Energy-neutral and QoS-aware Protocol in Wireless Sensor Networks for Health Monitoring of Hoisting Systems

Houlian Wang, Gongbo Zhou, *Senior Member, IEEE*, Laksh Bhatia,
Zhencai Zhu, Wei Li, Julie A McCann, *Member, IEEE*

***Abstract*—Hoisting equipment is core to many industrial systems and therefore their state of health significantly affects production lines and personnel safety; this is especially important in environments such as coal mines. The health of the hoisting system, can be estimated by deploying energy harvesting wireless sensor nodes that monitor the drum surface stress. In this network of sensor devices, it is very costly to send highly sampled data as it causes radio congestion and consumes energy. However, from our experience of sensing hoist systems, we note that the data observed at the upper surface of the hoist is significantly more indicative of the state of health of the whole system, compared with data sensed at the lower surface. Therefore, we need to take advantage of this to optimise the communications of sensor nodes. However, scarce energy can be collected for these devices from the hoist itself, along with the prioritised Quality of Service (QoS) requirements (throughput, delay) of monitoring signals, raises important challenges for energy management. In this paper, we use Lyapunov optimisation techniques and propose an Energy-neutral and QoS-aware Protocol (EQP), including duty cycling and network scheduling to solve it. Extensive simulations show that EQP helps sensor nodes realize consecutive monitoring, and achieve more than 38% utility gain compared with existing strategies.**

***Index Terms*—Prioritised QoS, energy-neutral operation, heath monitoring, wireless sensor networks, hoisting systems**

I. Introduction

HOISTING systems are core to many industrial environments. In underground mining, the hoisting systems reside in shafts and deliver mined materials, equipment and personnel, and are described as the ”throat” of a mine due to their importance. Although the hoisting system is vital and designed to be stable, sometimes it fails and accidents happen [1]. The major causes of failure are over-winding and jamming, which is caused by the failure of electric control systems and mechanical deformation of a container or an elevator guide rail, respectively. These failures can cause damage to personnel and impact production lines accordingly [2].

Common methods to judge the state of operation of a hoist is to monitor the tension of hoist wire ropes by deploying tension detection devices [3, 4]. These devices are usually put between wire ropes and a hoist container, transmitting tension signals using wireless communications. However, when the mine depth is larger than 500 meters [5], the reliability of transmission may not be guaranteed. Therefore, it is necessary to develop a new approach to monitor the health of the hoisting system to prevent accidents.

Zhou et al. [6] have invented a system to monitor the health of hoisting systems through the real-time detection of drum stress with wireless sensor networks (WSNs). Also, a wind-induced piezoelectric energy harvesting device is designed to be placed perpendicular on the surface of the drum to charge the battery continuously [7].

However, there are still two important requirements for monitoring reliability. Sensor nodes should keep working perpetually during the operation of hoisting activity to avoid missed detection and therefore not detecting events leading to failure, and the data should indeed be valid and timely.

In order to satisfy the aforementioned requirements, we propose an Energy-neutral and Quality of Service-aware (QoS-aware) algorithm (EQP), which is a cross-network and MAC layer protocol. The problem is solved in two steps. First, perpetual operation is provided via energy sustainability which is guaranteed through dynamic duty cycling of the nodes (both sensing and communications). Second, we use Lyapunov optimisation [8, 9] to maximize and prioritise throughput within maximum tolerable data communications delay. A distributed algorithm is presented that supports QoS by varying the packet dropping, prioritised admission, and the power allocation.

The remainder of this paper is organised as follows. Section II introduces the monitoring system. The optimisation problem definition is given in Section III. In Section IV, the monitoring system models are established. The proposed scheduling algorithm is designed in Section V. Simulation results are presented in VI and conclusions are made in Section VII.

Manuscript received April 19, 2005; revised August 26, 2015. This work was supported by National Key Research and Development Program (No. 2016YFC0600905), National Natural Science Foundation of China (No. 51575513 and 61971423), Jiangsu Provincial Natural Science Foundation of China (No. BK20180033), China Scholarship Council (No. 201706420015), as well as A Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).(*Corresponding author: Gongbo Zhou*)

H. Wang, G. Zhou, Z. Zhu, W. Li are with Jiangsu Key Laboratory of Mine Mechanical and Electrical Equipment, China University of Mining and Technology, Xuzhou, Jiangsu 221116, China; with School of Mechanical and Electrical Engineering, China University of Mining and Technology, Xuzhou 221116, China. (email: whl@cumt.edu.cn, gbzhou@cumt.edu.cn, zhuzhencai@cumt.edu.cn, liwei_cmee@163.com.)

L. Bhatia, JA. McCann are with the Department of Computing, Imperial College London, London SW7 2AZ, UK. (email: laksh.bhatia16@imperial.ac.uk, jamm@imperial.ac.uk.)

II. Related Work

The two main challenges in our systems are to provide the QoS guarantees and energy sustainability. In terms of prioritised QoS assurance, the current work can be divided into three main approaches to this multi-attribute problem. We describe the methods according to the protocol layer that impacts.

MAC layer solutions. Here protocols are designed to help nodes obtain channel resources at the link level. Liu et al. proposed a MAC protocol to optimize the transmission order and transmission duration [10] of nodes based on channel status and application context. The proposed scheme jointly exploits channel resource information, traffic and energy to maintain QoS and ensure energy efficency. An IEEE 802.15.4 standard based MAC protocol was proposed to support QoS by adjusting the data type and size [11]. This strategy introduces emergency handling mechanism to ensure QoS requirements of emergency data.

Network layer solutions. Network layer based protocols are used to ensure end-to-end resource negotiation, reservation and reconfiguration so the system can adapt to various application scenarios. Hammoudeh presented a cluster-based route optimisation and load-balancing protocol. This protocol can balance energy efficiency, scalability and robustness [12]. Cheng exploited a geographic opportunistic routing for QoS provisioning with both end-to-end reliability and delay constraints in WSNs [13]. However, this strategy aims to improve wireless performance in dense wireless networks.

Cross-layer solutions. Cross-layered interactions can help networks use information from different layers to achieve better performance. An integrated cross-layer framework was presented, spanning the MAC and network layers. This protocol can satisfy stringent reliability and maximum delay constraints paired with priority demand [14]. However, it requires the involvement of additional sink sources, increasing hardware costs of monitoring. A cross-layered framework was proposed to maintain QoS with an increasing number of video transmitting sources [15], but system needs to know prior knowledge about the location of each other node.

Concerning energy sustainability guarantees, there are two main methods. One way is adaptive duty cycling. E.g., an LMP algorithm was used to optimise the maximum duty cycle based on predicted solar power and monitored energy level [16]. A wake-up variation reduction power management was proposed for Energy Neutral Operation (ENO) in WSNs [17]. The other is adaptive topology control. A cluster head group mechanism was proposed that allows a cluster to use multiple cluster heads to share heavy traffic load [18]. Similarly, Bozorgi et al. proposed a hybrid clustering method combining static and dynamic clustering to solve the energy constraints [19]. However, cluster-based strategies are more suitable for WSNs with a larger number of nodes.

Although the aforementioned work provides innovative methods to ensure QoS in WSNs, these strategies cannot be used directly in health monitoring of hoisting systems. This is because a suitable strategy should consider the following factors.

