Multiperiod Scheduling for Wireless Sensor Networks: A Distributed Consensus Approach

Jianping He, Lingjie Duan, *Member, IEEE*, Fen Hou, *Member, IEEE*, Peng Cheng, *Member, IEEE*, and Jiming Chen, *Senior Member, IEEE*

***Abstract*—In wireless sensor networks, many sensors face energy constraints and can switch among different work modes to save energy. How to properly schedule work modes is important for network utility maximization (NUM) in the long run. This paper proposes multiperiod scheduling to maximize total network utility by considering energy constraints and periodic sensing requirements. This NUM problem presents challenging mixed-integer programming, and it is difficult to solve by using a centralized approach under complete information. Thus, we first simplify the multiperiod problem to an equivalent single-period problem, and then further reduce it to a pure-integer programming problem, which can be solved easily in a centralized way. As for the cases without a centralized coordinator among all sensors, we propose an average consensus-based distributed algorithm (ACDA) to distributively schedule the work modes of all sensors using only local information. We prove that ACDA converges exponentially fast and reaches global optimum as long as the energy consumption of running the algorithm is ignorable. The proposed distributed solution is also robust against packet drop, node failures, and the changes of communication topology. Extensive simulation results have also shown the effectiveness of the proposed distributed algorithms.**

***Index Terms*—Consensus, distributed algorithms, multiperiod scheduling, network utility maximization, wireless sensor networks.**

I. Introduction

THE sensor nodes in wireless sensor networks (WSNs) usually have three work modes: sensing, communication and sleep, where sensing is to monitor ambient environment and sleep is to save energy [1]. storage capability, it is necessary to schedule sensors' work modes in order to achieve the optimal network performance or to prolong network lifetime [2]. Accordingly, many studies have looked into network utility (performance) maximization (NUM) by considering energy constraints at sensors [3]–[10].

Previous works on NUM have focused on sensors' communication to maximize data collection to the fusion center. However, the energy constraint has not been investigated intensively. Many sensors (e.g., MICA and camera sensors) actually consume far more energy in the sensing mode than in the communication/reporting mode and it is important to optimize all the modes in scheduling. For example, a MICA sensor consumes about 90 milliampere (mA) for sensing, 6 mA for sleep, and 12 mA for communication [11], which shows that sensing consumes the most energy and cannot be ignored in sensor scheduling. Even in the platforms established using low-power-listening techniques [12], sensing is still the work mode that consumes the most energy. In some other applications of sensors, network performance depends on sensing duration mostly rather than network throughput. For example, for event detection in WSNs, a longer sensing time provides a higher detection probability. Besides, many applications require the sensors to transmit their sensing data to their neighbors or the fusion center before a given deadline, and the sensors need to work periodically. To practically capture all these features or network performances, it is necessary to redefine the current network utility model as a function of sensing time. Based on the new network utility model, we can take three work modes (i.e., sensing, communication and sleep) into consideration to design multi-period scheduling for optimal network performance.

In this paper, we formulate a multi-period scheduling problem by considering energy constraints and periodic sensing requirements. Different from prior works that mainly focused on maximizing network throughput by flow control [3]–[9], we design a multi-period scheduling over three work modes to achieve NUM. We develop a novel consensus-based algorithm to solve the proposed problem in a completely distributed way. The consensus algorithms usually have a lower computation complexity (only to average each node's neighbors' states at each iteration) compared with traditional decomposition methods, e.g., Sub-gradient method [37], and have an exponential convergence speed. The key contributions of this work are summarized as follows:

- **Multi-period and multi-mode scheduling:** To the best of our knowledge, this is the first work that investigates multi-period scheduling for NUM by considering energy constraints and periodic sensing requirements. The formu-

Manuscript received April 12, 2014; revised September 04, 2014 and November 25, 2014; accepted January 05, 2015. Date of publication January 20, 2015; date of current version February 23, 2015. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Walaa Hamouda. This work was partially supported by NSFC U1405251 and 61473251, National Program for Special Support of Top Notch Young Professionals, CPSF under grant 2014M551739, and State Key Lab. of Synthetical Automation for Process Industries; the work by Lingjie Duan was supported by the SUTD-MIT International Design Centre (IDC) Grant (Project Number: IDSF1200106OH) and the SUTD-ZJU Joint Collaboration Grant (Project Number: SUTD-ZJU/RES/03/2014). The work of F. Hou was supported by the Research Grant MYRG2014–00140-FST from the University of Macau. (*Corresponding author: P. Cheng.*)

J. He, P. Cheng, and J. Chen are with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, 310027, China (e-mail: jphe@iipc.zju.edu.cn; pcheng@iipc.zju.edu.cn; jmchen@iipc.zju.edu.cn).

L. Duan is with the Engineering Systems and Design Pillar, Singapore University of Technology and Design, 138682, Singapore (e-mail: lingjie_duan@sutd.edu.sg).

F. Hou is with the Science and Technology, University of Macau, Taipa, Macau, China (e-mail: fenhou@umac.mo).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2015.2394507

lated scheduling problem considers three work modes and the dynamic changes of battery states.

- **Problem simplification for tractable analysis :** The multi-period problem is first simplified to an equivalent single-period problem. Then, we investigate a single-period problem with the maximum lifetime constraint, which helps simplify the initial mixed-integer programming to a solvable pure-integer programming problem.
- **Optimal design of distributed algorithms :** A minimum consensus-based distributed algorithm (MCDA) is first proposed to solve the single-period problem and to satisfy the maximum lifetime constraint. With the insights derived from the MCDA, we develop ACDA to optimally solve the original multi-period NUM problem. We prove that ACDA converges exponentially fast and is robust against various network uncertainties (e.g., dynamic network lifetime and the changes of communication topology).

The remainder of this paper is organized as follows. Section II presents the related work. We introduce the system model and set up a multi-period scheduling problem in Section III, and analyze the simplification of the problem in Section IV. Then, we propose a novel distributed algorithm to solve the multi-period scheduling problem in Section V. Finally, we conduct extensive simulations to evaluate the performance of the algorithms in Section VI.

II. Related Work

Many efforts have been devoted to investigating the NUM problem by considering the limited energy of sensors [3]–[10]. The studies have focused on the design, control and scheduling of various network parameters, e.g., flow control, power control, rate allocation and scheduling, etc., to obtain optimal or sub-optimal network utility. For example, the authors in [3]–[5] designed an optimal flow control to maximize total network utility with different constraints, e.g., lifetime constraint and link interference. Chiang in [9] presented a distributed power control algorithm to increase end-to-end throughput and energy efficiency of the network, which enhances network performance by balancing power control in the physical layer and congestion control in the transport layer. Palomar and Chiang in [10] provided a survey for the NUM problem, and proposed a general framework for solving NUM problem, including primal, dual, partial and hierarchical decompositions, to name a few. Recently, Shi and Zhang in [15], Shi and Xie in [16], and Wu *et al.* in [17] studied the sensor power scheduling problem for state estimation in a network control system and their goal was primarily to minimize the average error covariance of the estimation. However, for most of the above work, the network utility is a function of transmission data rate, which depends on the transmission power and the radio channel attenuation. Thus, these can be regarded as throughput optimization problems. That is, the corresponding scheduling problem is often communication-driven without considering the sensing and sleep modes.

Consensus algorithms, as important distributed computing methods, have gained intense attention recently [21]–[24], and have been widely used in various areas, e.g., time synchronization [25]–[27], distributed optimization [36], and environmental monitoring [39]. Especially, Nedic *et al.* in [36] combined a consensus algorithm with a distributed sub-gradient iteration algorithm to solve a constrained convex optimization problem. The work in this field provides us with pertinent perspective to utilize consensus concept to address a convex optimization problem in a distributed way.

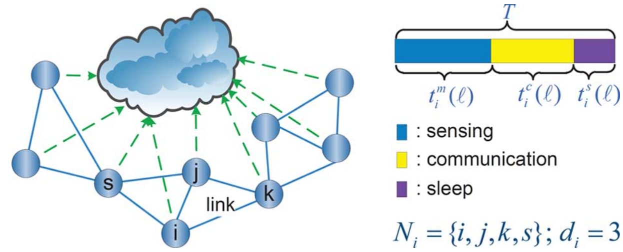


Fig. 1. Network model.

Compared with the existing studies, the main novelty of this paper includes: i) the formulation of practical multi-period scheduling problem for NUM, where three work modes are considered in each period of the scheduling; ii) network utility is redefined as a function of the sensing time, which thus characterizes the sensing-dependent network performance; and iii) we proposed distributed algorithms to directly solve the problems without using decomposition methods.

III. System Model and Problem Formulation

We use a undirected connected graph $\mathcal{G} = (V, \mathcal{E})$ to abstract the topology of sensors, where V denotes the set of vertices (sensors) and $\mathcal{E}$ is the set of communication links (do not consider the links between a cloud layer and sensors). Let $N_i = \{j | \{i, j\} \in \mathcal{E} \text{ and } i, j \in V\}$ be the neighbor set of node i, where node i is also considered as a neighbor of itself (i.e., $i \in N_i$ or $\{i, i\} \in \mathcal{E}$), and only neighboring sensors can communicate with each other. Let d_i be the degree of node i for $i \in V$, which tells the number of the nodes directly connecting to node i except the node i. Clearly, we have $d_i = |N_i| - 1$. Considering node i in Fig. 1 as an example, its neighbor set is $N_i = \{i, j, k, s\}$, and the degree is $d_i = 3$.

A. System Model

We consider a WSN consisting of N nodes (sensors), where each node monitors the environment and a targeted event in its surrounding area. We consider a time-slotted model with time interval T being the length of each slot/period. In each period, every node can have three work modes: sensing, communication and sleep. As shown in Fig. 1, let $t_i^m(\ell), t_i^c(\ell)$ and $t_i^s(\ell)$ represent the time interval that node $i \in V$ operates at the sensing mode, communication mode, and sleep mode, respectively, in time slot/period ℓ. The detailed elaboration is given as follows.

- *Sensing mode:* In time interval $t_i^m(\ell)$, node i monitors the environment. Its energy consumption in this time interval equals $t_i^m(\ell) \times p^m$, where p^m is per time energy consumption in sensing mode.
- *Communication mode:* In time interval $t_i^c(\ell)$, node i reports its monitored data to a cloud layer (e.g., data collection nodes) and broadcasts to its neighbors, for data fusion and network control decisions. Its energy consumption in this time interval equals to $t_i^c(\ell) \times p^c$, where p^c is per time energy consumption in communication mode.
- *Sleep mode:* In time interval $t_i^s(\ell)$, node i stops monitoring and communicating. Its energy consumption in this time

interval equals $t_i^s(\ell) \times p^s$, where p^s is per time energy consumption in sleep mode.

