{th} column of the converted table above since ones do not change its value. For each column, the starting error covariance is \bar{P} , and for further analysis, we also need the following lemma:

Lemma 2 ([15]). *Let $\lambda_{\min}(A'A)$ be the minimal eigenvalue of $A'A$, then when $\lambda_{\min}(A'A) \geq 1$ and $t \geq 0$, we have:*

$$\text{Tr}(h^{t+2}(\bar{P}) - h^{t+1}(\bar{P})) \geq \text{Tr}(h^{t+1}(\bar{P}) - h^t(\bar{P})),$$

which leads to the following corollary:

Corollary 1. *When $\lambda_{\min}(A'A) \geq 1$, the node should send the m ones (high power signals) at the m time steps that have the m largest error covariances.*

From Lemma 2 and Corollary 1, we can obtain our core theorem:

Theorem 2. *When we have $\lambda_{\min}(A'A) \geq 1$ and $\alpha = 1$, the optimal offline strategy for each node is to make the sensor send θ_0 , which takes the form of the optimal sequences in Theorem 1, and make the rest of the relay nodes follow the strategy in Corollary 1.*

Proof of Theorem 2. To prove that the combination of Theorem 2 and Corollary 1 gives the optimal strategy, first we set up four sets, \mathcal{A} , \mathcal{A}' , \mathcal{B} , and \mathcal{B}' , where \mathcal{A}' is the complement of \mathcal{A} and \mathcal{B}' is the complement of \mathcal{B} . Then, if the scheduling sequence θ_0 follows Theorem 2, we put it in the set \mathcal{A} and represent their relationship as $\theta_0 \in \mathcal{A}$; otherwise, we put θ_0 in \mathcal{A}' and get $\theta_0 \in \mathcal{A}'$. Next, if θ_i follows

Corollary 1, we put it in set \mathcal{B} and represent it as $\theta_i \in \mathcal{B}, i \geq 1$; or else, we put it in set \mathcal{B}' and get $\theta_i \in \mathcal{B}', i \geq 1$. From Lemma 2, we learn that when θ_0 is fixed, the combination of $\theta_0 \in \mathcal{A}$ (or $\theta_0 \in \mathcal{A}'$) and $\theta_i \in \mathcal{B}, i \geq 1$ is better than that of $\theta_0 \in \mathcal{A}$ (or $\theta_0 \in \mathcal{A}'$) and $\theta_i \in \mathcal{B}', i \geq 1$. Therefore, we only need to compare the combination of $\theta_0 \in \mathcal{A}$ and $\theta_i \in \mathcal{B}, i \geq 1$ (or $\theta_i \in \mathcal{B}', i \geq 1$) versus the combination of $\theta_0 \in \mathcal{A}'$ and $\theta_0 \in \mathcal{B}, i \geq 1$ (or $\theta_i \in \mathcal{B}', i \geq 1$) when each $\theta_i, i \geq 1$ in \mathcal{B} (or \mathcal{B}') is fixed.

The differences between the combination of $\theta_0 \in \mathcal{A}$ and $\theta_i \in \mathcal{B}, i \geq 1$ and the combination of $\theta_0 \in \mathcal{A}'$ and $\theta_i \in \mathcal{B}, i \geq 1$ are:

- I. $\theta_0 \in \mathcal{A}'$ will bring more zeros into the imaginary nodes' scheduling $\theta_i, i < 0$;
- II. the number of zeros of $\theta_0 \in \mathcal{A}$ at each time step (i.e., the zeros of each column in the "converted table") is the most evenly distributed among all possible scheduling sequences.

Therefore,

- (i) when we use $\theta_0 \in \mathcal{A}$ and $\theta_i \in \mathcal{B}, i \geq 0$ to schedule the total communication energy in a time period at a length of t , we can assume that after applying the "converted table", we get $n_{1,j}$ zeros (including the ones in $\theta_i, i < 0$) in each column of the table, and $\sum_{j=1}^t n_{1,j} = N_1$;
- (ii) when we use $\theta_0 \in \mathcal{A}'$ for the scheduling and keep all the other conditions the same as those in (i), we will get $n_{2,j}$ zeros in each column, and $\sum_{j=1}^t n_{2,j} = N_2$;
- (iii) according to the previous analysis in I and II, we have $N_1 < N_2$ and:

$$\sum_{j=1}^t |n_{1,j} - N_1/t| < \sum_{j=1}^t |n_{2,j} - N_2/t|;$$

- (iv) if we set $f(x) = \text{Tr}(h^x(\bar{P}))$, then $f(x)$ is convex since we can verify that:

$$\frac{f(x_3) - f(x_2)}{x_3 - x_2} \geq \frac{f(x_2) - f(x_1)}{x_2 - x_1} \tag{8}$$

for $x_3 > x_2 > x_1, \forall x_i \in \mathbb{N}$. Note that though $f(x)$ is not continuous, its domain consists of discrete points that constitute a closed set. Therefore, on such a set, $f(x)$ is still convex.