Unstable energy supply. The piezoelectric energy harvester supplies energy to the sensor device continuously but not uniformly. This is because the amount of harvested energy depends on the spinning speed of the hoist drum which also varies. We cannot guarantee nodes receive the same energy constantly. Some works [10–15] do not support dynamic energy management and so the sensor nodes will eventually fail due to energy shortage.

Difference in data importance and QoS requirements. As the drum spins, it receives significantly more stress at the top. The data received from sensors while positioned at the top also have the highest impact on the health status of the system. These data sampled at the top are more vital and should be prioritised. However, [16–19] do not meet prioritised QoS requirements.

Change in data priority. Different from static networks [20], sensor nodes rotate along with the drum and pass through the top and bottom points. The priority of sampled data also changes correspondingly. Although [8, 21] make contributions to QoS assurance, these works do not take the change of data priority into consideration. [22, 23] support ENO and QoS guarantee but they assume a system without priorities.

Therefore we propose EQP to satisfy the application-specific requirements of a coal mine. The major contributions are as follows:

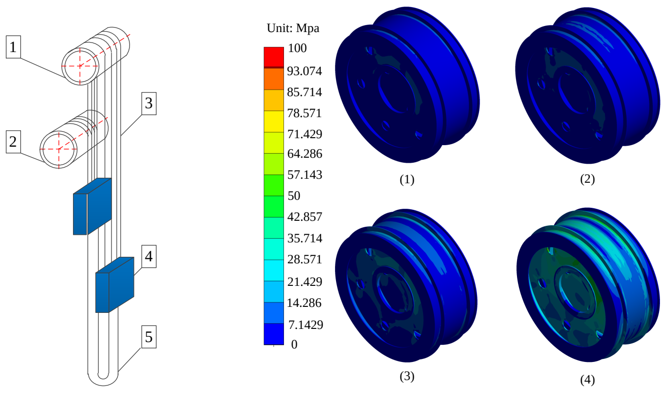
1) An energy-neutral strategy is proposed to ensure energy sustainability, which also helps to monitor the health of the hoisting system.
2) Weight factors are introduced into the Lyapunov optimisation technique. This parameter helps classify the importance of different priority data to accomplish QoS service.
3) A virtual arrival data is added to a data queue to solve the worst delay problem when the data priority changes. This ensures that nodes with older data are not starved of their chance to communicate.

III. Monitoring System

A. Monitoring Principle

A classical structure of a friction hoist is shown in Fig. 1(a). In this hoisting system, the drum is a driving member and its rotation drives the ropes on a drum which raises one container and lowers the other one. The direction of drum rotation is reversible so any of the containers can be raised or lowered. During this process, the load is on the drum and this load is determined by the working conditions. For instance, in normal condition (constant speed and acceleration), the load is only calculated by the force of gravity on the coal and equipment, as well as the product of their mass and acceleration. In contrast, in abnormal conditions (over-winding and jamming), ropes undertake additional force caused by other objects which impede the hoisting process.

Different loads lead to changes in surface stress of the drum, and four different operating conditions are simulated in ANSYS (Fig. 1(b)), including two normal conditions (constant speed and acceleration) and two abnormal conditions (over-winding and jamming). The maximum stress in the jamming



(a) (b)

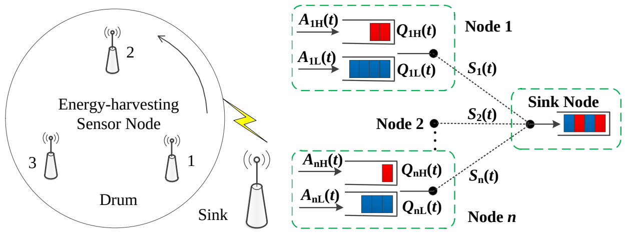
Fig. 1. Monitoring principle (a) Structure of hoisting systems in coal mines: (1) a drum (friction wheel), (2) a guide wheel, (3) wire ropes, (4) hoisting containers, (5) tail ropes (b) Stress nephogram of a drum at four different operating conditions: (1) constant speed (2) acceleration (3) overwinding (4) jamming.

condition is nearly 5 times that of the acceleration mode. The maximum stress for the over-winding condition doubles that in the acceleration mode. This means the difference of stress can be easily recognized by commercial off-the-shelf sensors. Furthermore, the largest stress change of the drum from normal conditions to abnormal conditions happens in the upper half surface and the stress change is less significant in the lower half surface. Hence, the data sampled in the upper half surface are more useful to detect faults than the data sampled in the lower half. This motivates the need to set different sampling and communication requirements and assign them different priorities. We define the High Priority Zone (HPZ) as the area above the horizontal line (i.e. the upper half surface), and the data sampled in this zone as High Priority (HP) data. Similarly, the Low Priority Zone (LPZ) is the area below the horizontal line (i.e. lower half surface), and data sampled there is defined as Low Priority (LP) data.

B. Monitoring Network Architecture

Based on the simulation results of the drum surface stress, we design a many-to-one network. This can be seen in Fig. 2(a). In this figure, the sensor nodes are uniformly installed on the surface of a spinning drum. Each node is equipped with a wind-induced piezoelectric energy harvesting device, a strain gauge sensor, and an angle detection sensor. The strain gauge sensor is used to monitor stress. The angle detection sensor is used to calculate its current location and decide whether the sampling data is in HPZ or LPZ. All sensor nodes are equipped with radio transceivers that can send and receive on one channel at a time. We use a single-hop network architecture in our system where the sink can receive data from only one node at a time.

The network topology is described in Fig. 2(b), where the monitoring network consists of a set of sensor nodes $\mathcal{N} = \{1, 2, ..., n\}$ and a set of sink nodes $\mathcal{D}$. Let $\mathcal{L} = \{1, 2, ..., l\}$ represent the set of all wireless links. We assume that the wireless channel remains unchanged in each time slot and channel state $s_n(t)$ varies independently in every time slot.



(a) (b)

Fig. 2. Monitoring network (a) The deployment of network nodes on the surface of a drum (b) Dynamic queue model in the monitoring network.

Each node has two queues, a HP queue and a LP queue. HP data are placed in the HP queue and LP data are placed in the LP queue. We denote C to be the set of priority classes, including a higher class H and a lower class L, i.e., $c \in C = \{H, L\}$. The network operates in a finite-horizon period consisting of discrete time slots $t \in \{1, 2, 3..., T\}$. We assume reliable links where if data is transmitted once, sink nodes receive it successfully and there is no need for retransmissions. Let Δt represent a time slot interval. Some key notations of symbols are listed in Tab. I.