Clearly, we have $t_i^m(\ell) + t_i^s(\ell) + t_i^c(\ell) = T$ for $\ell \in \mathrm{N}^+$. As long as sensors have computing function to fuse sensing data locally (e.g., in cognitive radios [38] and distributed localization application [40]), they do not need to transmit a large amount of data but only the fusion results. Thus, we assume the communication time in each period is fixed for each node i, i.e., $t_i^c(\ell) = t_i^c, \forall i \in V$ for each period ℓ. Let $t_{th}^m (t_{th}^m > 0)$ be the minimum required sensing time in each period, i.e., $t_i^m(\ell) \geq t_{th}^m, \forall \ell \in \mathrm{N}^+$, which guarantees each node i can obtain enough monitor data during each sensing period. Clearly, we have

$$t_{th}^m \leq t_i^m(\ell) \leq T - t_i^c, \ell \in \mathrm{N}^+, \forall i \in V. \quad (1)$$

Suppose that each node i has an initial energy $e_i(0)$ for itself to perform a task over time[1]. At time t, the energy of node i, say $e_i(t)$, satisfies

$$e_i(t) = e_i(0) - \int_0^t h_i(\tau) d\tau, \quad (2)$$

where $h_i(\tau) \geq 0$ is the energy consumption rate of node i at time τ [28]. The value of $h_i(\tau)$ depends on the work mode of node i. Specifically, when node i is sensing, sleep and communication at time τ, we respectively have $h_i(\tau) = p^m, h_i(\tau) = p^s$, and $h_i(\tau) = p^c$.

Given scheduling $(t_i^m(\ell), t_i^s(\ell), t_i^c(\ell)), \forall i \in V$ for each node i in one period ℓ, the energy consumption, defined as $g_i(t_i^m(\ell), t_i^s(\ell), t_i^c(\ell))$, can be calculated by

$$g_i(t_i^m(\ell), t_i^s(\ell), t_i^c(\ell)) = p^m t_i^m(\ell) + p^s t_i^s(\ell) + p^c t_i^c(\ell). \quad (3)$$

Since $t_i^m(\ell) + t_i^s(\ell) + t_i^c(\ell) = T$ and $t_i^c(\ell) = t_i^c$, $g_i(t_i^m(\ell), t_i^s(\ell), t_i^c(\ell))$ can be rewritten as a linear function of $t_i^m(l)$ as follows.

$$g_i(t_i^m(\ell)) = \alpha t_i^m(\ell) + \beta_i, i \in V, \quad (4)$$

where $\alpha = p^m - p^s$ and $\beta_i = p^c t_i^c + p^s(T - t_i^c)$. It is reasonable to assume that $p^m > p^s$ as the sensing energy consumption is usually larger than that sleep energy consumption [4], [28]. Thus, we have $\alpha = p^m - p^s > 0$. Clearly, each $g_i(t_i^m(\ell)) > 0$ holds for $t_i^m(\ell) \in [t_{th}^m, T - t_i^c]$, and $g_i(t_i^m(\ell))$ is a linear increasing function of $t_i^m(\ell)$.

Remark 3.1: We can also extend our results to the scenario where the communication time is a constant plus an additional time proportional to the sensing time $t_i^m(\ell)$ for node i in each period ℓ, i.e., $t_i^c(\ell) = t_i^c + c t_i^m(\ell)$, where c is a coefficient to transmit sensing data. In this case, the energy cost function (4) in each period is still a linear function, with $\alpha = p^m + cp^c - p^s - cp^s$ and $\beta_i = p^c t_i^c + p^s(T - t_i^c)$. And the upper bound $T - t_i^c$ of node i's sensing time becomes $\frac{T - t_i^c}{1+c}$. Based on a linear energy cost function (4), the analysis of optimal solutions and the design of the algorithms only depend on the parameters α and β_i and the upper bound of node sensing time. Therefore, the main results of the remaining parts of this paper will not be affected in this scenario.

We summarize all key notations in Table I.

[1]One can view $e_i(0)$ as the battery capacity of node i.

TABLE I
NOTATION DEFINITIONS

Symbols	Definitions
T	the common time interval of all nodes for each period;
t_i^m	the sensing time of node i in each period;
t_i^s	the sleep time of node i in each period;
t_i^c	the communication time of node i in each period;
t_{th}^m	the minimum required sensing time in each period;
p^m	the per time energy consumption at sensing stage;
p^s	the per time energy consumption at sleep stage;
p^c	the per time energy consumption at communication stage;
e_i	the energy of node i;
g_i	the energy consumption of node i in each period;
L_i	the lifetime of node i;
U_i	the utility function at node i;
k^-	the time just before updating at iteration k;
Δt	the iteration time step;
$\mathbf{N}^+$	the set of positive integer.

B. Performance Metrics

At time t, the residual network lifetime $L(t)$ is defined as the number of periods that all sensors can work under constraint (1), i.e.,

$$L(t) = \min_{i \in V} \max_{K_i} \{K_i | e_i(t) - \sum_{\ell=1}^{K_i} g(t_i^m(\ell)) \geq 0\}, \quad (5)$$

which depends on the scheduling of sensing time $t_i^m(\ell)$ in each period ℓ and the setting of lowest sensing time t_{th}^m. This network lifetime definition is practical for some applications [4], [14], e.g., in applications for forest fire detection and military surveillance. When we set the same sensing time duration for all nodes in each period (i.e., $t_i^m(\ell) = t_i^m$ for $\ell \in \mathrm{N}^+$), we have

$$L(t) = \min_{i \in V} L_i(e_i(t), t_i^m) = \min_{i \in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor. \quad (6)$$

Actually, when there exists one node i with $e_i(t) < g_i(t_i^m)$ at time t, the energy $e_i(t)$ is no longer sufficient for it to finish its one period work (including sensing and communication), and then the network is deemed dead.

Note that the node with a longer sensing time may obtain a better accuracy estimation or a higher event detect probability, which benefits the control decision. Hence, a utility function, $U_i(t_i^m(\ell))$, is defined to describe the utility of each node i with the sensing time $t_i^m(\ell)$ in each period ℓ. We assume $U_i(t_i^m(\ell))$ for $\forall i \in V$ is twice differentiable on its variable $t_i^m(\ell)$ and satisfies the following two conditions:

1) $U_i(t_i^m(\ell)) \geq 0$ and $U_i(t_i^m(\ell)) = 0$ if and only if $t_i^m(\ell) \leq t_{th}^m$ for $i \in V$. This utility depends on the value of the sensing time and has a positive value only when the sensing time is larger than t_{th}^m;
2) $U_i'(t_i^m(\ell)) > 0$ and $U_i''(t_i^m(\ell)) < 0$ for $i \in V$, i.e., each utility function is monotonically increasing during the sensing time $t_i^m(\ell)$ and is a strictly concave function.

There are various applications, where a longer sensing time will obtain more utility (e.g., detection accuracy) in WSNs, similar to our focus in this paper. For example, in a cognitive radio network, sensors in the secondary system try to opportunistically sense and access unused channels of the primary system in order to improve the efficiency of spectrum usage. As pointed out by [38], a longer sensing time leads to a more accurate channel estimation (with a smaller false alarm probability) of spectrum occupancy, where each sensor only makes a binary decision about primary signal presence or absence after sensing. As the sensing time becomes longer, the utility in terms of sensing accuracy does not increase linearly and usually presents a diminishing return from sensing. In this paper, we adopt a widely used logarithmic function that satisfies the above assumptions, consisting with [3], [4], [9], to characterize the network utility.

C. Scheduling Problem Formulation

Note that the utility $U_i(t_i^m(\ell))$ in time period ℓ is an increasing function of $t_i^m(\ell)$, while network lifetime is a decreasing function of sensors' sensing time. The goal of this paper is to design the sensing time $t_i^m(\ell)$ for each node i in each period ℓ such that the total network utility throughout the network lifetime is maximized. We formulate it as a multi-period scheduling optimization problem as follows.

$$P_1: \max_{t_i^m(\ell)} \sum_{\ell=1}^{L(t)} \sum_{i \in V} U_i(t_i^m(\ell)) \quad s.t. \quad (1) \text{ and } (5) \text{ hold.} \tag{7}$$

It follows from (5) that $e_i(t) - \sum_{\ell=1}^{L(t)} g(t_i^m(\ell)) \geq 0$ holds for $i \in V$. Hence, condition (5) implies the total energy consumption of each node i under the setting of $t_i^m(\ell), \ell = 1, 2, \ldots, L(t)$, should be less than its current energy $e_i(t)$. Note that $L(t)$ should be an integer, which means P_1 is a mixed integer programming problem. Therefore, it is inherently difficult to find the optimal solution for problem P_1. The major challenges include:

1) Classical approaches, e.g., Lagrange multiply-based approach, are ineffective, as they require the objective function to be continuously differentiable which is not satisfied in the objective function in (7).
2) There are infinite values of $t_i^m(\ell), \ell = 1, 2, \ldots, L(t)$, available for each node i, thus it is difficult to solve this problem directly by using a general centralized algorithm, e.g., enumeration algorithm.
3) From (5), the variable $L(t)$ is a function of scheduling from all nodes without a closed-form expression, thus it is difficult to determine an optimal solution.
4) Due to the nature of mixed-integer programming problem, the optimal solution of P_1 may not be unique.

As shown in Fig. 2, we first introduce optimization problem P_2 as equivalent to initial problem P_1 with the same solution, which successfully simplifies problem P_1 to a single-period scheduling problem. Then we consider one special interesting optimization problem P_3, a sub-problem of P_2, and design MCDA to solve problem P_3. Utilizing the solution from problem P_3, we further simplify problem P_2 to a pure-integer programming problem, and thus this can be solved by a classical integer programming approach in a centralized way. Lastly, we design ACDA to solve the pure-integer programming problem. Hence, the initial problem P_1 can be solved in a completely distributed way. Especially, when the energy consumption in each iteration is ignorable, we prove that ACDA can obtain an optimal solution for the initial problem P_1. When the energy consumption in each iteration is taken into consideration, we

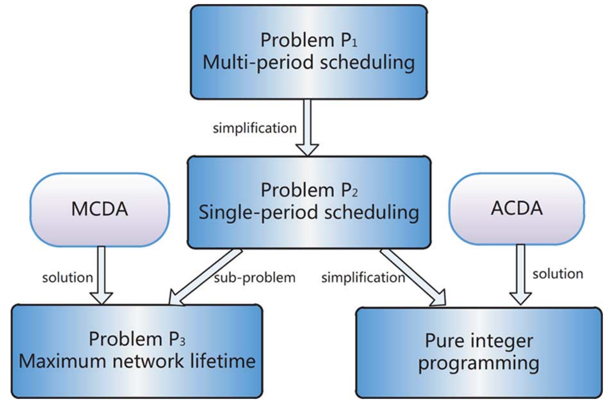


Fig. 2. Solution logic of different problems.

give the upper bound of the error between the optimal solution and the solution obtained from ACDA.

IV. Problem Simplification

In this section, we first transfer the multi-period problem to an equivalent single-period problem, which largely decreases the number of variables in the objective function. To further simplify the problem such that it is solvable, we study the distributed single-period scheduling problem with maximum lifetime. Then, based on observed insights, we transfer the problem to a pure-integer programming problem, which can be easily solved in a centralized way.