- (v) Then, the optimization of the combination of $\theta_0 \in \mathcal{A}$ and $\theta_i \in \mathcal{B}, i \geq 1$ can be proven by using the following lemma on majorization theory:

Lemma 3 ([4,29,30]). For $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n, y = (y_1, y_2, \dots, y_n) \in \mathbb{R}^n$, we rearrange the order of the elements and put the i^{th} largest element as $x_{[i]}$, i.e.,

$$\begin{cases} x_{[1]} \geq x_{[2]} \geq \dots \geq x_{[n]} \\ y_{[1]} \geq y_{[2]} \geq \dots \geq y_{[n]}. \end{cases} \tag{9}$$

if the following conditions hold,

$$\begin{cases} \sum_{i=1}^k x_{[i]} \leq \sum_{i=1}^k y_{[i]}, k = 1, 2, \dots, n-1 \\ \sum_{i=1}^n x_{[i]} = \sum_{i=1}^n y_{[i]}. \end{cases} \tag{10}$$

then x is majorized by y .

What is more, x is majorized by y if and only if for all convex functions $\phi: \mathbb{R} \rightarrow \mathbb{R}$,

$$\sum_{i=1}^n \phi(x_i) \leq \sum_{i=1}^n \phi(y_i). \tag{11}$$

When ϕ is non-decreasing, (10) can be relaxed into:

$$\sum_{i=1}^k x_{[i]} \leq \sum_{i=1}^k y_{[i]}, k = 1, 2, \dots, n. \quad (12)$$

(vi) if $N_1 = N_2$ and $\sum_{j=1}^t |n_{1,j} - N_1/t| < \sum_{j=1}^t |n_{2,j} - N_2/t|$, then it is easy to verify that $n_{1,i}$ and $n_{2,i}$ have the relationship just as $x_{[i]}$ and $y_{[i]}$ in (10); but since what we have is $N_1 < N_2$, $n_{1,i}$ and $n_{2,i}$ actually follow (12). Because $f(x) = \text{Tr}(h^x(\bar{P}))$ is convex, from (12) in (v) we get:

$$\sum_{i=1}^t f(n_{1,i}) \leq \sum_{i=1}^t f(n_{2,i}), \quad (13)$$

which implies the relationship of the two sets of strategies' error covariance and thus completes the proof. \square

Remark 2. The case of comparing the combination $\theta_0 \in \mathcal{A}$ and $\theta_i \in \mathcal{B}'$, $i \geq 1$ and the combination $\theta_0 \in \mathcal{A}'$ and $\theta_i \in \mathcal{B}'$, $i \geq 1$ is the same and thus omitted.

2.2.2. Case 2: The Optimal Scheduling Algorithm When $0 < \alpha < 1$

When $0 < \alpha < 1$, the sensor and the relay nodes no longer share the same error covariance update equation. Hence, if we still want to use the "converted table" method, we need to modify θ_i , $i < 0$ in the converted table. First, we use Table 5 to indicate how the error covariance of each node evolves at different time step (when there are two relay nodes).

Table 5. Error covariance of each node at each time step.

Time k	0	1	2	...	T
Node 0 (sensor)	$P_{0,0}$	$P_{1,0}$	$P_{2,0}$...	$P_{T,0}$
Relay Node 1	$P_{0,1}$	$P_{1,1}$	$P_{2,1}$...	$P_{T,1}$
Relay Node 2	$P_{0,2}$	$P_{1,2}$	$P_{2,2}$...	$P_{T,2}$
Node 3 (estimator)	$P_{0,3}$	$P_{1,3}$	$P_{2,3}$...	$P_{T,3}$

Remark 3. The reason we list the $k = 0$ column is to help develop the algorithm to decide the number of imaginary zeros, and the reasons and details will be discussed in the algorithms we are going to use.