TABLE I
DEFINITION OF THE KEY VARIABLES

Symbols	Description
τ_n^c	End-to-end delay of c-class data at node n
$d_n^c(t)$	The number of dropped c-class data at node n due to maximum delay tolerance
$p_t(t)$	Transmission power at time slot t
$e_n(t)$	The power of harvested energy at node n at time slot t
$h_n(t)$	The amount of harvested energy at node n at time slot t
p_{sen}	Sensing power of sensor nodes
p_{slp}	Sleep power of sensor nodes
$p_n(t, D(t))$	The amount of consumed energy when the duty cycle is $D(t)$
$\mu_n^c(t)$	Service rate of c-class data at node n at time slot t
$\widetilde{\mu}_n^c(t)$	The expected service rate of class c at node n at time slot t
$R_n^c(t)$	The real arrival of c-class data to a queue at node n at time slot t
$Rvir_n^c(t)$	Virtual arrival of c-class data to a queue at node n at time slot t
w^c	The weight factor of c-class data
ψ^c	Priority control evaluation factor of c-class data
$G_n(t)$	Power allocation evaluation factor at sensor node n

IV. PROBLEM DEFINITION

Our objective is to maximize prioritised throughput within a tolerable delay while ensuring nodes survive. We get the following optimisation problem:

$$\underset{D(t),d_n^c(t),p_t(t)}{\textbf{Maximize:}} \quad util = \frac{1}{T}\sum_{t=0}^{T-1}\sum_{n=1}^{N}(\varrho^H\mu_n^H + \varrho^L\mu_n^L) \tag{1}$$

$$\textbf{Subject to:} \quad \varrho^H \geqslant \varrho^L \tag{2}$$

$$\tau_n^c \leqslant \tau_{max}^c \tag{3}$$

$$d_n^c(t) \leqslant d_{max}^c \tag{4}$$

$$E_n(t) \geqslant E_{min} \tag{5}$$

$$p_t(t) \in \{0, p_{max}\} \tag{6}$$

In the above formulation, ϱ^H and ϱ^L represents the weight factor for actual service rate μ^H and μ^L of HP and LP data respectively. We define the utility *util* as the accumulated sum of the weighted service rate. Constraint (2) ensures that throughput for HP data has a higher priority than the throughput for LP data. Constraint (3) ensures that data delay τ_n^c is less than the maximum delay τ_{max}^c for c-class. Constraint (4) guarantees the dropped data $d_n^c(t)$ is less than the maximum dropped value $d_{max}^c(t)$. Constraint (5) ensures that sensor nodes are energy neutral, i.e., the current energy level $E_n(t)$ should be larger than the minimum energy requirement E_{min}. Constraint (6) explains the range of transmitting power, and the maximum transmission power p_{max} is constant here.

V. System Model

In the following section, an energy model and a network queue model are designed. We derive the channel state $s_n(t)$ from the Received Signal Strength Indication (RSSI). We also assume senor nodes have enough memory to store the sampled data.

A. Energy Model

In this section, we describe the energy model which includes energy harvesting, consumption and storage.

Energy is harvested from piezoelectric vibration energy which converts ambient vibration energy into electric energy. The amount of generated energy depends on the rotation speed of the drum. We conducted energy harvesting experiments on a drum test-bed (Fig. 3(b)). The results of our experiments are shown in Fig. 3(a). The data points obtained can be curve fitted by using the least-squares approximation method,

$$e_n(t) = 1.017 \times 10^{-9} v(t)^{5.686} \tag{7}$$

where $e_n(t)$ is the harvesting power and $v(t)$ is spinning speed. The amount of harvested energy is $h_n(t) = e_n(t) \cdot \Delta t$.

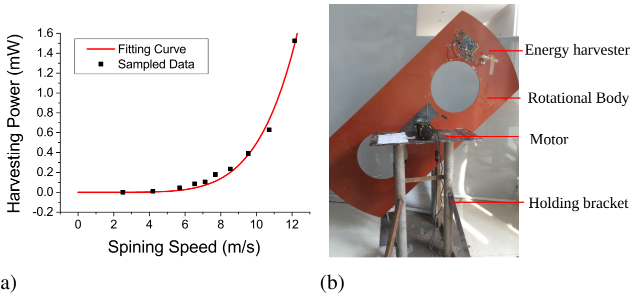


Fig. 3. Energy harvesting experiments (a) Relationship of harvesting power versus spinning speed of a drum (b) Test bed.

Most of the energy is spent in sensing and transmission whereas the node spends most of its time in sleep mode. We denote p_{sen}, p_t and p_{slp} as the power prices for sensing, transmission, and sleep. Duty cycle is represented as $D(t)$. Therefore the consumed energy is

$$p_n(t, D(t)) = [p_{sen} + p_t(t)] \cdot D(t) \cdot \Delta t + p_{slp} \cdot [1 - D(t)] \cdot \Delta t. \tag{8}$$

The remaining energy of sensor node n is represented by $E_n(t)$, and it is given by:

$$E_n(t+1) = E_n(t) - p_n(t) + h_n(t) \tag{9}$$

where $E_n(t)$ should be less than C_b, and C_b is the capacity of energy storage devices.

B. Network Queue Model

The first queue is $Q_n^c(t)$, representing the real c-class data queue backlog at node n in the network, where $Q_n^c(0) = 0$. Let $R_n^c(t)$ represent the number of c-class sampling data at time slot t, and it is subjected to $0 \leqslant R_n^c(t) \leqslant R_{max}^c$, where R_{max}^c is the maximum sampling rate for c-class data. $R_n^c(t)$ is also the data arrival rate to $Q_n^c(t)$. When sensor nodes are located in HPZ, the sampled data is represented by $R_n^H(t)$. On the contrary, $R_n^L(t)$ is the data sampled in LPZ. All the sampled data will be stored in their respective queues. Let $\mu_n(t) = \widehat{\mu}_n(p_t(t), s_n(t))$ represent service rate on a channel; $\widehat{\mu}_n(p, s)$ is a function of transmission power and channel state. The service rate of c-class data satisfies $0 \leqslant \mu_n^c(t) \leqslant \mu_n(t)$. $d_n^c(t)$ is the number of data that can be dropped in slot t. The queue dynamics is given by

$$Q_n^c(t+1) = \max[Q_n^c(t) - d_n^c(t) - \mu_n^c(t), 0] + R_n^c(t). \tag{10}$$

The second queue is a virtual queue. We define $Z_n^c(t)$ for c-class data [8] to ensure that the worst-case delay is bounded. Our queue dynamics are expressed as follows:

$$Z_n^c(t+1) = \begin{cases} 0, & Q_n^c(t+1) = 0 \ \& \ PosChange = 1 \\ \max[Z_n^c(t) + \epsilon^c - d_n^c - \mu_n^c, 0], & \text{else} \end{cases} \tag{11}$$

with $Z_n^c(0) = 0$ for all n, and $PosChange$ is an indicator that is 1 if a node moves from HPZ to LPZ or from LPZ to HPZ. ϵ^c is a preset constant, satisfying $0 \leqslant \epsilon^c \leqslant R_{max}^c$.

According to [8], suppose a scheduling algorithm can guarantee $Q_n^c(t) \leqslant Q_{max}^c$ and $Z_n^c(t) \leqslant Z_{max}^c$ for all $t \in \{0, 1, 2 ...\}$. Then, the worst-case delay of non-dropped c-class data is bounded by the constant

$$t_{max}^c = \left\lceil \frac{Q_{max}^c + Z_{max}^c}{\epsilon^c} \right\rceil, \tag{12}$$

where $\lceil x \rceil$ denotes the smallest integer that is greater than or equal to x.

However, in this paper, the formula (12) needs to be modified to help the dropped data, as there are some special cases especially when data priority changes. When a node is crossing the horizontal line from LPZ to HPZ, the sampling data will change their priority from low to high. Hence, these data will be put in the HP queue while the LP queue will have no arrivals. Once the data in the LP queue are chosen to be sent and there are still some data left in the LP queue, $Q_n^L(t)$ decreases and only $Z_n^L(t)$ increases due to lack of arrival data. As a consequence, $\lceil (Q_n^L(t) + Z_n^L(t))/\epsilon^L \rceil$ may be constant for certain time slots. The remaining data in the LP queue will not be dropped in time, since $\lceil (Q_n^L(t) + Z_n^L(t))/\epsilon^L \rceil$ is already larger than the real delay, which adds additional delay to the items in queue.