A. Equivalent Single-period Scheduling Problem

In this subsection, we introduce and prove an equivalent problem for problem P_1, which can simplify problem P_1 to a single-period scheduling optimization problem.

Consider an optimization problem as follows

$$P_2: \quad \max_{t_i^m} \sum_{i \in V} U_i(t_i^m) L(t) \quad s.t. \quad (1) \text{ and } (6) \text{ hold,} \tag{8}$$

where $L(t)$ denotes residual network lifetime at the time period t. Note that the number of the variables in single-period optimization problem P_2 is largely reduced comparing to the multi-period optimization problem P_1.

By referring to Jensen's inequality, it is obvious that for any concave function $u(x)$ satisfying $u'(x) > 0$ and $u''(x) < 0$, we have $u(x_0) \geq \frac{u(x_1)+u(x_2)+\cdots+u(x_m)}{m}$, where $x_0 = \frac{x_1+x_2+\cdots+x_m}{m}$, and the equality holds if and only if (iff) $x_\ell = x_0$ for $\ell = 1, 2, \ldots, m$. Hence, in (8), by setting the same scheduling for every period, we can obtain the maximum network utility when the total sensing time is fixed. Meanwhile, note that each $g_i(t_i^m(\ell))$ is a linear function of $t_i^m(\ell)$, which means that $g_i(t_i^m(0)) = \frac{\sum_{\ell=1}^m g_i(t_i^m(\ell))}{m}$ for $t_i^m(0) = \frac{t_i^m(1)+t_i^m(2)+\cdots+t_i^m(m)}{m}$ and $m \in \mathrm{N}^+$. This implies that the total energy consumption under different multi-period schedules is the same when these multi-period schedules have the same total sensing time. Therefore, we can obtain a conclusion that NUM can be achieved by equally allocating the available sensing time to each period. Thus, we can prove that P_2 is equivalent to P_1 in the following theorem.

Theorem 4.1: Problem P_1 is equivalent to problem P_2, i.e., these two problems have the same optimal solutions.

Proof: The proof is provided in the Appendix A. ■

With the help of this theorem, we only need to focus on P_2: a single-period scheduling problem.

Remark 4.2: We now quantitatively can compare problems P_1 and P_2 in terms of the computation complexity. Note that there are $N \times (L(t)+1)$ variables, i.e., $t_i^m(\ell), \ell = 1, 2, \ldots, L(t)$, and $e_i(t)$ for $i \in V$, in problem P_1. Only $2N$ variables, i.e., t_i^m and $e_i(t)$ for $i \in V$, in problem P_2. However, as each variable t_i^m can still be any value in $[t_{th}^m, T - t_i^c]$ and $L(t) = \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor$ is an integer and not differentiable, problem P_2 is also a mixed integer programming problem. Hence, it is still difficult to solve this problem directly.

To further simplify the problem such that it is solvable, we consider a special interesting optimization problem P_3 in the following subsection. It helps to convert problem P_2 to a solvable pure-integer programming problem. The corresponding results will be highly useful for us to find the optimal solution of problem P_2.

B. Reduced Pure-integer Programming Problem

Consider the following optimization problem,

$$P_3: \quad \max_{t_i^m} \quad \sum_{i\in V} U_i(t_i^m) L_{\max}(t) \tag{9}$$

$$s.t. \quad L_{\max}(t) = \max_{t_i^m \in [t_{th}^m, T - t_i^c]} \min_{i \in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor \tag{10}$$

where $L_{\max}(t)$ is the maximum residual network lifetime at the time t under constraint (1). It should be pointed out that network lifetime maximization is also crucial in a sensor network [4], [5], [34], [35], thus problem P_3 is also an important problem in WSNs.

Note that every $g_i(t_i^m)$ is an increasing function, it implies

$$\begin{aligned} L_{\max}(t) &= \min_{i\in V} \max_{t_i^m \in [t_{th}^m, T-t_i^c]} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor \\ &= \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_{th}^m)} \right\rfloor. \end{aligned} \tag{11}$$

Clearly, when the sensing time of each node equals t_{th}^m, the network lifetime is $L_{\max}(t)$. However, if every node i sets $t_i^m = t_{th}^m$, the objective (9) may not be achieved. Hence, determining an optimal solution and designing a distributed algorithm to obtain the optimal solution are the main challenges of problem P_3.

Since each $U_i(t_i^m)$ is a monotonically increasing function of t_i^m as $U_i'(t_i^m) > 0, \forall i \in V$, each node sets a maximum t_i^m in its feasible values to obtain the maximum network utility. Hence, under a given network lifetime $L(t) = L_{\max}(t)$, we have the following theorem, which guarantees there is a unique optimal solution for problem P_3 and also provides a closed-form optimal solution.

Theorem 4.3: The optimal solution for each node i of problem P_3 is unique and satisfies

$$t_i^m = \min\{T - t_i^c, \frac{\frac{e_i(t)}{L_{\max}(t)} - \beta_i}{\alpha}\}, i \in V. \tag{12}$$

With the guarantee of Theorem 4.3, in order to find the optimal solution t_i^m, each node can directly solve (12) in a distributed way once all nodes obtain the maximum network lifetime $L_{\max}(t)$. What's more, the value of maximum network lifetime can also be obtained through distributed information exchange. Thus, problem P_3 can be solved in a fully distributed manner. Hereafter, we will propose a minimum consensus based algorithm to solve problem P_3.

The key idea of minimum consensus is that neighboring nodes exchange their information iteratively, and at each iteration each node uses the minimum state among all its neighbor sensors' as its update. Then all sensors' states will converge to the smallest state among all sensors. Note from (11) that $L_{\max}(t) = \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_{th}^m)} \right\rfloor$, which is the smallest lifetime among all sensors' under the setting of $t_i^m = t_{th}^m, i \in V$. Intuitively, using minimum consensus can help every node to obtain $L_{\max}(t)$. The details are shown as follows.

Algorithm 1 : *Minimum Consensus-based Distributed Algorithm (MCDA)*

1:Given the initial conditions $e_i(0)$, t_i^c, T, t_{th}^m, α and β_i for $i \in V$. Let $t_k = k\Delta t$ for $k \in \mathrm{N}^+$ and let $l_i(0) = \left\lfloor \frac{e_i(0)}{g_i(t_{th}^m)} \right\rfloor$ be the initial maximum network lifetime estimation for $i \in V$.

2:At each iteration $k, k \in \mathrm{N}^+$, each node i reads its current energy $e_i(t_k)$ and calculates $\left\lfloor \frac{e_i(t_k)}{g_i(t_{th}^m)} \right\rfloor$. Then, let

$$l_i(k^-) = \min\{l_i(k-1), \left\lfloor \frac{e_i(t_k)}{g_i(t_{th}^m)} \right\rfloor\}, \tag{13}$$

and then broadcast $l_i(k^-)$ to its neighbor sensors.

3:If node $i, i \in \mathcal{V}$, receives a packet from its neighbor sensors, $j \in N_i$, then it updates its $l_i(k)$ according to

$$l_i(k) = \min_{j\in N_i} l_j(k^-). \tag{14}$$

4:After updating, set

$$t_i^m(k) = \min\{T - t_i^c, \frac{\frac{e_i(t_k)}{l_i(k)} - \beta_i}{\alpha}\}, \forall i \in V. \tag{15}$$

Since for minimum consensus, after k iterations, the state of the k-hop neighbors of i_m (say i_m who has minimum state) has been updated as the minimum state. That is, the convergence time of a minimum consensus for a connected network is not more than $N-1$. Thus, when the energy consumption during each iteration time interval $[t, t+\tau]$ is ignorable, i.e., $\int_t^{t+\Delta t} |h_i(\tau)| d\tau = 0, \forall i \in V$, the convergence time of MCDA is $N-1$. Considering the energy consumption of the algorithm, we give a theorem as follows.

Theorem 4.4: Suppose that the sensor network $G(V, \mathcal{E})$ is connected and $h_i(t) \le \frac{g_i(t_{th}^m)}{2N\Delta t}, \forall i \in V$, by using MCDA, the maximum network lifetime obtained by nodes satisfies

$$|l_i(k) - l_j(k)| \le 1, i, j \in V, \forall k \ge N-1. \tag{16}$$

There exists a $\hat{k} \in [N-1, 2N]$, such that

$$|l_i(\hat{k}) - l_j(\hat{k})| = 0, i, j \in V. \tag{17}$$

Proof: The proof is provided in the Appendix B. ■

In the above theorem, Δt is the step size of MCDA, and a smaller Δt decreases the value of $\int_t^{t+\Delta t}|h_i(\tau)|d\tau$. Thus, we can set a small Δt for two continuous iterations, such that the condition $h_i(t) \le \frac{g_i(t_{th}^m)}{2N\Delta t}$ is satisfied. For MCDA, each node can stop communicating when it finds that all its neighbors have the same states, and $h_i(t) = 0$ after the communication stops. Therefore, we infer from Theorem 4.4 that the convergence time of MCDA is the minimum $\hat{k}$ defined in the above theorem. Meanwhile, note that the condition $h_i(t) \le \frac{g_i(t_{th}^m)}{2N\Delta t}, \forall i \in V$ takes account of a node's energy variation, it follows from Theorem 4.4 that MCDA can be robust against the dynamic change of node energy.

From the above obtained results, we observe two insights. First, when the network lifetime is fixed, i.e., $L(t) = L$, in problem P_2, there is a unique optimal solution satisfying

$$t_i^m = \min\{T - t_i^c, \frac{\frac{e_i(t)}{L} - \beta_i}{\alpha}\}, \forall i \in V \tag{18}$$

i.e., all nodes will evenly allocate their energy to each period for sensing. Second, the maximum or minimum network lifetime under (1) can be obtained in a distributed way from minimum and maximum consensus-based algorithms.

Based on these two insights, problem P_2 can be transferred to a pure-integer programming problem. Specifically, it follows from the definition of the network lifetime that $L(t)$ should satisfy the following inequality,

$$L_{\min}(t) \le L(t) \le L_{\max}(t), \tag{19}$$

where $L_{\min}(t) = \min_{i\in V}\left\lfloor \frac{e_i(t)}{g_i(T-t_i^c)} \right\rfloor$ is the minimum network lifetime under (1). Since $L(t)$ should be an integer in interval $[L_{\min}(t), L_{\max}(t)]$, there are a finite number of feasible values for $L(t)$. Meanwhile, for each given $L(t)$, there is a unique optimal t_i^m for each node i. Hence, problem P_2 is a pure-integer programming problem, and can be solved by classical integer programming [29], which is a centralized approach. Specifically, we can calculate the maximum utility under optimal scheduling, $\sum_{i\in V} U_i(t_i^m)L$, by setting $t_i^m = \min\{T - t_i^c, \frac{\frac{e_i(t)}{L} - \beta_i}{\alpha}\}$ for any given $L(t) = L$. Note that there are a finite number of feasible values for $L(t)$, we can obtain an optimal $L^*(t)$ by comparing the values of $\sum_{i\in V} U_i(t_i^m)L(t)$ for different $L(t) \in [L_{\min}(t), L_{\max}(t)]$, so that $\sum_{i\in V} U_i(t_i^m)L(t)$ is the maximum. Thus, the optimal solutions of problem P_2 are

$$t_i^m = \min\{T - t_i^c, \frac{\frac{e_i(t)}{L^*(t)} - \beta_i}{\alpha}\}, i \in V. \tag{20}$$

Then, in order to solve problem P_2, it is equivalent to find the optimal network lifetime L^* such that the network utility is the maximum, which will be discussed in the next section.