From (7), we see the expectation of the relay nodes' error covariance increases with i monotonically, and from (3), we can also deduce that the sensor's error covariance expectation increases with k monotonically. Hence, if we convert the sensor's expectation of error covariance into the relay nodes' by approximation and unify the two kinds of updates, which is shown in Algorithm 1 in detail, then the converted value is unique and still represents the relative magnitude of the original error covariance. We are able to use this kind of "converting" because we do not require the exact value of the error covariance while searching for the optimal strategy; instead, we only need to compare the relative magnitudes of different strategies and obtain the strategy that generates the smallest magnitude. In Algorithm 1, we will use the converted approximation to compare the number of zeros added in θ_i , $i < 0$. After this approximation, the optimization of the $0 < \alpha < 1$ case is exactly the same as the $\alpha = 1$ case. Therefore, finally, we can get the optimal strategy using the method from Section 2.2.1.

Algorithm 1 Optimal Offline Scheduling**Require:**

- 1: The time horizon, T ;
- 2: The total amount of the times of the high power sent by each node, m ;

Ensure:

- 3: The optimal sensor and relay nodes' power scheduling $\theta^* = (\theta_0^*, \dots, \theta_i^*, \dots, \theta_{n-1}^*)$ that minimizes $Tr[J_{[n]}(\theta)]$ at the remote estimator;
- 4: At time $k = 0$ and node $i = 0$, set $P_{0,0} = \bar{P}$, and we have $\mathbb{E}[P_{0,0}] = \bar{P}, \mathbb{E}[P_{0,0}^{mark}] = \bar{P}$;
- 5:
- 6: **while** $i \neq n$ **do**
- 7: $i = i + 1$
- 8: $\mathbb{E}[P_{0,i+1}^{mark}] = (1 - \alpha)\mathbb{E}[P_{0,i}^{mark}] + \alpha h(\mathbb{E}[P_{0,i}^{mark}])$.
- 9: **end while**
- 10: **while** $k \neq T$ **do**
- 11: $k = k + 1$
- 12: $\mathbb{E}[P_{k+1,0}] = (1 - \alpha_k)\bar{P} + \alpha_k h(\mathbb{E}[P_{k,0}])$;
- 13: For each $\mathbb{E}[P_{k,0}]$, find $i \in [0, n]$ such that $\mathbb{E}[P_{0,i}^{mark}] \leq \mathbb{E}[P_{k,0}] < \mathbb{E}[P_{0,i+1}^{mark}]$;
- 14: **if** $\mathbb{E}[P_{1,i}^{mark}] = \mathbb{E}[P_{k,1}]$ **then**
- 15: $e_{k,1} = i$;
- 16: **else**
- 17: $e_{k,1} = i + 0.5$;
- 18: **end if**
- 19: **end while**
- 20: Set $S_i = \{e_{1,i}, \dots, e_{T,i}\}, i \in [0, n]$.
- 21: **while** $i \neq n$ **do**
- 22: $i = i + 1$
- 23: Find the $\theta_i^* = (\lambda_{1,i}^*, \dots, \lambda_{T,i}^*), i \in [0, n - 1]$ that minimizes $\sum_{k=1}^{k=T} P_{k,i+1}$ (the minimization is shown in Algorithm 2);
- 24: **end while**
- 25: At time $k = T$, we obtain θ^* .
- 26: **return** θ^*

We show our method in Algorithm 1 as a summary. To validate this algorithm, there are several points requiring further illustration. Line 8 computes the expectations of the error covariances $\mathbb{E}[P_{1,i}^{mark}]$ for a sequence that is comprised of T zeros. Line 12 computes the expectations of the error covariances $\mathbb{E}[P_{k,1}]$ for the optimal sequence whose ones and zeros are distributed as uniformly as possible as is described in Theorem 1, and these expectations are updated by (3). Line 10 to Line 19 give us the number of imaginary nodes we should add at different time steps k , i.e., $e_{k,1}$, to get the “converted table”. This conversion is achieved by comparing $\mathbb{E}[P_{k,1}]$ and $\mathbb{E}[P_{1,i}^{mark}]$. For each $\mathbb{E}[P_{k,1}]$, Line 13 finds $i \in [0, n]$ such that $\mathbb{E}[P_{0,i}^{mark}] \leq \mathbb{E}[P_{k,1}] < \mathbb{E}[P_{0,i+1}^{mark}]$, and i is the imaginary number of zeros from $\theta_{i-s}, i < 0$ added in the corresponding *Time* k column in the above tables. i only measures how the error covariances are rounded after the conversion. The reason for using Line 14 to Line 18 will be explained afterwards. Line 20 counts the total number of zeros each node has emitted until time k , including those from imaginary nodes. Line 21 to 24 are valid due to the fact that when $0 < \alpha < 1$, the conclusions of Lemma 2 and Corollary 1 still hold, because from (6) and (7), we have:

$$\begin{aligned}
 & Tr(\mathbb{E}[P_{k,i+2}] - \mathbb{E}[P_{k,i+1}]) - Tr(\mathbb{E}[P_{k,i+1}] - \mathbb{E}[P_{k,i}]) \\
 &= Tr(\alpha\{h(\mathbb{E}[P_{k,i+1}]) - \mathbb{E}[P_{k,i+1}]\}) - Tr(\alpha\{h(\mathbb{E}[P_{k,i}]) - \mathbb{E}[P_{k,i}]\}) \\
 &= \alpha Tr(h(\mathbb{E}[P_{k,i+1}]) - h(\mathbb{E}[P_{k,i}]) - Tr(\mathbb{E}[P_{k,i+1}] - \mathbb{E}[P_{k,i}])) \\
 &= \alpha Tr(A(\mathbb{E}[P_{k,i+1}] - \mathbb{E}[P_{k,i}])A') - Tr(\mathbb{E}[P_{k,i+1}] - \mathbb{E}[P_{k,i}]) \\
 &= \alpha Tr((A'A - I)(\mathbb{E}[P_{k,i+1}] - \mathbb{E}[P_{k,i}])).
 \end{aligned} \tag{14}$$

Thus, the rest of the analysis is the same as the case when $\alpha = 1$, and that is why we can use Line 21 to Line 24 in our strategy when $0 < \alpha < 1$. Line 23 is shown in Algorithm 2 in detail. What is more, now we explain the reason for using Line 14 to Line 18. Line 17 helps discriminate $e_{k,i}$, because in this way, if there exists $\mathbb{E}[P_{k_1,0}]$ and $\mathbb{E}[P_{k_2,0}]$ ($k_1 \neq k_2$) that fall in the same $[\mathbb{E}[P_{0,i}^{mark}], \mathbb{E}[P_{0,i+1}^{mark}]]$ range, then the one that happens to equal $\mathbb{E}[P_{0,i}^{mark}]$ will get the priority to have its $e_{k,i}$ enter the set $S^{[m]}$ in Algorithm 2.

Algorithm 2 Line 23 in Algorithm 1.

Require:

- 1: The time horizon, T ;
- 2: The total amount of the times of the high power sent by each node, m ;

Ensure:

- 3: Find the $\theta_i^* = (\lambda_{1,i}^*, \dots, \lambda_{T,i}^*), i \in [0, n-1]$ that minimizes $\sum_{k=1}^{k=T} P_{k,i+1}$;
 - 4: Choose the m smallest $e_{k,i}, k \in [1, T]$ in S_i , and put them in a new set $S^{[m]}$.
 - 5:
 - 6: **for** $k=1:T$ **do**
 - 7: **if** $e_{k,i} \in S^{[m]}$ **then**
 - 8: $\lambda_{k,i+1}^* = 1$
 - 9: $e_{k,i+1} = e_{k,i}$
 - 10: **else**
 - 11: $\lambda_{k,i+1}^* = 0$
 - 12: $e_{k,i+1} = e_{k,i} + 1$
 - 13: **end if**
 - 14: **end for**
 - 15: **return** θ_i^*
-

In summary, the correctness of our algorithms is described in the following way: Algorithm 2 is the core of Algorithm 1, and the majorization theory supports Algorithm 2. We can consider Algorithm 1 as the combination of the converted table method that converted the $0 < \alpha < 1$ case into the $\alpha = 1$ case and the sending strategy that is applied to the $\alpha = 1$ case in Section 2.2.1.