To solve this problem, we use a virtual arrival $Rvir_n^c(t)$ which is added to maintain the increase in the queue backlogs, but in fact, there are no real data in the actual queues. $Qvir_n^c(t)$ is represented as the virtual queue backlog, which is the third queue. The dynamic is as follows:

$$Qvir_n^c(t+1) = \begin{cases} Q_n^c(t+1) + Rvir_n^c(t), & Q_n^c(t) \neq 0 \\ 0, & \text{else} \end{cases} \tag{13}$$

where $Rvir_n^c(t)$ satisfies

$$Rvir_n^c(t) = \begin{cases} 2u_{max}^c, & R_n^c(t) = 0, Q_n^c(t) \neq 0 \\ 0, & \text{else.} \end{cases} \tag{14}$$

where u_{max}^c is the maximum service rate for c-class data. Therefore, formula (12) can be changed as

$$t_{max}^c = \left\lceil \frac{Qvir_{max}^c + Z_{max}^c}{\epsilon^c} \right\rceil . \tag{15}$$

The reason why we set $Rvir_n^c$ as $2u_{max}^c$ is that $Qvir_{max}^c + Z_{max}^c$ will not decrease in this setting as the maximum decrease for $Qvir_{max}^c + Z_{max}^c$ is $2u_{max}^c$, which guarantees out-of-time data can be dropped according to the formula (15). Otherwise, once $Qvir_{max}^c + Z_{max}^c$ decreases or remains constant, it will take the remaining data more time to accumulate to a certain level for the data to be dropped or sent, which will increase its delay.

VI. EQP Algorithm

We will solve the proposed problem in two steps: duty cycle management and network scheduling. Duty cycle management is used to maximize the number of data that can be sampled while avoiding energy deficiency. Based on these sampled data, network scheduling is used to maximize the number of transmitted data while satisfying the QoS requirements.

A. Duty Cycle Adjustment

The optimisation problem is defined as follows:

$$\underset{D(t)}{\textbf{Maximize:}} \quad \sum_t D(t) \tag{16}$$

$$\textbf{Subject to:} \quad D_{min} \leqslant D(t) \leqslant D_{max} \tag{17}$$

$$\text{Constraint (5)}$$

where D_{min} is the minimum duty cycle and D_{max} is the maximum duty cycle. The constraint (17) guarantees that duty cycle is in a certain range. One simple way for this problem is to maintain the stored energy around the initial energy level by changing duty cycle dynamically. Following this idea, a sub-optimal solution is presented in Algorithm 1 to achieve ENO.
where,

$$\Delta D(t) = \frac{\{E_n(t) + h_n(t) - E_n(0)\}/\Delta t - p_{slp}}{p_{sen} + p_{max} - p_{slp}} \tag{18}$$

Our proposed algorithm handles two cases. The first case is when the battery energy is lower than the minimum energy required for work. In that case, the duty cycle is set to zero, which means the node is sleeping and is waiting for enough energy to be harvested. The second case is when the battery energy is higher than the minimum energy requirement. This case deals with two further conditions. One condition is that the battery energy is higher than the initial energy. The strategy is to increase the duty cycle to consume excess energy. The other condition is that the battery energy is lower than the initial value, where the duty cycle is reduced based on an energy deficit. In both conditions, duty cycling is chosen to avoid overuse.

Algorithm 1 Duty Cycle Assignment

```
Initialize
While ($t \leqslant T$) do
  If ($E_n(t) < E_{min}$)                                   /*case 1*/
    $D(t+1) = 0$
  Else                                                       /*case 2*/
    If ($E_n(t) \geqslant E_n(0)$)                         /*case 2.1*/
      $D(t+1) = \min[D_{max}, (D(t) + \Delta D(t))]$
      If $E_n(t) + h_n(t) - p_n(t, D(t+1)) < E_{min}$
        If $E_n(t) + h_n(t) - p_n(t, D_{min}) < E_{min}$
          $D(t+1) = 0$
        Else
          $D(t+1) = D_{min}$
        End if
      End if
    Else                                                     /*case 2.2*/
      $D(t+1) = \max[D_{min}, (D(t) + \Delta D(t))]$
      If $E_n(t) + h_n(t) - p_n(t, D(t+1)) < E_{min}$
        If $E_n(t) + h_n(t) - p_n(t, D_{min}) < E_{min}$
          $D(t+1) = 0$
        Else
          $D(t+1) = D_{min}$
        End if
      End if
  End if
End while
```

B. Network Scheduling Design

A Lyapunov optimisation method is used in this section and the control strategy is developed by solving 3 sub-problems.

1) Lyapunov Optimisation: Now denote $\boldsymbol{U}(t) \triangleq [w^c \boldsymbol{Qvir}^c(t); w^c \boldsymbol{Z}^c(t)]$ as a collective vector of all $w^c Q_n^c(t)$ and $w^c Z_n^c(t)$ weighted queues. Then we have a Lyapunov function as follows:

$$L(\boldsymbol{U}(t)) \triangleq \frac{1}{2} \sum_{n=1}^{N} \sum_c \{[w^c Qvir_n^c(t)]^2 + [w^c Z_n^c(t)]^2\}.$$

Theorem 1: (1-slot drift) we define 1-slot conditional Lyapunov drift as $\Delta L(\boldsymbol{U}(t))$. For all t, all $\mathbf{U}(t)$, we have,

$$\begin{aligned} \Delta L(\boldsymbol{U}(t)) \leqslant & B - \sum_{n=1}^{N} \sum_c {w^c}^2 \mu_n^c [Qvir_n^c(t) + Z_n^c(t)] \\ & - \sum_{n=1}^{N} \sum_c {w^c}^2 d_n^c(t) [Qvir_n^c(t) + Z_n^c(t)] \\ & + \sum_{n=1}^{N} \sum_c {w^c}^2 \{[Rvir_n^c(t) + R_n^c(t)] \\ & \cdot Qvir_n^c(t)\} + \sum_{n=1}^{N} \sum_c {w^c}^2 \epsilon^c Z_n^c(t) \end{aligned} \tag{19}$$

where B is a constant defined as

$$\begin{aligned} B = & \frac{1}{2} \sum_{n=1}^{N} \{[{w^H}^2 + {w^L}^2]\{2[\mu_{max} + d_{max}]^2 \\ & + {\epsilon_{max}}^2 + [R_{max} + Rvir_{max}]^2\}. \end{aligned} \tag{20}$$

μ_{max}, d_{max}, R_{max}, $Rvir_{max}$ are the maximum service rate, dropping rate, real arrival rate, virtual arrival rate for all c-class data, respectively.

Proof: see Appendix A and queue stability is also proofed in Appendix B [1]. □

[1] https://www.dropbox.com/s/kw0ol6n7wxuqyjy/Appendix.pdf?dl=0

2) Control Strategy: We now design a distributed algorithm to minimize the right-hand side of (19) subject to listed constraints.