V. Optimal Solution and Distributed Algorithm

As proved in Theorem 4.1, problem P_1 is equivalent to problem P_2, thus we can focus on how to find the optimal solution for problem P_2 directly. Although problem P_2 is simplified as a pure-integer programming problem, it is still a challenging problem as the global information is not available for the sensors. This section first analyzes the closed-form expression of the optimal solution. Then, based on this analysis, we propose ACDA to solve problem P_2.

A. Optimal Solution Analysis

Based on the observed insights discussed in the previous section, we know that the optimal solution of problem P_2 under the optimal network lifetime, denoted by $L^*(t)$, satisfies (20). Let $\tilde{L}(t) = \max_{i\in V}\left\lfloor \frac{e_i(t)}{g_i(T-t_i^c)} \right\rfloor$ be the lifetime of the node which will survive the longest when the sensing time is the maximum (i.e., $t_i^m = T - t_i^c, \forall i \in V$). Divide the feasible interval of the network lifetime $L(t)$ into two sets, $[L_{\min}(t), \tilde{L}(t))$ and $[\tilde{L}(t), L_{\max}(t)]$. When $L^*(t) \in [\tilde{L}(t), L_{\max}(t)]$, for $\forall i \in V$, one has

$$\begin{aligned} \frac{\frac{e_i(t)}{L^*(t)} - \beta_i}{\alpha} \le \frac{\frac{e_i(t)}{\tilde{L}(t)} - \beta_i}{\alpha} &\le \frac{\frac{e_i(t)}{\left\lfloor \frac{e_i(t)}{g_i(T-t_i^c)} \right\rfloor} - \beta_i}{\alpha} \\ &\le \frac{g_i(T - t_i^c) - \beta_i}{\alpha} \le T - t_i^c, \end{aligned} \tag{21}$$

which means that $\min\{T - t_i^c, \frac{\frac{e_i(t)}{L^*(t)} - \beta_i}{\alpha}\} = \frac{\frac{e_i(t)}{L^*(t)} - \beta_i}{\alpha}$, i.e., the optimal solution is

$$t_i^m = \frac{\frac{e_i(t)}{L^*(t)} - \beta_i}{\alpha}, \forall i \in V. \tag{22}$$

Hence, when $L^*(t) \in [\tilde{L}(t), L_{\max}(t)]$, each sensor will allocate its energy $e_i(t)$ evenly for each period to achieve the maximum network utility. In this case, the variable t_i^m in problem P_2 can be expressed as a function of network lifetime, which has the form as the right side of (22). Then, by substituting t_i^m into the function problem P_2, the objective function becomes an univariate function, where the unique variable is the network lifetime. Removing the integer constraint on network lifetime, problem P_2 can be transferred to a univariate optimization problem P_4, which is given by

$$\begin{aligned} P_4\colon \max_x \quad & \sum_{i\in V} U_i(f_i(x,t))x \\ s.t. \quad & f_i(x,t) = \frac{\frac{e_i(t)}{x} - \beta_i}{\alpha}, \ i \in V; \\ & \tilde{L}(t) \le x \le L_{\max}(t), \end{aligned} \tag{23}$$

where the continuous variable x is the network lifetime. For this problem, we have a theorem as follows.

Theorem 5.1: Given a time t, problem P_4 is a concave problem and has a unique optimal solution.

Proof: The proof is provided in the Appendix C. ■

Hence, the optimal solution of problem P_4 can be easily obtained by solving equation $\frac{\partial \sum_{i\in V} U_i(f_i(x,t))x}{\partial x} = 0$. The $L^*(t)$ of problem P_2 and its optimal solution can be found after obtaining the optimal solution of problem P_4.

Theorem 5.2: Suppose that x_t^* is the optimal solution of problem P_4 for a given time t. Then,

1) If $x_t^* \in [\tilde{L}(t), L_{\max}(t)]$, we have

$$L^*(t) = \arg \max_{x=\lfloor x_t^*\rfloor, \lceil x_t^*\rceil} \sum_{i\in V} U_i(f_i(x,t))x, \tag{24}$$

and the optimal solution of problem P_2 satisfies (22);

2) If $x_t^* > L_{\max}(t)$, $L^*(t) = L_{\max}(t)$ and the optimal solution of problem P_2 satisfies (22);

3) Otherwise, the maximization of problem P_2 is achieved when $L(t) \in [L_{\min}(t), \tilde{L}(t)]$.

Proof: The proof is provided in the Appendix D. ■

Remark 5.3: It is observed that $L^*(t) \in [\tilde{L}(t), L_{\max}(t)]$ iff $x_t^* \geq \tilde{L}(t)$ and $L^*(t) \in [L_{\min}(t), \tilde{L}(t)]$ iff $x_t^* < \tilde{L}(t)$, i.e., the value of x_t^* determines the interval that the optimal $L^*(t)$ belongs to. Specifically, if $x_t^* \geq \tilde{L}(t)$, $L^*(t) \in [\tilde{L}(t), L_{\max}(t)]$ and it can be easily obtained by comparing $\sum_{i\in V} U_i(f_i(x,t))x$ for $x = \lfloor x_t^* \rfloor$ and $\lceil x_t^* \rceil$. However, if $x_t^* < \tilde{L}(t)$, $L^*(t) \in [L_{\min}(t), \tilde{L}(t)]$, it can be obtained by comparing the maximum network utilities under all different lifetime values in $[L_{\min}(t), \tilde{L}(t)]$.

B. *Average Consensus-based Distributed Algorithm (ACDA)*

A completely distributed algorithm is naturally preferred for WSNs, as it has stronger scalability and robustness, e.g., robust against packet drop and dynamic network, than a centralized algorithm. In this section, we propose a fully distributed algorithm, ACDA, to help each node obtain the optimal scheduling.

From [21], [22], it follows that after given initial state for each node, an average consensus algorithm can help nodes with only local information to achieve average consensus (average of all nodes initial states). Moreover, the algorithm under a connected network has an exponential convergence speed. Herein, we propose ACDA to solve problem P_2 in a distributed way. To save the communication time, we consider the following two cases for the optimal solution:

1) **Medium lifetime regime**: When $L^*(t) \geq \tilde{L}(t)$, the optimal solution of problem P_2 can be obtained from solving problem P_4. Specifically, let $U_i(f_i(x,t))x$ be the initial state of node i. An average consensus algorithm can help each node to obtain $\frac{\sum_{i\in V} U_i(f_i(x,t))x}{N}$ when the algorithm converges. Then, each node can get x_t^* by solving $\frac{\partial \frac{\sum_{i\in V} U_i(f_i(x,t))x}{N}}{\partial x} = 0$ and obtain the optimal solution based on Theorem 5.2.

2) **Short lifetime regime**: When $L^*(t) < \tilde{L}(t)$, the optimal solution of problem P_2 is obtained by comparing the maximum network utility under different lifetime settings in $[L_{\min}(t), \tilde{L}(t)]$. Specifically, let m be the number of integers in $[L_{\min}(t), \tilde{L}(t)]$ and let $\mathrm{U}_i(t) = [U_i^1(t), U_i^2(t), \ldots, U_i^m(t)]^T$, where $U_i^\ell(t) = U_i(t_i^m(\ell))(L_{\min}(t) + \ell - 1)$ and $t_i^m(\ell) = \min\{T - t_i^c, \frac{\frac{e_i(t)}{L_{\min}(t)+\ell-1} - \beta_i}{\alpha}\}, \ell = 1, 2, .., m$. Set $\mathrm{U}_i(t)$ as the initial state of node i. Similarly, an average algorithm can help each node to obtain $\frac{\sum_{i\in V} \mathrm{U}_i(t)}{N}$. The maximum element of vector $\frac{\sum_{i\in V} \mathrm{U}_i(t)}{N}$, say k-th element, is the average of the maximum network utility. Then, $L^*(t) = L_{\min}(t) + k - 1$ and the corresponding optimal solution is $t_i^m(\ell) = \min\{T - t_i^c, \frac{\frac{e_i(t)}{L_{\min}(t)+\ell-1} - \beta_i}{\alpha}\}, i \in V$.

Firstly, consider ACDA algorithm for the medium lifetime regime case. Let $l_i^{\max}(0) = \left\lfloor \frac{e_i(0)}{g_i(t_{th}^m)} \right\rfloor$ and $l_i^{\min}(0) = \left\lfloor \frac{e_i(0)}{g_i(T-t_i^c)} \right\rfloor$ be the initial estimated maximum and minimum network lifetime of node i, respectively; Let $\tilde{l}_i(0) = \lfloor \frac{e_i(0)}{g_i(T-t_i^c)} \rfloor$ be the initial estimated maximum network lifetime under the scheduling of $t_i^m = T - t_i^c$; Let $U_i^a(x,0) = U_i(f(x,0))x$ be the node i's initial estimation of the average optimal network utility.

Algorithm 2 : *Average Consensus-based Distributed Algorithm (ACDA)*

1:At each iteration $k, k \in \mathrm{N}^+$, each node i reads its current energy $e_i(t_k)$, then

$$l_i^{\max}(k^-) = \min\{l_i^{\max}(k-1), \left\lfloor \frac{e_i(t_k)}{g_i(t_{th}^m)} \right\rfloor\},$$
$$\tilde{l}_i^{\max}(k^-) = \min\{\tilde{l}_i^{\max}(k-1), \left\lfloor \frac{e_i(t_k)}{g_i(T-t_i^c)} \right\rfloor\}, \tag{25}$$

and

$$U_i^a(x,k^-) = U_i^a(x,k-1) + \varepsilon_i(x,k^-), \tag{26}$$

where $\varepsilon_i(x,k^-) = U_i(f(x,t_k))x - U_i(f(x,t_{k-1}))x$.

2:Broadcast $l_i^{\max}(k^-)$, $\tilde{l}_i^{\max}(k^-)$ and $U_i^a(x,k^-)$ to its neighbors.