3. The Infinite Time and Energy Case

Now, we know that the updates of error covariances in our system can all be converted into the form described by (6), so a clearer discussion on the non-existence of optimization for the infinite time and infinite energy budget case for our system can be conveyed. Opposite to our conclusion, the authors found the optimal sensor scheduling strategy for the infinite case in the WSN with no relay nodes by using the steady-state analysis of the Markov chain in [7]. The reason is that we use (6) to update the error covariance, whereas they used (3). As a result, their states of the average error covariances were recurrent, yet our control for the nodes can only bring transient states for the covariance because the error covariance will only grow larger and never reduce. Hence, in our model, we cannot use any scheduling sequence that forms an optimal distribution or any distribution for the error covariance as a final stable state. Therefore, in this work, we only discuss the finite time and finite energy case.

4. Illustrative Examples

Here, we give the example for our models and algorithms. Assume the system parameters are defined as follows:

$$A = \begin{pmatrix} 2 & 0.6 \\ 0.5 & 2 \end{pmatrix}, P = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix},$$

$$Q = \begin{pmatrix} 0.3 & 0.3 \\ 0.3 & 0.3 \end{pmatrix}, C = \begin{pmatrix} 1 & 2 \end{pmatrix},$$

$$R = 1, T = 6, m = 2, n = 3, \alpha = 0.9,$$

where m is the times of high power signals one node can send in Problem 2 and n is the number of nodes. Note that according to the previous theorems, A has to satisfy $\lambda_{\min}(A'A) \geq 1$. Then, with our algorithm, we get the optimal sensor and relay nodes' scheduling sequences θ_0 , θ_1 , and θ_2 , respectively, in Table 6.

Table 6. Strategy of the sensor and two relay nodes in 6 time steps.

Time k	1	2	3	4	5	6
θ_0	0	0	1	0	1	0
θ_1	0	1	0	0	0	1
θ_2	1	0	0	1	0	0

To show that our strategy is better than others, we first compared the error covariance of our strategy to the error covariances from another 10 randomly generated strategies. The results of the simulations are shown in Figure 2, and the implications of the labels are listed below:

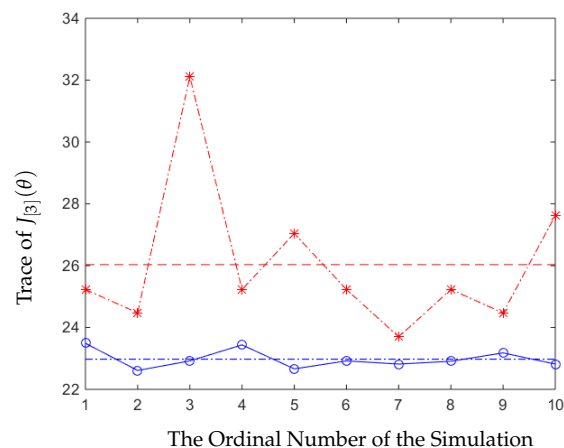


Figure 2. Comparing the error covariance of our strategy with the random strategies.

1. The y -axis refers to $Tr[J_{[n]}(\theta)]$, and the x -axis implies the ordinal number of time the simulation runs.
2. The circled line is the expected value of the average trace of the error covariance generated by our optimal strategy, and the dashed line that goes across it is the average value.
3. The star-marked line is the value of the 10 average traces of the error covariance generated by random strategies, and the dashed line is their average value.

Next, we compared our optimal strategy with the strategy that every node was sending signals using their optimal stationary strategy mentioned in [2], i.e., the nodes all sent high power signals as uniformly as possible. Then, we obtained the following result in Figure 3:

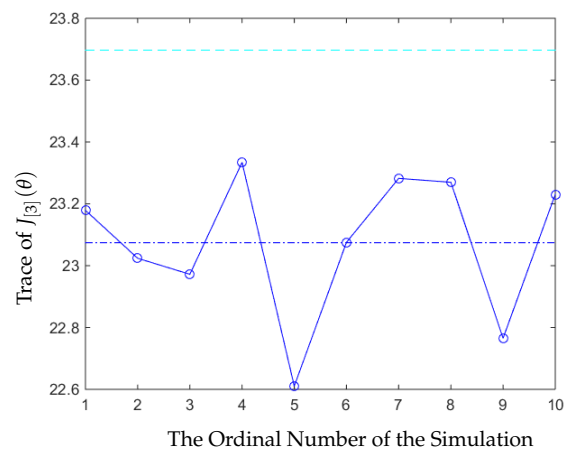


Figure 3. Comparing the error covariance of our strategy with the optimal stationary strategy.