1. Dropping Control

$$\underset{d_n^c(t)}{\textbf{Maximize:}} \quad \sum_{n=1}^{N} w^{c2} d_n^c(t)[Qvir_n^c(t) + Z_n^c(t)] \tag{21}$$

$$\textbf{Subject to:} \quad 0 \leqslant d_n^c(t) \leqslant d_{max}^c \tag{22}$$

A sub-optimal strategy is adopted here: only data exceeding or potentially exceeding delay constraints will be dropped, which will also minimize the number of dropped data and help maximize the throughput.

$$d_n^c(t) = \begin{cases} 0, & \left\lceil \dfrac{Qvir_{max}^c + Z_{max}^c}{\epsilon^c} \right\rceil \leqslant \tau_{max}^c \\ d_{max}^c, & \text{else} \end{cases} \tag{23}$$

2. Priority Control

Based on the optimisation problem (1), we still need to decide which priority data should be sent if a node has access to the channel. We need to ensure a high throughput while guaranteeing our delay requirements. To ensure this, we adopt the following method: select the queue which has the maximum product of weight and criticality between classes [21]. Then, admission can be decided based on capacity. The priority control evaluation factor is defined as:

$$\psi^c(t) = \upsilon^c[\tau_{max}^c - (Qvir_n^c(t) + Z_n^c(t))/\epsilon^c], \tag{24}$$

where υ^c is the admission weight factor, $(Qvir_n^c(t)+Z_n^c(t))/\epsilon^c$ is the possible maximum delay of a node n at time slot t, and $\tau_{max}^c - (Qvir_n^c(t) + Z_n^c(t))/\epsilon^c$ is the difference between the delay constraint and the current delay. Since there are only two classes in the network $c \in \{H, L\}$, $\psi^H(t)$ and $\psi^L(t)$ are the two effect factors which can determine admission priority. By usage of the two factors, the expected admission for classes H and L can be decided as follows:

$$\widetilde{\mu}_n^H = \begin{cases} \min(\mu_{max}(s), Q_n^H(t) - d_n^H(t)), & \psi^H(t) \geqslant \psi^L(t) \\ \min(\mu_{max}(s) - \widetilde{\mu}_n^L, Q_n^H(t) - d_n^H(t)), & \text{else} \end{cases} \tag{25}$$

$$\widetilde{\mu}_n^L = \begin{cases} \min(\mu_{max}(s), Q_n^L(t) - d_n^L(t)), & \psi^L(t) > \psi^H(t) \\ \min(\mu_{max}(s) - \widetilde{\mu}_n^H, Q_n^L(t) - d_n^L(t)), & \text{else} \end{cases} \tag{26}$$

where $\mu_{max}(s) = \widehat{\mu}_n(p_{max}, s_n(t))$ and it is the maximum service when the channel state is $s_n(t)$ at node n. Formulas (25) and (26) means if $\psi^H(t) \geqslant \psi^L(t)$, node n will send data in class H first. If there is any surplus capacity, the node will send data from its class L queue. If $\psi^H(t) \leqslant \psi^L(t)$, this is opposite to the aforementioned case.

3. Power Allocation

According to the constraint (6), the value of allowable transmission power is limited, and also the power must be allocated to satisfy that only one node can transmit data to the sink node in each time slot. Therefore, transmission power is allocated according to different priority queue backlogs.

$$\begin{aligned} \underset{p_t(t), \xi_n(t)}{\textbf{Maximize:}} \quad & \sum_{n=1}^{N} \{ w^{H2} \widetilde{\mu}_n^H [Qvir_n^H(t) + Z_n^H(t)] \\ & + w^{L2} \widetilde{\mu}_n^L [Qvir_n^L(t) + Z_n^L(t)] \} \end{aligned} \tag{27}$$

$$\begin{aligned} \textbf{Subject to:} \quad & \sum_{j=1}^{N} \sum_{i=1}^{N} \xi_i(t)\xi_j(t) = 0, \quad i \neq j \\ & p_t(t) \in \{0, p_{max}\} \end{aligned} \tag{28}$$

where the expected service rates $\widetilde{\mu}_n^H$ and $\widetilde{\mu}_n^L$ are allocated based on formulas (25) and (26). In (28), $\xi_n(t)$ is a channel control parameter which is 1, if a channel is selected and the transmission power is allocated as p_{max} else $p_t(t) = 0$. Since only one channel can be accessed in each time slot, the value of a control parameter ξ_n must be chosen from the maximum power allocation evaluation factor, i.e., $G_n(t) = \sum_c w^{c2} \widetilde{\mu}_n^c(t)[Qvir_n^c(t) + Z_n^c(t)]$ among queues. Finally, transmission power and real service rates can be determined by the following rule:

$$\xi_n(t) = \begin{cases} 1, & G_n(t) > G_i(t), i \neq n, i, n \in \mathcal{N} \\ 0, & \text{else} \end{cases} \tag{29}$$

$$p_t(t) = \begin{cases} p_{max}, & \xi_n(t) = 1 \\ 0, & \text{else} \end{cases} \tag{30}$$

$$\mu_n^c(t) = \begin{cases} \widetilde{\mu}_n^c(t), & p_t(t) = p_{max} \\ 0, & \text{else} \end{cases} \tag{31}$$

C. Algorithm Execution

We now give the pseudocode of EQP for each sensor node that is used for the execution of the distributed algorithm,and shows the operational process clearly.

Algorithm 2 EQP

```
Synchronization
While ($t \leqslant T$) do
  Duty cycling according to Algorithm 1
  Drop data according to formula (23)
  Calculate expected admission according to formula (25) and (26)
  Each sensor node sends $G_n(t)$ to the sink node,
  wait the sink node compares $G_n(t)$ and feedback $\xi_n(t)$
  If $\xi_n(t)$ == 1
    Send data
  Else
    Sleep
  End if
  Update $E_n^c(t)$, $Qvir_n^c(t)$, $Z_n^c(t)$
End while
```

VII. Simulation Results

First, simulation settings are provided in this section. Then, we vary our key parameters to study their influence on the algorithm in terms of *i) normalised throughput* (the number of successfully transmitted bits within the tolerable delay) and *ii) maximum delay time slot*. At last, two other algorithms are compared with EQP by using the following two metrics: *i) energy sustainability* and *ii) utility* which is presented in formula (1) .

A. Simulation Setup

We assume that the hoisting system lifts coal from underground to overground. The initial speed for the drum is 0, the acceleration is 0.7 m/s^2, and the maximum speed is 12 m/s. The monitoring network is designed as a one-hop star topology (Fig.2(a)) and consists of $n = 3$ sensor nodes and 1 sink node. The monitoring network operates on $l = 3$ channels and each channel s_n has three different states ”Good”, ”Medium” and ”Bad”, which represent 150, 50 and 30 kbps service rates respectively [24]. The channel states arise as i.i.d. vectors $(s_1(t), s_2(t), s_3(t))$ in every time slot. Based on [25], the probability of each vector is as follows, $Pr[(G, M, M)] = 1/3$, $Pr[(M, B, B)] = 1/9$, $Pr[(M, M, M)] = 1/9$, $Pr[(G, B, B)] = 2/9$, $Pr[(M, G, G)] = 1/9$.