3:If node i receives a packet from its neighbor nodes, then

$$l_i^{\max}(k) = \min_{j\in N_i} l_j^{\max}(k^-),$$
$$\tilde{l}_i^{\max}(k) = \min_{j\in N_i} \tilde{l}_j^{\max}(k^-), \tag{27}$$

and

$$U_i^a(x,k) = \sum_{j\in N_i} w_{ij} U_j^a(x,k^-), \tag{28}$$

where each w_{ij} is Metropolis weight **[19]**, which is given by

$$w_{ij} = \begin{cases} \frac{1}{1+\max\{d_i,d_j\}}, & i \neq j, \\ 1 - \sum_{j\in N_i, i\neq j} w_{ij}, & i = j. \end{cases} \tag{29}$$

4:Solve equation $\frac{\partial U_i^a(x,k)}{\partial x} = 0$ to obtain the solution $x_i(k)$. Then,

1) if $\tilde{l}_i^{\max}(k) \leq x_i(k) \leq l_i^{\max}(k)$, then

$$t_i^m = \frac{\frac{e_i(t_k)}{\arg\max_{x\in\Omega_i(k)} U_i^a(x,k)} - \beta_i}{\alpha}, \tag{30}$$

where

$$\Omega_i(k) = \{\lfloor x_i(k) \rfloor - 1, \lfloor x_i(k) \rfloor, \lceil x_i(k) \rceil, \lceil x_i(k) \rceil + 1\} \tag{31}$$

2) if $x_i(k) > l_i^{\max}(k)$, then $t_i^m = \frac{\frac{e_i(t_k)}{l_i^{\max}(k)} - \beta_i}{\alpha}$.

5:Store $l_i^{\max}(k)$, $\tilde{l}_i^{\max}(k)$, $U_i(f(x,t_k))$ and $U_i^a(x,k)$.

Let $U(k) = [U_1^a(x,k), U_2^a(x,k), \ldots, U_N^a(x,k)]^T$ and $\varepsilon(k-1) = [\varepsilon_1(x,k), \varepsilon_2(x,k), \ldots, \varepsilon_N(x,k)]^T$. By ACDA, we have

$$U(k) = W\left(U(k-1) + \varepsilon(k-1)\right), \tag{32}$$

where W is a matrix with the Metropolis weights. Note that when the energy consumption during each iteration time interval $[t, t+\tau]$ is ignored, i.e., $\int_t^{t+\Delta t} |h_i(\tau)| d\tau = 0, i \in V$, we have $\varepsilon(k-1)=0$. Since W is a doubly stochastic matrix (it is a square matrix of nonnegative real numbers, with each of rows and columns sum to be 1), according to the classical theory of consensus [21], it has

$$\lim_{k\to\infty} U(k) = \frac{1}{N}\sum_{i\in V} U_i^a(x,0)\mathbf{1}, \tag{33}$$

with an exponential convergence speed, where $\mathbf{1} = [1,1,\ldots,1]^T$. However, if the energy consumption of the algorithm is considered, i.e., $\int_t^{t+\Delta t} |h_i(\tau)| d\tau > 0, i \in V$, then (32) is a dynamic average consensus. For this case, we give the following theorem, which gives an upper bound of the differences between nodes' states, where the proof of the theorem is inspired by the stability analysis given in [30], [31].

Theorem 5.4: Suppose that G is connected and $\| \varepsilon(k) \|_2 \leq \epsilon$, where ϵ is a positive constant, using ACDA, we have

$$\| U(k) - \bar{U}(k) \|_2 \leq \lambda_2^k \| U(0) \|_2 + \frac{\lambda_2 - \lambda_2^{k+1}}{1-\lambda_2}\epsilon, \tag{34}$$

where $\bar{U}(k) = \frac{\mathbf{1}\mathbf{1}^T U(k)}{N}$ is the average state of all nodes and λ_2 is the second largest eigenvalue of W.

Proof: The proof is provided in the Appendix E. ■

Note from [22] that $|\lambda_2| < 1$, taking limitation on both sides of (34), we have

$$\lim_{k\to\infty} \| U(k) - \bar{U}(k) \|_2 \leq \frac{\lambda_2}{1-\lambda_2}\epsilon. \tag{35}$$

Thus, with the iteration increasing, each $U_i^a(x,k)$ will become closer to the real $\frac{1}{N}\sum_{i\in V} U_i^a(x,k)$, and the difference between them is bounded by $\frac{\lambda_2}{1-\lambda_2}\epsilon$. Note that ϵ is the upper bound of $\| \varepsilon(k) \|_2$. Fortunately, we can set a small Δt such that $\| \varepsilon(k) \|_2$ is small enough (while it should ensure that sensors can communicate with each other at lease once in such a small time interval Δt), and then ϵ can be a small value. Hence, a smaller setting of Δt helps decrease the error between each $U_i^a(x,k)$ and $\frac{1}{N}\sum_{i\in V} U_i^a(x,k)$ of ACDA. Note that (34) guarantees that ACDA can converge in a small error bound under the condition that $\| \varepsilon(k) \|_2 \leq \epsilon$ which has taken energy variation into consideration.

Secondly, consider a short lifetime regime. Since $L_{\min}(t)$ and $\tilde{L}(t)$ can be obtained from Algorithm 1, we assume these two variables are known to each node in this case. If we have obtained the optimal solution for each node from step 4 of Algorithm 1, then problem P_2 can be solved; otherwise, we give the following iteration equation, which is named ACDA algorithm to handle this short time regime case,

$$F_i^\ell(k) = \sum_{j\in N_i} w_{ij}\left(F_j^\ell(k-1) + \delta_j^\ell(k-1)\right), \ell = 1,..,m, \tag{36}$$

with initial conditions $F_i^\ell(0) = U_i^\ell(0), i \in V$, and where $\delta_j^\ell(k-1) = U_i^\ell(t_k) - U_i^\ell(t_{k-1})$ and m is a number of integers in $[L_{\min}(t), \tilde{L}(t)]$. Note that (36) is also a dynamic average consensus algorithm, its convergence can be guaranteed by the similar theoretical analysis as Theorem 5.4. Especially, when $h_i(t) = 0, i \in V$, we have $\lim_{k\to\infty} F_i^\ell(k) = \frac{1}{N}\sum_{i=1}^N U_i^\ell(0)$ for $\ell = 1,2,..,m$. Therefore, each node can obtain the $L^*(t)$ for problem P_2 by comparing the values of $\lim_{k\to\infty} F_i^\ell(k)$ for $\ell = 1,2,..,m$, i.e., $L^*(t) = L_{\min}(t) + \arg\{\max_{\ell=1,2,..,m} \lim_{k\to\infty} F_i^\ell(k)\} - 1$, and then obtain the optimal solution as (22). Since theoretical results and algorithm design are similar to that of the medium lifetime regime case, we omit the details. The main difference for handling these two cases is the computation time on average consensus. For short lifetime regime, since each node i needs to calculate F_i^ℓ at each iteration k for $\ell = 1,..,m$, there are m variables needed to calculate at each iteration. In the medium lifetime regime, it follows from (28) that there is only one variable needed to calculate at each iteration.

Remark 5.5: It should be pointed out that we can use the above approach to solve problem P_2 directly, i.e., by comparing the maximum network utilities under all the possible values of $L(t)$ for $L(t) = [L_{\min}(t), L_{\max}(t)]$ to get the optimal network lifetime. However, it may increase the computation and storage for sensors compared to handling these two cases separately. Note that there may be many more integers in $L(t) = [L_{\min}(t), L_{\max}(t)]$ than those in $[L_{\min}(t), \tilde{L}(t)]$, and thus the value of m in (36) becomes much larger. Specifically, when we handle these two cases separately, the computation time on average consensus for each node is $\tilde{L}(t) - L_{\min}(t) + 2$ at each iteration; while it becomes $L_{\max}(t) - L_{\min}(t) + 1$ when we solve problem P_2 directly. Hence, handling these two cases separately can reduce $L_{\max}(t) - \tilde{L}(t) - 1$ computation time at each iteration. Since

$$L_{\max}(t) - \tilde{L}(t) = \min_{i\in V}\left\lfloor \frac{e_i(t)}{g_i(t_{th}^m)} \right\rfloor - \max_{i\in V}\left\lfloor \frac{e_i(t)}{g_i(T - t_i^c)} \right\rfloor \gg 1 \tag{37}$$

generally holds true when $t_{th}^m \ll T - t_i^c$, handling this two cases separately can largely save the computation time.

VI. Evaluation

In this section, we evaluate the performance of proposed algorithms. All the results were obtained with Matlab 7.0.

We simulated a WSN composed of 50 sensors that were randomly distributed in an $100\times 100\ \mathrm{m}^2$ area, and the communication distance for each node was 20 m. To guarantee proportional fairness [32], the utility function is given by

$$U_i(t_i^m) = \omega_i \ln t_i^m, i \in V, \tag{38}$$

where $\omega_i, i \in V$ are positive constants, which can be deemed as different weights on the utilities of the nodes. From a Tmote Sky Datasheet [33], we have per time energy cost for communication, sensing and sleep modes respectively satisfy $p^c = 63$ mW (Milliwatts), $p^m = 5.4$ mW, $p^s = 0.06$ mW, and the battery voltage is 3 V (Volt). Each initial energy for each node i is randomly selected from $[1800, 2300]$ mAh (Milliampere-hour), i.e., $e_i(0) \in [1800, 2300]$ mAh for $i \in V$. We set the length of each working period $T = 24$ h, the minimum required sensing time $t_{th}^m = 1$ h, and the communication time $t_i^c = 0.5$ h for $i \in V$. Hence, we have $\alpha = 5.34$ and $\beta_i = 22.41$ for (4).

A. Network Utility Under Different Lifetime Setting

We observed that network utility is maximized under optimal scheduling and is much larger than that under the maximum network lifetime constraint in (10). Fig. 3 shows how the network utility changes under feasible different network lifetime settings of problem P_1, which means that under the lifetime setting the sensing time of each node satisfies (1). Fig. 3(a) shows

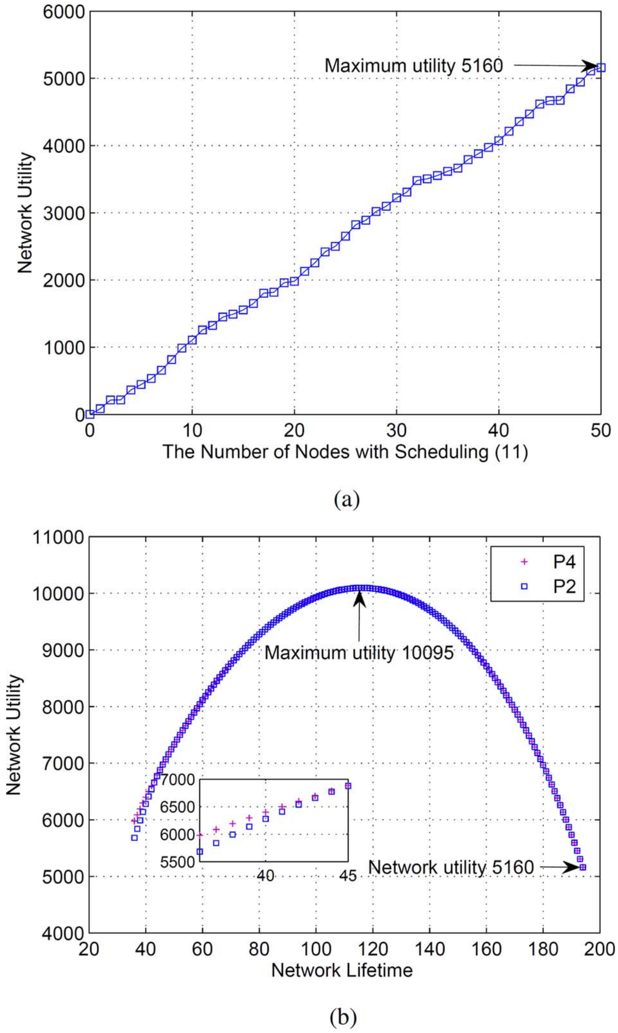


Fig. 3. The network utility for different problems under different lifetime requirements.