The labels in Figure 3 are as follows:

1. The y -axis, the x -axis, and the circled line are the same as these in Figure 2.
2. The dashed cyan line is the value of the average trace of the error covariance generated by the optimal stationary strategy.

From Figure 2, we can see that in this experiment, our strategy was always better than other strategies, which were picked randomly, and the trace of the average value of the error covariance of our strategy was also smaller. In Figure 3, we can see that the trace of the error covariance of the remote estimator was a fixed value when the sensor and the relay nodes all applied the optimal stationary scheduling to send the signals. The trace of the average value of the error covariance of our strategy was smaller than the optimal stationary strategy, so our strategy was better.

5. Conclusions and Future Work

We discussed the relay node structure for the sensor and relay nodes power scheduling problem that aims to optimize the remote state estimation under the communication energy constraint over a finite time horizon. We came up with the optimal scheduling strategy under the condition $\lambda_{\min}(A'A) \geq 1$ and proved the optimality of our scheduling strategy. Examples and simulations were provided to show that this strategy had better performance than other strategies. In the future, we will try to extend our algorithm to an online strategy that could adjust its scheduling due to feedback from the remote estimator. Furthermore, it would be interesting to consider the case when the relay nodes cannot only pass on information to the next node, but also send signals to its farther neighbors.

Author Contributions: Conceptualization, S.L.; methodology, Y.H.; investigation, Y.H. and M.C.; writing, original draft preparation, Y.H.; writing, review and editing, M.C. and S.L.; supervision, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financed by the Natural Science Foundation of Jiangsu (BK20170580).

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

References

1. Hespanha, J.P.; Naghshtabrizi, P.; Xu, Y. A Survey of Recent Results in Networked Control Systems. *Proc. IEEE*, **2007**, *1*, 138–162. [[CrossRef](#)]
2. Shi, L.; Cheng, P.; Chen, J. Sensor data scheduling for optimal state estimation with communication energy constraint. *Automatica* **2011**, *8*, 1693–1698. [[CrossRef](#)]
3. Shi, L.; Johansson, K.H.; Li, Q. Time and event-based sensor scheduling for networks with limited communication resources. *IFAC Proc. Volumes* **2011**, *1*, 13263–13268. [[CrossRef](#)]
4. Yang, C.; Yang, W.; Shi, H. Majorization theory in sensor scheduling. In Proceedings of the 2015 54th IEEE Conference on Decision and Control (CDC), Osaka, Japan, 15–18 December 2015; pp. 3063–3068.