We set the sampling frequency for the sensing task as 40kHz. The maximum duty cycle is 0.8 and the minimum duty cycle is 0.1. Therefore, the maximum number of arrival data (bits) for two classes is R^c_{max}=32k in one slot. We use CC2650 sensor tags as our wireless sensor nodes. The sensing power $p_{sen} = 0.001W$ and the transmission power $p_{max} = 0.01W$. The sleep power consumption is negligible compared to the sensing and transmissions powers and is ignored, i.e. p_{slp}=0W. The maximum tolerable delay for HP data τ^H_{max} and for LP data τ^L_{max} is set as 3 and 6 time slots respectively. One time slot is 1s. The initial battery level for all sensors $E_n(0) = 0$. The maximum number of bits that can be dropped d^c_{max} is 32k. The admission weight for HP data υ^H=2 and LP data υ^L=1. ϵ^c is set as R^c_{max}. We conduct numerical calculations of our algorithm using Matlab. We evaluate our algorithm for 10000 time slots.

B. Parameter Studies

1) Influence of Parameter w^H: We first set LP weight factor $w^L = 1$, and then change HP weight factor w^H to study the influence. Fig. 4 shows the relationship between w^H and average throughput for HP data and LP data. In the figure, the average throughput for HP data first increases significantly with w^H and then stabilises at a certain level after reaching an inflexion point. The trend for LP data is the exact opposite of the HP data. Average throughput for LP data first increases slightly with w^H before decreasing significantly. After that, average throughput remains stable at a certain level but larger than zero. Tab. II shows the 95% confidence interval of throughput corresponding to each w^H and it can be seen that the range of these values is relatively small.

Now, we will analyze the theoretical reason for the impact of w^H on throughput. Throughput is directly affected by the number of transmissions, which is primarily determined by the power allocation strategy.

We first analyse the direct reason. When the number of transmissions for c-class data increases, more c-class data can be serviced and fewer c-class data will be out of time. Therefore, the number of transmissions has a positive impact on throughput.

Next, we will study how w^H impacts the number of transmissions. Fig. 5 shows the relationship between w^H and the number of transmissions. The profiles of the number of transmissions (Fig. 5) have a similar trend of throughput in Fig. 4. The number of transmissions for HP data increases with w^H and then remains at a certain level. The number of transmissions for LP data has the opposite trend.

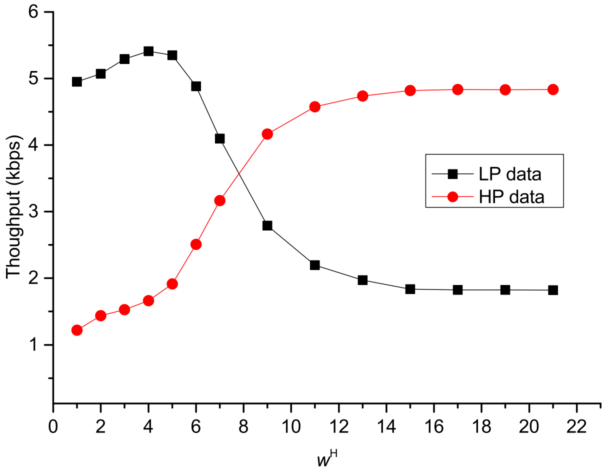


Fig. 4. Effect of w^H on throughput for HP and LP data .

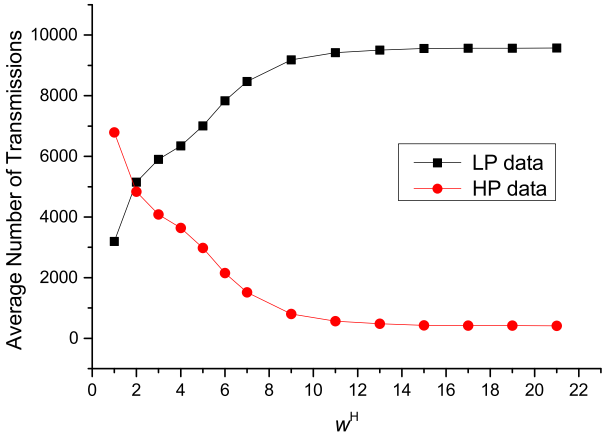


Fig. 5. Effect of w^H on number of transmissions for HP and LP data.

The power allocation strategy (29) is the main reason for the change in the number of transmissions. In the channel access competition, $G_n(t) = \sum_c w^{c2}\widetilde{\mu}^c_n(t)[Qvir^c_n(t) + Z^c_n(t)]$ must be chosen from nodes to achieve the maximum value. The HP weight factor w^H is designed to increase the importance of HP data in G_n, thereby increasing the significance of high-priority data. When both weight factors are equal, i.e. $w^H = 1$ and $w^L = 1$, there is a lesser chance for the HP data to be sent. This is because most of the HP data does not have enough time to increase its data queue backlogs required for transmission and so that the data will time out and be dropped. The reason for this is that the maximum delay time for HP data is smaller than that for LP data. Hence, the number of transmissions and throughput for the HP data is less than the LP data. When w^H increases, all the HP data have larger weighted queue backlogs than the LP queue backlogs and so the HP data gets more access to channel resources than LP data. The number of transmissions increases. In contrast, this

TABLE II
THE 95 % CONFIDENCE INTERVAL OF THROUGHPUT

w^H	Confidence interval of LP throughput	Confidence interval of HP throughput
1	[4.93,4.97]	[1.17,1.27]
2	[4.97,5.18]	[1.41,1.46]
3	[5.21,5.38]	[1.51,1.55]
4	[5.31,5.51]	[1.62,1.71]
5	[5.30,5.40]	[1.87,1.95]
6	[4.79,4.98]	[2.41,2.61]
7	[4.06,4.14]	[3.13,3.20]
9	[2.68,2.89]	[4.15,4.19]
11	[2.09,2.30]	[4.49,4.66]
13	[1.92,2.02]	[4.65,4.83]
15	[1.78,1.89]	[4.78,4.86]
17	[1.76,1.89]	[4.78,4.88]
19	[1.76,1.89]	[4.81,4.86]
21	[1.76,1.88]	[4.81,4.86]

index is lowering for LP data because they have lower channel access opportunities. If w^H keeps increasing and is larger than a certain point (inflexion point), the number of transmissions for HP data will be at the maximum value. In this case, once the expected HP data $\widetilde{\mu}_n^H$ is not zero in G_n, all HP data will be sent as soon as possible and the system reaches a saturation point. In contrast, the LP data still have a chance to send when $\widetilde{\mu}_n^H = 0$. Of course, since the channel quality varies, the throughput changes slightly even though the number of transmissions is the same.

We also calculate the value of the inflexion point. The lowest w^H should ensure that the HP data is sent even for the worst case. Assuming the power allocation evaluation factor $G_n(t)$ is calculated at each node, there are admitted LP data but no HP data at node 1, $\widetilde{\mu}_1^L > 0$, $\widetilde{\mu}_1^H = 0$. We also assume that node 1 has the largest LP queue backlogs, $Qvir_3^L(t) + Z_3^L(t) = 6\epsilon$. The relevant channel state is "Good", representing 150 kbps. The duty cycle has a maximum value of 0.8. In comparison, there are admitted HP data but no LP data in node 3's queues, $\widetilde{\mu}_3^H > 0$, $\widetilde{\mu}_3^L = 0$. Besides, the HP data at node 3 are new arrivals, and the sum of virtual queues is $Qvir_3^H(t) + Z_3^H(t) = \epsilon$. Assuming that the relevant channel state is "Bad" (representing 30 kbps), the duty cycle takes the minimum value of 0.1. Therefore, the critical point is $\sqrt{6\epsilon/\epsilon \times 150 \times 0.8/(30 \times 0.1)} = 15.49$. This is the theoretical explanation for the inflexion point around 15 and 16 in Fig. 5.