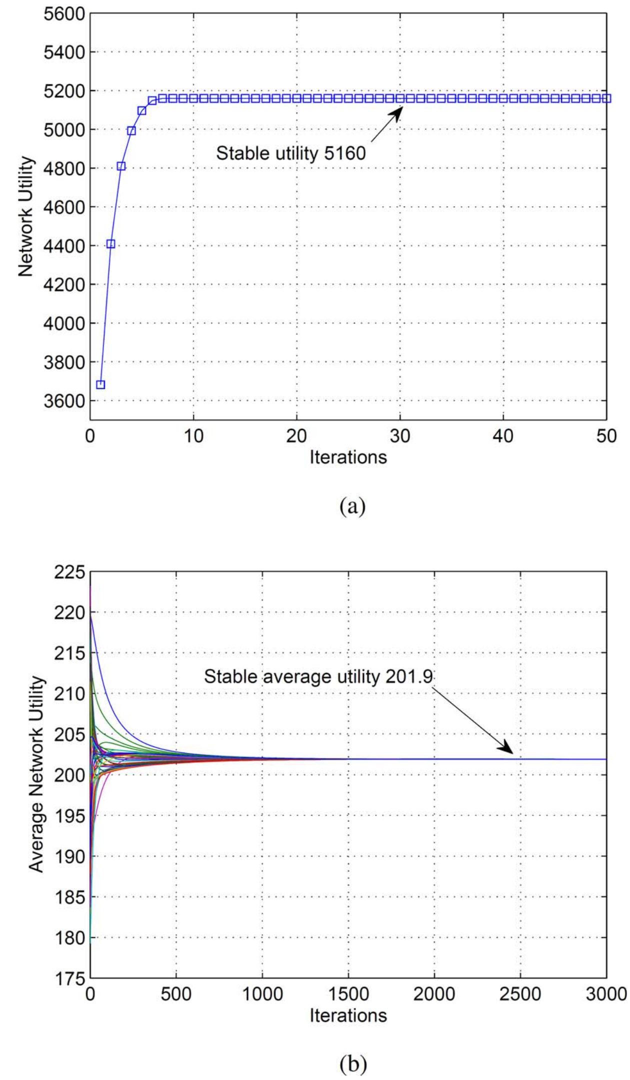


Fig. 4. The performance of MCDA and ACDA under the setting of $\int_t^{t+\Delta t} |h_i(\tau)| d\tau = 0$, i.e., battery variation is ignored. (a) The network utility under MCDA, (b) The average utility $U_i^a(x, k)$ under ACDA for each node i.

that the relationship between network utility and the number of nodes with scheduling (12) when the network lifetime always equals to $L_{\max}$. It is observed that when all nodes with their sensing time satisfy (12), the network utility is at maximum, $5,160$, i.e., the network utility is maximized when all nodes set their sensing time equal to the maximum values among their feasible intervals under the maximum lifetime constraint, which supports the results given in Theorem 4.3. Then, Fig. 3(b) shows the network utility under different network lifetime settings for the objective functions of problems P_2 and P_4, where the network lifetime $\tilde{L}(t)$ is 45. The network utility under each network lifetime for problem P_2 is the optimal utility obtained from (18). We can see that the network utility is a concave function of network lifetime and different network lifetime settings can get different network utilities. Specifically, when the network lifetime is in $[\tilde{L}(t), L_{\max}(t)]$, problem P_2 and P_4 have the same network utility, and the network utility is maximized when $L(t) = 116$ and the optimal network utility is 10,095 which is much larger than 5160 (the optimal network utility when $L(t) = L_{\max}(t) = 194$). It is also observed that the optimal network utility of problem P_4 equals that of problem P_2 when the maximum network utility is in $[\tilde{L}(t), L_{\max}(t)]$, which supports Theorem 5.2.

B. Evaluation of Distributed Algorithms

We observed that MCDA obtains a global optimal solution under the maximum lifetime constraint in a finite time, and ACDA obtains a global optimal solution for the multi-period scheduling problem with an exponential convergence speed: Considering the performance of the proposed algorithms, MCDA and ACDA, we studied whether the global optimal solutions of problem P_3 and P_2 can be obtained from applying MCDA and ACDA, respectively. Firstly, when the energy consumption of the algorithms is omitted, i.e., $\int_t^{t+\Delta t} |h_i(\tau)| d\tau = 0$ holds before the algorithm converges, Fig. 4 shows that the network utility changes with iterations. It is observed from Fig. 4(a) that with MCDA the network utility of problem P_3 can be maximized within finite iteration times and the stable utility equals 5160 which is the same optimal utility as that obtained in Fig. 3(a). And, it is seen from Fig. 4(b) that with ACDA the nodes' estimation of the average optimal network utility will converge with an exponential speed, where each line in the figure denotes the change of the average utility $U_i^a(x, k)$ under ACDA for node i, and at the stable state (the utility no longer

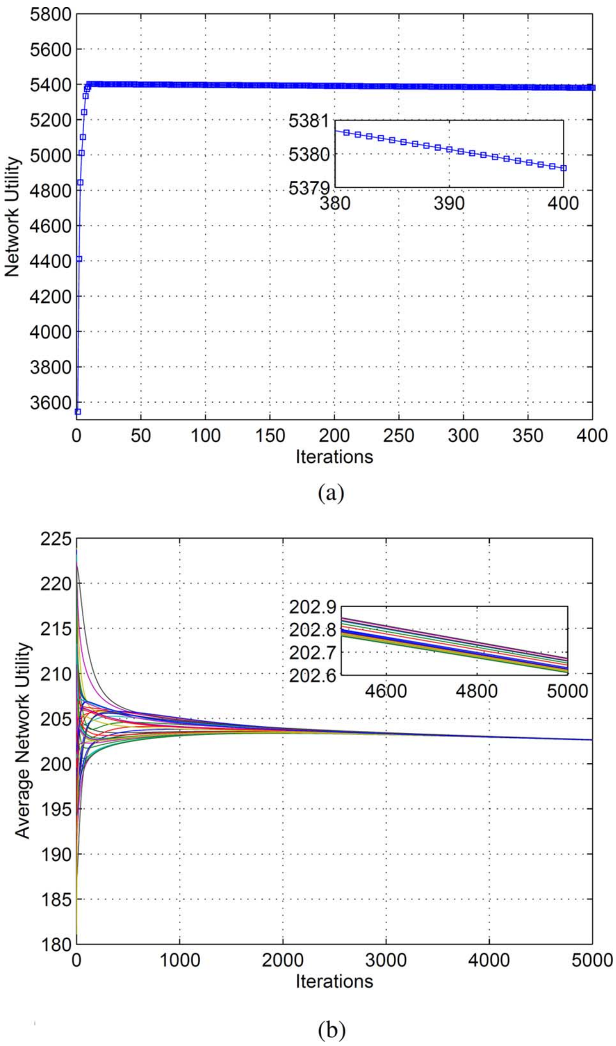


Fig. 5. The performance of MCDA and ACDA under the setting of $\int_t^{t+\Delta t} |h_i(\tau)| d\tau = 0.01$, i.e., battery variation is considered. (a) The network utility under MCDA, (b) The average utility $U_i^a(x, k)$ under ACDA for each node i.

change with iterations), the average utility of each node equals 201.9, which is exactly $\frac{10095}{50} = 201.9$. That is, the optimal network utility shown in Figs. 3(b) and 4(b) are equal, which means that the scheduling obtained by ACDA is the global optimal scheduling. Thus, the optimal solution of problem P_3 and P_2 can be obtained from MCDA and ACDA, respectively. It should be pointed out that although it needs more than 500 (seen from Fig. 4(b)) iterations for ACDA to converge, ACDA has an exponential convergence speed and thus can converge fast in a given small error bound. We can set a small iteration step size time, e.g., 100 ms (which should be guaranteed once there is communication exchange between neighbors), such that the algorithm can converge more quickly. We can also run the algorithm in an additional communication phase.

We observed that both MCDA and ACDA are robust against battery variation: Considering the energy consumption of the algorithms and setting $h_i \Delta t = 0.01$, i.e., the energy cost at each iteration is 0.01, Fig. 5 shows the dynamics of network utility with iterations of both algorithms. It is observed that both MCDA and ACDA converge fast such that the differences between nodes are less than a given small bound in (34). This is consistent with the results in Theorem 5.4 that ACDA cannot converge completely. However, it is clear to see that the network utility may decrease with iterations at the stable state for both algorithms, as the energy consumption of the algorithms is increased with iterations that will decrease network utility and network lifetime. It is observed from Fig. 5 that both MCDA and ACDA can retain the difference between nodes in a small bound. Thus, the scheduling obtained by both of these algorithms is very close to the optimal scheduling.

VII. Conclusions

By considering energy constraints and periodic sensing requirements, this paper has investigated a multi-period scheduling for NUM in WSNs, which introduces challenging mixed-integer programming. By using the properties of the convex objective function and the linear energy cost function, we first transfer the multi-period problem to an equivalent but simplified single-period problem. Then, we find that once the network lifetime is given, the optimal solution is that each node evenly allocates its energy to each period for sensing. Based on this observation, the MUM problem is further simplified to a pure-integer programming problem that can then be easily solved in a centralized way. Meanwhile, two novel consensus-based algorithms, MCDA and ACDA, were proposed to solve these problems in a completely distributed way. These algorithms have an exponential convergence speed and are robust against time-varying network lifetime.

With the consideration of both sensing time and communication time in multi-period scheduling for a NUM, the objective function depends not only on the sensing time but also on the communication time. The linear energy cost function may not be a linear function as the communication time is not a linear function of the sensing time. Then, the objective function and energy cost function become more complex and the proposed approach is invalid. We leave this more challenging problem as a future work.

Appendix A
Proof for Theorem 4.1

Proof: Note that problem P_1 is more general than problem P_2. Therefore, if we prove that the optimal solution of problem P_1 is also the optimal solution of problem P_2, the equivalence between P_1 and P_2 is proved.