5. Shi, L.; Xie, L. Optimal Sensor Power Scheduling for State Estimation of Gauss–Markov Systems Over a Packet-Dropping Network. *IEEE Trans. Signal Process.* **2012**, *5*, 2701–2705. [[CrossRef](#)]
6. Gao, X.; Akyol, E.; Basa, T. On remote estimation with multiple communication channels. In Proceedings of the 2016 American Control Conference (ACC), Boston, MA, USA, 6–8 July 2016.
7. Ren, Z.; Cheng, P.; Chen, J.; Shi, L.; Zhang, H. Dynamic sensor transmission power scheduling for remote state estimation. *Automatica* **2014**, *4*, 1235–1242. [[CrossRef](#)]
8. Shi, L.; Cheng, P.; Chen, J. Optimal Periodic Sensor Scheduling With Limited Resources. *IEEE Trans. Automat. Contr.* **2011**, *9*, 2190–2195. [[CrossRef](#)]
9. Anastasi, G.; Conti, M.; Di Francesco, M.; Passarella, A. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.* **2009**, *5*, 537–568. [[CrossRef](#)]
10. Raghunathan, V.; Schurgers, C.; Sung, P.; Srivastava, M.B. Energy-aware wireless microsensor networks. *IEEE Signal Process. Mag.* **2002**, *5*, 40–50. [[CrossRef](#)]
11. Li, H.; Jaggi, N.; Sikdar, B. Relay Scheduling for Cooperative Communications in Sensor Networks with Energy Harvesting. *IEEE Trans. Wirel. Commun.* **2011**, *9*, 2918–2928. [[CrossRef](#)]
12. Rossi, M.; Rizzon, L.; Fait, M.; Passerone, R. Energy Neutral Wireless Sensing for Server Farms Monitoring. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2014**, *4*, 324–334. [[CrossRef](#)]
13. Brunelli, D.; Maggiorotti, M.; Benini, L.; Bellifemine, F.L. Analysis of Audio Streaming Capability of Zigbee Networks. In *European Conference on Wireless Sensor Networks*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 189–204.
14. Aderohunmu, F.A.; Paci, G.; Brunelli, D.; Deng, J.D.; Benini, L.; Purvis, M. An Application-specific Forecasting Algorithm for Extending WSN Lifetime. In proceedings of the 2013 IEEE International Conference on Distributed Computing in Sensor Systems, Cambridge, MA, USA, 20–23 May 2013; pp. 374–381.
15. Qin, J.; Li, M.; Shi, L.; Kang, Y. Optimal Denial-of-Service Attack Energy Management over an SINR-Based Network. *arXiv* **2018**, *10*, arXiv:1810.02558.
16. Li, Y.; Shi, L.; Cheng, P.; Chen, J.; Quevedo, D.E. Jamming Attacks on Remote State Estimation in Cyber-Physical Systems: A Game-Theoretic Approach. *IEEE Trans. Automat. Contr.* **2015**, *10*, 2831–2836. [[CrossRef](#)]
17. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, 35–45. [[CrossRef](#)]
18. Zhang, H.; Cheng, P.; Shi, L.; Chen, J. Optimal Denial-of-Service Attack Scheduling With Energy Constraint. *IEEE Trans. Automat. Contr.* **2015**, *11*, 3023–3028. [[CrossRef](#)]
19. Li, J.; Liu, W.; Wang, T.; Song, H.; Li, X.; Liu, F.; Liu, A. Battery-Friendly Relay Selection Scheme for Prolonging the Lifetimes of Sensor Nodes in the Internet of Things. *IEEE Access* **2019**, *5*, 33180–33201. [[CrossRef](#)]
20. Luo, J.; Hu, J.; Wu, D.; Li, R. Opportunistic Routing Algorithm for Relay Node Selection in Wireless Sensor Networks. *IEEE Trans. Ind. Inf.* **2015**, *2*, 112–121. [[CrossRef](#)]
21. Wang, W.; Srinivasan, V.; Chua, K.-C. Extending the Lifetime of Wireless Sensor Networks Through Mobile Relays. *IEEE/ACM Trans. Networking* **2008**, *10*, 1108–1120. [[CrossRef](#)]
22. Asshad, M.; Khan, S.A.; Kavak, A.; Kucuk, K.; Msongaleli, D.L. Cooperative communications using relay nodes for next-generation wireless networks with optimal selection techniques: A review. *IEEE Trans. Electr. Electron. Eng.* **2019**, *4*, 658–669. [[CrossRef](#)]
23. Holland, M.; Aures, R.; Heinzelman, W. Experimental investigation of radio performance in wireless sensor networks. In proceedings of the 2006 2nd IEEE Workshop on Wireless Mesh Networks, Reston, VA, USA, 25–28 September 2006; pp. 140–150.
24. Cheng, P.; Qi, Y.; Xin, K.; Chen, J.; Xie, L. Energy-efficient data forwarding for state estimation in multi-hop wireless sensor networks. *IEEE Trans. Autom. Control* **2015**, *7*, 1322–1327.
25. Yao, Y.; Zou, K.; Chen, X.; Xu, X. A distributed range-free correction vector based localization refinement algorithm. *Wirel. Netw.* **2016**, *11*, 2667–2680. [[CrossRef](#)]
26. Daflapurkar, P.M.; Pradnya, M.; Gandhi, M.; Patil, B. Tree based distributed clustering routing scheme for energy efficiency in wireless sensor networks. In proceedings of the 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), Chennai, India, 21–22 September 2017; pp. 2450–2456.
27. Zhu, R.; Cheng, P.; Shi, L.; Dai, Y. State Estimation Over Delayed Muthop Network. *IEEE Trans. Autom. Control* **2018**, *1*, 3545–3550.

28. Yao, L.; Chen, C.; Zhu, S.; Guan, X. Sensor scheduling for relay-assisted wireless control systems with limited power resources. *ISA Trans.* **2019**, *5*, 246–257.
29. Hardy, G.; Littlewood, J.E.; Polya, G. Some simple inequalities satisfied by convex functions. *Messenger Math.* **1929**, *58*, 145–152.
30. Kadelburg, Z.; Dukic, D.; Lukic, M.; Matic, I. Inequalities of Karamata, Schur and Muirhead, and some applications. *Teach. Math.* **2005**, *8*, 31–45.



© 2020 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/>).