2) Influence of Parameter $Rvir_n^c$: In this section, we evaluate the delay of the proposed algorithm with $Rvir_n^c$ and without $Rvir_n^c$. In order to show the comparative results better, we set $w^H = 1$, $w^L = 1$, $\epsilon^H = 0.7R_{max}$, $\epsilon^L = 0.7R_{max}$ and repeat the simulation 5 times. Fig. 6 shows the average value of maximum delay and the 95% confidence interval of the maximum delay. Queue number pairs $\{1, 2\}$, $\{3, 4\}$, $\{5, 6\}$ are the data queues that belong to node 1, 2 and 3 respectively. Queue numbers 1, 3 and 5 are LP queues, and queue numbers 2, 4 and 6 are HP queues.

For the proposed algorithm with $Rvir_n^c$, the delay requirements are satisfied for different priority data. The average of the maximum delay for LP data is 5 time slots, and the requirement is also satisfied for HP data, not larger than 3 time slots. However, the delay requirements are not satisfied with the algorithm without $Rvir_n^c$. The average of the maximum delay in simulations is 6.8, 6.4 and 6.8 time slots for three nodes respectively, which are more than the requirement of 6 time slots. Additionally, the right endpoints of corresponding confidence intervals are also larger than 6. For HP data, the average of one of the maximum delay is 4 time slots, which is also larger than 3 time slots. Therefore, $Rvir_n^c$ is necessary to guarantee our delay requirements.

The theoretical reason for the difference in delay has been given when we set the value of $Rvir_n^c$. The worst-case delay is ensured by limiting the maximum sum of Q_{max}^c and Z_{max}^c with a continuous $R_n^c(t)$ [8]. This theory holds when the data priority remains unchanged. However, when the data priority changes, one of the priority queue has no arrival data anymore. This violates the prerequisite for the theory and the worst-case delay will not be bounded. Hence we use the concept of virtually arriving data $Rvir_n^c(t)$ to update the data queues to ensure that the theory works. The results also show the correctness and necessity.

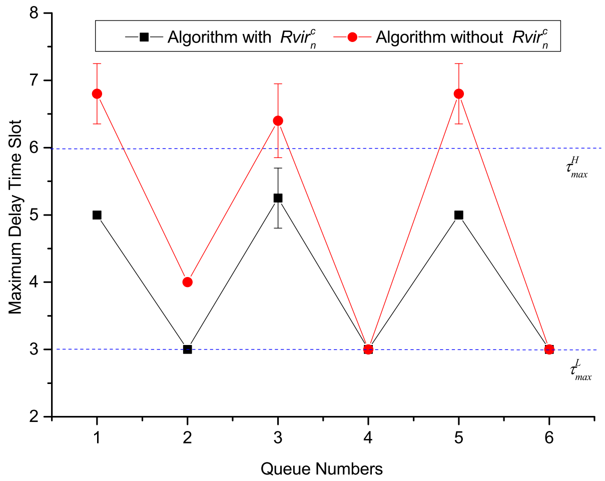
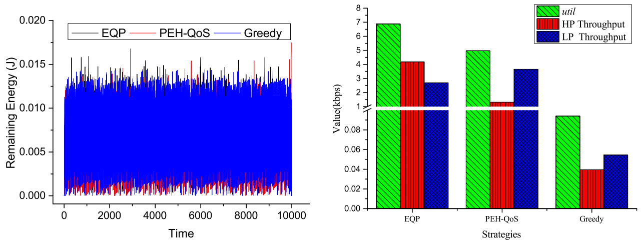


Fig. 6. Effect of $Rvir_n^c$ on the maximum delay.

C. Performance Comparisons

We compare our EQP algorithm with two different algorithms. The first one is a joint control scheme (PEH-QoS) [26] and the second one is a greedy strategy in which all harvested energy is used to transmit in every time slot [27]. We use the data queue aware control strategy from the PEH-QoS scheme. In the greedy strategy, all three sensors randomly have access to the channel to send their data. For the EQP algorithm, we set $w^H = 9$ and $w^L = 1$.

The results for the comparisons are shown in Fig. 7(a). The figure shows the energy dynamics and the throughput for the three strategies. As can be seen from Fig. 7(b), there is always some energy remaining in the batteries during a run. All three strategies ensure ENO and that sensor nodes work continuously. The main advantage of our algorithm can be seen in Fig. 7(b). This figure shows the system utility and the throughput of the priority data when ϱ^H=ϱ^L=1. Our proposed EQP strategy achieves at least 38% higher utility than the other two algorithms. The reason this happens is because EQP considers the channel status before making a decision and uses the channel with good conditions for as many transmissions as possible. High priority data also have higher throughput in EQP compared to other strategies. This happens because the priority-based admission control strategy is designed into EQP and therefore high priority data have a higher chance to be sent. In contrast, there is no priority control strategy in the other two algorithms. The greedy strategy achieves the worst performance of them all. This is because sensor nodes have no dropping strategy and so many expired messages are occupying the channel. This also reduces the chances for newer or higher-priority messages to be sent.



(a) (b)

Fig. 7. Simulation comparisons (a) Remaining battery energy dynamics (b) Utility and throughput in strategies.

VIII. Conclusions

In this paper, we study how to optimise piezoelectric harvesting WSNs to improve the health monitoring quality of hoisting systems. First, the health monitoring system is introduced and based on this, the optimisation problem is formulated. Then, the system models are designed. With the system model in mind, we design the EQP algorithm to solve the problem. In this algorithm, based on the energy model, a duty cycle management is proposed to ensure ENO of sensor nodes (so that they operate perpetually). Network scheduling is implemented via queue approaches and the Lyapunov optimisation technique ensures that the data is timely but not at the cost of energy neutrality. During this process, we develop a novel technique called HP weight factor w^H to differentiate and balance QoS service of different priority data depending on the priority or significance of the data (from different parts of the hoist). In addition, we also develop a novel scheme that utilises the notion of a virtual arrival $Rvir_n^c$ to ensure the liveliness of the data. Simulations show that EQP achieves significant utility improvements over state-of-art algorithms. Future work will examine how similar principles can be applied to other elements of condition monitoring of machinery.

References

[1] L. Giraud and B. Galy, “Fault tree analysis and risk mitigation strategies for mine hoists,” *Safety science*, vol. 110, pp. 222–234, 2018.

[2] G. Zhou, P. Wang, Z. Zhu, H. Wang, and W. Li, “Topology control strategy for movable sensor networks in ultradeep shafts,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 2251–2260, 2018.

[3] X. Zhang, Z. Song, J. Da, and J. Fu, “A novel acoustic filtering sensor for real-time tension monitoring of hoist wire ropes,” *Sensors*, vol. 18, no. 9, p. 2864, 2018.

[4] F. Jiang, Z. Zhu, W. Li, S. Xia, and G. Zhou, “Lifting load monitoring of mine hoist through vibration signal analysis with variational mode decomposition.” *Journal of Vibroengineering*, vol. 19, no. 8, 2017.