Denote the optimal sensing sequence of problem P_1 as $t_i^m(k), k = 1, 2, \ldots, L(t)$. Let $t_i^m = \frac{\sum_{k=1}^{L(t)} t_i^m(k)}{L(t)}$. Since each $t_i^m(k)$ satisfies (5), we have $g_i(t_i^m)L(t) = \sum_{k=1}^{L(t)} g_i(t_i^m(k)) \leq e_i(t), i \in V$, i.e, t_i^m also satisfies (5). Since t_i^m is the average of $\frac{\sum_{k=1}^{L(t)} t_i^m(k)}{L(t)}$ and each $t_i^m(k)$ satisfies (1), we also have that every t_i^m satisfies (1). Hence, every t_i^m satisfies the constrains given in problem P_1, which means that every t_i^m is in the feasible region of this problem. By using Jensen's inequality, one obtains that $U_i(t_i^m)L(t) \geq \sum_{k=1}^{L(t)} U_i(t_i^m(k))$. Meanwhile, since the sensing sequence $t_i^m(k), k = 1, 2, \ldots, L(t)$, is the optimal solution of problem P_1, $\sum_{k=1}^{L(t)} U_i(t_i^m(k)) \geq U_i(t_i^m)L(t)$. Thus, we have $\sum_{k=1}^{L(t)} U_i(t_i^m(k)) = U_i(t_i^m)L(t)$, which means that $t_i^m(k) = t_i^m$ holds for $k = 1, 2, \ldots, L(t)$.

Then, we prove that every sensing sequence $t_i^m(k), k = 1, 2, \ldots, L(t)$, given above is also the optimal solution of problem P_2. We prove it by contradiction. We have proved that $t_i^m(k) = t_i^m$ holds for $k = 1, 2, \ldots, L(t)$, thus we infer that $\sum_{i\in V} U_i(t_i^m)L(t)$ achieves the maximum value under the constrains of problem P_1. Hence, if there exists $L(t)$ satisfying (6), i.e., $L(t) = \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor$, then $\sum_{i\in V} U_i(t_i^m)L(t)$ achieves the maximum value under the constrains of problem P_2, i.e., $t_i^m = \frac{\sum_{k=1}^{L(t)} t_i^m(k)}{L(t)}$ is the optimal solution of problem P_2. Since the t_i^m satisfies condition (5), we have $g_i(t_i^m)L(t) \le e_i(t)$ holds for $i \in V$, i.e., $L(t) \le \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor$. If $L(t) < \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor$, we can increase the value of each t_i^m such that $L(t) = \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor$, which will increase the value of $\sum_{i\in V} U_i(t_i^m)L(t)$. It is a contradiction since $\sum_{i\in V} U_i(t_i^m)L(t)$ achieves the maximum value. Hence, $L(t) = \min_{i\in V} \left\lfloor \frac{e_i(t)}{g_i(t_i^m)} \right\rfloor$, i.e., $L(t)$ satisfies (6). ■

Appendix B
Proof for Theorem 4.4

Proof: From (13) and (14), one infers that

$$l_i(k) = \min_{j\in N_i}\{l_j(k-1), \left\lfloor \frac{e_j(t_k)}{g_j(t_{th}^m)} \right\rfloor\}, i \in V. \quad (39)$$

According to (39), each $l_i(k)$ is a decreasing function of iteration k, the convergence time of the minimum consensus is $N-1$, when $k \ge N$, we have

$$l_i(k) \le \min_{j\in V}\{l_j(k-m)|m = n-1, n, \ldots, k\}, \quad (40)$$

holds for $i \in V$. From $h_i(t) \le \frac{g_i(t_{th}^m)}{2n\Delta t}$, we infer that

$$\int_t^{t+m\Delta t} |h_i(\tau)|d\tau \le g_i(t_{th}^m), \forall\, 0 < m \le 2N \quad (41)$$

holds for $i \in V$, which means that

$$\left\lfloor \frac{e_i(t_k)}{g_i(t_{th}^m)} \right\rfloor - 1 \le \left\lfloor \frac{e_i(t_{k+m})}{g_i(t_{th}^m)} \right\rfloor, \forall\, 0 < m \le 2N. \quad (42)$$

Thus, for (39), we have

$$\min_{i\in V}\{l_i(t_{k+m})\} \ge \min_{i\in V}\{l_i(t_k)\} - 1, \forall\, 0 < m \le 2N. \quad (43)$$

From (40) and (43), it follows that when $\forall k \ge N-1$, we have $|l_i(k) - l_j(k)| \le 1$ holds for $i, j \in V$.

Meanwhile, from (43),

$$\min_{i\in V}\{l_i(t_m)\} \ge \min_{i\in V}\{l_i(t_0)\} - 1, 0 < m \le 2N. \quad (44)$$

If $\min_{i\in V}\{l_i(t_{N-1})\} = \min_{i\in V}\{l_i(t_0)\}$, it can be deduced from (40) that $\hat{k} = N-1$. Otherwise,

$$\min_{i\in V}\{l_i(t_m)\} = \min_{i\in V}\{l_i(t_0)\} - 1, n-1 \le m \le 2N, \quad (45)$$

and then it infers from (40) that $\hat{k} = 2N$. ■

Appendix C
Proof for Theorem 5.1

Proof: Taking the derivative of the objective function (23) over the variable x, we have

$$\begin{aligned} &\frac{\partial \sum_{i\in V} U_i(f_i(x,t))x}{\partial x} \\ &= \sum_{i\in V}\left(U_i(f_i(x,t)) + \frac{\partial U_i(f_i(x,t))}{\partial f_i(x,t)}\frac{\partial f_i(x,t)}{\partial x}x\right) \\ &= \sum_{i\in V}\left(U_i(f_i(x,t)) - U_i'(f_i(x,t))\frac{e_i(t)}{\alpha x}\right) \end{aligned} \quad (46)$$

and

$$\begin{aligned} &\frac{\partial^2 \sum_{i\in V} U_i(f_i(x,t))x}{\partial x^2} \\ &= \sum_{i\in V}\frac{\partial\left(U_i(f_i(x,t)) - U_i'(f_i(x,t))\frac{e_i(t)}{\alpha x}\right)}{\partial x} \\ &= \sum_{i\in V}\left(U_i''(f_i(x,t))\frac{e_i^2(t)}{\alpha x^3} + U_i'(f_i(x,t))\frac{e_i(t)}{\alpha x^2}\right) \\ &\quad - \sum_{i\in V} U_i'(f_i(x,t))\frac{e_i(t)}{\alpha x^2} \\ &= \sum_{i\in V} U_i''(f_i(x,t))\frac{e_i^2(t)}{\alpha x^3}. \end{aligned} \quad (47)$$

Since each U_i is a concave function and $U_i'' < 0$, it follows

$$\frac{\partial^2 \sum_{i\in V} U_i(f_i(x,t))x}{\partial x^2} < 0, \forall x > 0. \quad (48)$$

Hence, problem P_4 is a concave optimization problem. Based on the convex optimization theorem [18], there is a unique optimal solution for problem P_4. ■

Appendix D
Proof for Theorem 5.2

Proof: Since x_t^* is the optimal solution of problem P_4,

$$\begin{aligned} &\max_{x\in \mathrm{N}^+} \sum_{i\in V} U_i(f_i(x,t))x \\ &= \max_{x=\lfloor x_t^*\rfloor, \lceil x_t^*\rceil} \sum_{i\in V} U_i(f_i(x,t))x. \end{aligned} \quad (49)$$

For problem P_2, as each $U_i(t_i^m)$ is an increasing function and t_i^m satisfies $t_i^m \le f_i(L(t), t)$, it follows that

$$\sum_{i\in V} U_i(t_i^m)L(t) \le \sum_{i\in V} U_i(f_i(x,t))x, \quad (50)$$

holds for $x = L(t)$. Hence, when $x_t^* \ge \tilde{L}(t)$, we infer that for problem P_2, when $L(t) \in [L_{\min}(t), \tilde{L}(t)]$, its objective function satisfies

$$\sum_{i\in V} U_i(t_i^m)L(t) \le \sum_{i\in V} U_i(f_i(\tilde{L}(t), t))\tilde{L}(t). \quad (51)$$

We have known that when $L^*(t) \in [\tilde{L}(t), L_{\max}(t)]$, problem P_2 can be transformed into problem P_4 with adding a condition $x \in \mathrm{N}^+$. Thus, when $x_t^* \geq \tilde{L}(t)$, it follows from (51) that $L^*(t) \in [\tilde{L}(t), L_{\max}(t)]$. Specifically, when $x_t^* \in [\tilde{L}(t), L_{\max}(t)]$, the $L^*(t)$ in problem P_2 should equal $\lfloor x_t^* \rfloor$ or $\lceil x_t^* \rceil$, the first result is obtained directly from (49) and (22). When $x_t^* > L_{\max}(t)$, note $L(t) \leq L_{\max}(t)$, we thus have the second result of this theory holds true. When $x_t^* < \tilde{L}(t)$, it infers that

$$\sum_{i \in V} U_i(f_i(\tilde{L}(t), t))\tilde{L}(t) \geq \sum_{i \in V} U_i(f_i(x, t))x, \tag{52}$$

holds for $x \in [\tilde{L}(t), L_{\max}(t)]$. Thus, the maximization of problem P_2 is achieved when $L(t) \in [L_{\min}(t), \tilde{L}(t)]$. ∎

Appendix E
Proof for Theorem 5.4

Proof: Based on (32), let $\mathbf{I}$ be the unit matrix, we have

$$\begin{aligned}
&\| U(k) - \bar{U}(k) \|_2 = \| (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})U(k) \|_2 \\
&= \| (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^k U(0) + \sum_{s=0}^{k-1} (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^{k-s}\varepsilon(s) \|_2 \\
&\leq \| (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^k U(0) \|_2 + \| \sum_{s=0}^{k-1} (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^{k-s}\varepsilon(s) \|_2 .
\end{aligned} \tag{53}$$

Using eigenvalue decomposition of matrix $W = S\Lambda S^{-1}$ with $\Lambda = \mathbf{diag}(1, \lambda_2, \ldots, \lambda_{\min})$, where S is unitary matrix, we have

$$\begin{aligned}
&\| (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^k U(0) \|_2 \leq \| (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^k \|_2 \| U(0) \|_2 \\
&\leq \| \mathbf{diag}(0, 1, \ldots, 1)\Lambda^k \|_2 \| U(0) \|_2 \leq \lambda_2^k \| U(0) \|_2
\end{aligned} \tag{54}$$

and

$$\begin{aligned}
&\| \textstyle\sum_{s=0}^{k-1} (\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{N})W^{k-s}\varepsilon(s) \|_2 \\
&\leq \textstyle\sum_{s=0}^{k-1} \| \mathbf{diag}(0, 1, \ldots, 1)\Lambda^{k-s} \|_2 \| \varepsilon(s) \|_2 . \\
&\leq \textstyle\sum_{s=0}^{k-1} \lambda_2^{k-s} \| \varepsilon(s) \|_2 \leq \frac{\lambda_2 - \lambda_2^{k+1}}{1-\lambda_2}\epsilon.
\end{aligned} \tag{55}$$

By combining (54) and (55), it yields (34). ∎

Acknowledgment

The authors would like to thank the associate editor Prof. Walaa Hamouda and the anonymous reviewers for their helpful and constructive comments, which have helped us improve the manuscript significantly.

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey, computer networks," *Comp. Netw.*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] S. Slawomir, M. Wiczanowski, and H. Boche, "Distributed utility-based power control: Objectives and algorithms," *IEEE Trans. Signal Process.*, vol. 55, no. 10, pp. 5058–5068, 2007.

[3] S. H. Low and E. L. David, "Optimization flow control-I: basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, pp. 861–874, 1999.

[4] J. Chen, W. Xu, S. He, Y. Sun, P. Thulasiraman, and X. Shen, "Utility-based asynchronous flow control algorithm for wireless sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 7, pp. 1116–1126, 2010.

[5] J. Chen, S. He, Y. Sun, P. Thulasiraman, and X. Shen, "Optimal flow control for utility-lifetime tradeoff in wireless sensor networks," *Comp. Netw.*, vol. 53, no. 18, pp. 3031–3041, 2009.

[6] C. K. Ho and R. Zhang, "Optimal energy allocation for wireless communications with energy harvesting constraints," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4808–4818, 2012.

[7] E. Sharanya, M. Archan, and P. Thomas, "Utility-based bandwidth adaptation in mission-oriented wireless sensor networks," *ACM Trans. Sens. Netw.*, vol. 8, no. 2, pp. 278–286, 2012.

[8] J. Zhu, S. Chen, B. Bensaou, and K. Hung, "Tradeoff between lifetime and rate allocation in wireless sensor networks: A cross layer approach," in *Proc. INFOCOM*, 2007.

[9] M. Chiang, "Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 104–116, 2005.

[10] D. P. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE J. Sel. Areas Commun.*, 2006.

[11] Atmega1281 datasheet [Online]. Available: http://www.atmel.com/dyn/resources/

[12] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. SenSys*, 2004.

[13] Q. Zhang, L. Fu, Y. Gu, Q. Cao, J. Chen, and T. He, "Collaborative scheduling in dynamic environments using error inference," *IEEE Trans. Parallel Distrib. Syst.*, 2013.

[14] R. Madan, Z. Q. Luo, and L. Sanjay, "A distributed algorithm with linear convergence for maximum lifetime routing in wireless networks," in *Proc. Allerton Conf. on Commun., Contr., Comput.*, 2005.

[15] L. Shi and H. Zhang, "Scheduling two Gauss-Markov systems: An optimal solution for remote state estimation under bandwidth constraint," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 2038–2042, 2012.

[16] L. Shi and L. Xie, "Optimal sensor power scheduling for state estimation of Gauss-Markov systems over a packet-dropping network," *IEEE Trans. Signal Process.*, vol. 60, no. 5, pp. 2701–2705, 2012.

[17] J. Wu, Y. Yuan, H. Zhang, and L. Shi, "How can online schedules improve communication and estimation tradeoff?," *IEEE Trans. Signal Process.*, 2013.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[19] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. IPSN*, 2005.

[20] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.

[21] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[22] A. Olshevsky and J. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Rev.*, vol. 53, no. 4, pp. 747–772, 2011.

[23] P. Braca, S. Marano, V. Matta, and P. Willett, "Asymptotic optimality of running consensus in testing binary hypotheses," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 814–825, 2010.

[24] M. Zhou, J. He, P. Cheng, and J. Chen, "Discrete average consensus with bounded noise," in *Proc. IEEE CDC*, 2013.

[25] L. Schenato and F. Fiorentin, "Average timesynch: A consensus-based protocol for time synchronization in wireless sensor networks," *Automatica*, pp. 1878–1886, 2011.

[26] J. He, P. Cheng, L. Shi, and J. Chen, "SATS: Secure average-consensus-based time synchronization in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 61, no. 24, pp. 6387–6400, 2013.

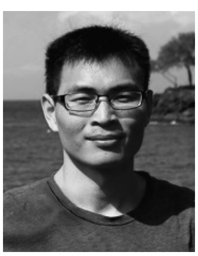
[27] J. He, P. Cheng, L. Shi, J. Chen, and Y. Sun, "Time synchronization in WSNs: A maximum-value-based consensus approach," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 660–675, 2014.

[28] S. Reddy and R. Chandra, "Dual-stage power management algorithms for energy harvesting sensors," *IEEE Trans. Wireless Commun.*, vol. 11, no. 4, pp. 1434–1445, 2012.

[29] G. Nemhauser and L. Wolsey, *Integer and Combinatorial Optimization*. New York, NY, USA: Wiley, 1988.

[30] C. Li and X. Liao, "Passivity analysis of neural networks with time delay," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 8, pp. 471–475, 2005.

[31] C. Li, L. Chen, and K. Aihara, “Stochastic stability of genetic networks with disturbance attenuation,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 10, pp. 892–896, 2007.
[32] J. Mo and W. Jean, “Fair end-to-end window-based congestion control,” *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, 2000.
[33] Tmote Sky Datasheet Moteiv , 2006 [Online]. Available: http://www.eecs.harvard.edu/konrad/projects/shimmer/references/tmote-sky-datasheet.pdf
[34] C. Hua and T. Yum, “Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks,” *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 892–903, 2008.
[35] M. Hajiaghayi, D. Min, and L. Ben, “Maximizing lifetime in relay cooperation through energy-aware power allocation,” *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4354–4366, 2010.
[36] A. Nedic, O. Asuman, and P. Pablo, “Constrained consensus and optimization in multi-agent networks,” *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 922–938, 2010.
[37] M. Rabbat and N. Robert, “Distributed optimization in sensor networks,” in *Proc. IPSN*, 2004.
[38] Y. C. Liang, Y. Zeng, E. C. Peh, and A. T. Hoang, “Sensing-throughput tradeoff for cognitive radio networks,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1326–1337, 2008.
[39] P. Braca, S. Marano, and V. Matta, “Enforcing consensus while monitoring the environment in wireless sensor networks,” *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3375–3380, 2008.
[40] H. Wymeersch, J. Lien, and M. Z. Win, “Cooperative localization in wireless networks,” *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, 2009.



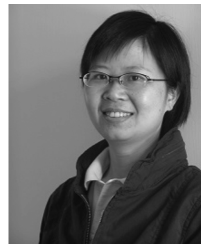
Jianping He received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China.

He is currently a Postdoctoral Research Fellow in both the State Key Laboratory of Industrial Control Technology at Zhejiang University and the Department of Electrical and Computer Engineering at University of Victoria. He is a member of Networked Sensing and Control group (Nesc). His research interests include the control and optimization in sensor networks and cyberphysical systems, the scheduling and optimization in vehicular networks and social networks, and the investment decision in financial market and electricity market.

Lingjie Duan (S'09–M'12) received the Ph.D. degree from The Chinese University of Hong Kong in 2012.

He is an Assistant Professor in the Engineering Systems and Design Pillar, Singapore University of Technology and Design (SUTD). During 2011, he was a Visiting Scholar in the Department of EECS, University of California at Berkeley. His research interests include network economics, resource allocation, and energy harvesting. He is now leading the Network Economics and Optimization Lab at SUTD.

Dr. Duan is the Finalist of Hong Kong Young Scientist Award 2014, and is the recipient of CUHK Global Scholarship for Research Excellence in 2011. He serves as a TPC Co-Chair of INFOCOM'2014 Workshop on GCCCN, Program Co-Chair of the ICCS'2014 Special Issue on Economic Theory and Communication Networks, and the Wireless Communication Systems (WCS) Symposium Co-Chair of IEEE ICCC 2015. He is also a TPC member for many top-tier conferences.



Fen Hou (M'10) received the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, Canada, in 2008.

She is an Assistant Professor in the Faculty of Science and Technology, University of Macau. Her research interests include resource allocation and protocol design in vehicular sensor network and participatory sensor networks, QoS provisioning for multimedia communications in cognitive radio network, and broadband wireless networks.

Dr. Hou was the recipient of IEEE Globecom Best Paper Award in 2010 and the Distinguished Service Award in IEEE MMTC in 2011. She has served as a Co-Chair in ICCS 2014 Special Session on Economic Theory and Communication Networks, Co-Chair in INFOCOM 2014 Workshop on Green Cognitive Communications and Computing Networks (GCCCN), and Co-Chair in IEEE Globecom Workshop on Cloud Computing System, Networks, and Application (CCSNA) 2013 and 2014, respectively. She served as a Technical Program Committee member for IEEE INFOCOM, WiOpt, ICC, WCNC, and Globecom.

Peng Cheng (M'10) received the B.E. degree in automation and the Ph.D. degree in control science and engineering in 2004 and 2009, respectively, both from Zhejiang University, Hangzhou, P.R. China.

Currently he is Associate Professor with the Department of Control Science and Engineering, Zhejiang University. His research interests include networked sensing and control, cyberphysical systems, and robust control.

Dr. Cheng serves as Associate Editor for *Wireless Networks*, *International Journal of Communication Systems*, and the *International Journal of Distributed Sensor Networks*. He is Local Arrangement Chair for IEEE MobiHoc 2015, and Publicity co-Chair for IEEE MASS 2013.

Jiming Chen (M'08–SM'11) received the B.Sc. and Ph.D degrees, both in control science and engineering, from Zhejiang University in 2000 and 2005, respectively.

He was a visiting researcher at INRIA in 2006, National University of Singapore in 2007, and the University of Waterloo from 2008 to 2010. Currently, he is a full professor with the Department of Control Science and Engineering, and the coordinator of group of Networked Sensing and Control in the State Key laboratory of Industrial Control Technology, Vice Director of Institute of Industrial Process Control at Zhejiang University, China.

Dr. Chen currently serves as Associate Editor for several international journals including the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE NETWORK, IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS. He was a Guest Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, *Computer Communication (Elsevier)*, *Wireless Communication and Mobile Computer (Wiley)*, and the *Journal of Network and Computer Applications (Elsevier)*. He also served/serves as *Ad hoc* and Sensor Network Symposium Co-chair, IEEE Globecom 2011; General Symposia Co-Chair of ACM IWCMC 2009 and ACM IWCMC 2010, WiCON 2010 MAC track Co-Chair, IEEE MASS 2011 Publicity Co-Chair, IEEE DCOSS 2011 Publicity Co-Chair, IEEE ICDCS 2012 Publicity Co-Chair, IEEE ICCC 2012 Communications QoS and Reliability Symposium Co-Chair, IEEE SmartGridComm The Whole Picture Symposium Co-Chair, IEEE MASS 2013 Local Chair, Wireless Networking and Applications Symposium Co-chair, IEEE ICCC 2013, *Ad hoc* and Sensor Network Symposium Co-Chair, IEEE ICC 2014 and TPC member for IEEE ICDCS'10, '12, '13, '14, IEEE MASS'10, '11, '13, IEEE SECON'11, '12, and IEEE INFOCOM'11, '12, '13, '14.