[5] S. Xue, J. Tan, and L. Shi, “Uhf wave propagation in mine shaft environment,” *Progress In Electromagnetics Research*, vol. 71, pp. 157–167, 2018.

[6] G. Zhou, H. Wang, Z. Zhu, L. Huang, and W. Li, “Performance analysis of wind-induced piezoelectric vibration bimorph cantilever for rotating machinery,” *Shock and Vibration*, vol. 2015, 2015.

[7] Z. Li, G. Zhou, Z. Zhu, and W. Li, “A study on the power generation capacity of piezoelectric energy harvesters with different fixation modes and adjustment methods,” *Energies*, vol. 9, no. 2, p. 98, 2016.

[8] M. J. Neely, “Opportunistic scheduling with worst case delay guarantees in single and multi-hop networks,” in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 1728–1736.

[9] Y. Deng, Z. Chen, D. Zhang, and M. Zhao, “Workload scheduling toward worst-case delay and optimal utility for single-hop fog-iot architecture,” *IET Communications*, vol. 12, no. 17, pp. 2164–2173, 2018.

[10] B. Liu, Z. Yan, and C. W. Chen, “Medium access control for wireless body area networks with qos provisioning and energy efficient design,” *IEEE transactions on mobile computing*, vol. 16, no. 2, pp. 422–434, 2016.

[11] M. B. Rasheed, N. Javaid, M. Imran, Z. A. Khan, U. Qasim, and A. Vasilakos, “Delay and energy consumption analysis of priority guaranteed mac protocol for wireless body area networks,” *Wireless networks*, vol. 23, no. 4, pp. 1249–1266, 2017.

[12] M. Hammoudeh and R. Newman, “Adaptive routing in wireless sensor networks: Qos optimisation for enhanced application performance,” *Information Fusion*, vol. 22, pp. 3–15, 2015.

[13] L. Cheng, J. Niu, J. Cao, S. K. Das, and Y. Gu, “Qos aware geographic opportunistic routing in wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1864–1875, 2014.

[14] F. Dobslaw, T. Zhang, and M. Gidlund, “Qos-aware cross-layer configuration for industrial wireless sensor networks,” *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1679–1691, 2016.

[15] M. Usman, N. Yang, M. A. Jan, X. He, M. Xu, and K.-M. Lam, “A joint framework for qos and qoe for video transmission over wireless multimedia sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 4, pp. 746–759, 2017.

[16] S. Yang, X. Yang, J. A. McCann, T. Zhang, G. Liu, and Z. Liu, “Distributed networking in autonomic solar powered wireless sensor networks,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 12, pp.

750–761, 2013.

[17] T. N. Le, A. Pegatoquet, O. Berder, and O. Sentieys, "Energy-efficient power manager and mac protocol for multi-hop wireless sensor networks powered by periodic energy harvesting sources," *IEEE Sensors Journal*, vol. 15, no. 12, pp. 7208–7220, 2015.

[18] S. Peng, T. Wang, and C. Low, "Energy neutral clustering for energy harvesting wireless sensors networks," *Ad Hoc Networks*, vol. 28, pp. 1–16, 2015.

[19] S. M. Bozorgi, A. S. Rostami, A. A. R. Hosseinabadi, and V. E. Balas, "A new clustering protocol for energy harvesting-wireless sensor networks," *Computers &amp; Electrical Engineering*, vol. 64, pp. 233–247, 2017.

[20] M. Shin and I. Joe, "Energy management algorithm for solar-powered energy harvesting wireless sensor node for internet of things," *IET Communications*, vol. 10, no. 12, pp. 1508–1521, 2016.

[21] G. A. Shah, V. C. Gungor, and O. B. Akan, "A cross-layer qos-aware communication framework in cognitive radio sensor networks for smart grid applications," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1477–1485, 2013.

[22] D. Zhang, Z. Chen, M. K. Awad, N. Zhang, H. Zhou, and X. S. Shen, "Utility-optimal resource management and allocation algorithm for energy harvesting cognitive radio sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3552–3565, 2016.

[23] Z. Liu, B. Liu, and C. W. Chen, "Joint power-rate-slot resource allocation in energy harvesting-powered wireless body area networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 12, pp. 12 152–12 164, 2018.

[24] M. O. Farooq and T. Kunz, "Contiki-based ieee 802.15. 4 channel capacity estimation and suitability of its csma-ca mac layer protocol for real-time multimedia applications," *Mobile Information Systems*, vol. 2015, 2015.

[25] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[26] E. Ibarra, A. Antonopoulos, E. Kartsakli, J. J. Rodrigues, and C. Verikoukis, "Qos-aware energy management in body sensor nodes powered by human energy harvesting," *IEEE Sensors Journal*, vol. 16, no. 2, pp. 542–549, 2015.

[27] C. Qiu, Y. Hu, Y. Chen, and B. Zeng, "Lyapunov optimization for energy harvesting wireless sensor communications," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1947–1956, 2018.

Houlian Wang received the Ph.D. degree in mechatronic engineering from China University of Mining and Technology, Xuzhou, China. He was also a visiting Ph.D. student with Imperial College London, UK. Now, he is working at School of Mechanical Engineering, Jiangsu University of Science and Technology.

His research interests include low-powered wireless sensor network and battery management.

Gongbo Zhou received the B.S. degree in computer science and technology from the Anhui University of Science and Technology, in 2005, and the D.E. degree from the School of Mechanical and Electrical Engineering, China University of Mining and Technology, Xuzhou, China, in 2010.

From 2009 to 2010, he was a visiting Ph.D. student with University of Wisconsin, Madison, WI, USA. He is currently a Professor with the School of Mechatronic Engineering, China University of Mining and Technology.

His current research interests include wireless sensor networks and and battery management.

Laksh Bhatia received his M.S. degree in Embedded systems from Technical University of Berlin and Aalto University in 2017. He is currently a Research Assistant and a PhD student in AESE labs at the Department of Computing in Imperial College London.

His research interests include Cyber-Physical Systems, Low Power Wide Area Networks and Wireless Sensor Networks.

Zhencai Zhu received Ph.D. degree in School of Mechanical and Engineering, China University of Mining and Technology, Xuzhou, China, in 2000. He is currently a professor the school of mechatronic engineering, China University of Mining and Technology, Xuzhou, China. He was also a visiting research scholar with University of Wuppertal, German, from 2010 to 2011. He finished the work as a postdoctor in Central South University, Changsha, China, in 2003. He was the dean of science and technology research institute of China University of Mining and Technology.

His current research interest is electromechanical system control and automation.

Wei Li received his PhD degree from University of Duisburg-Essen, Germany, in 2009. Now he is a professor of mechatronic engineering at School of Mechatronic Engineering, China University of Mining and Technology.

His current research interest is monitoring of mechatronic system.

Julie A McCann is a Full-Professor of Computing Systems at Imperial College London, Department of Computing. Her research centres on highly decentralized and self-organizing scalable algorithms for sensor-based computing systems, wireless networks and energy neutral computing.

She leads the Adaptive Embedded Systems Engineering Research Group and has worked with Intel, Cisco and NEC etc. on substantive smart city projects. She is an elected peer for the EPSRC and is a Fellow of the BCS.