BUCKET: Scheduling of Solar-Powered Sensor Networks via Cross-Layer Optimization

Sungjin Lee, Beom Kwon, Sanghoon Lee, *Senior Member, IEEE*, and Alan Conrad Bovik, *Fellow, IEEE*

***Abstract*— Renewable solar energy harvesting systems have received considerable attention as a possible substitute for conventional chemical batteries in sensor networks. However, it is difficult to optimize the use of solar energy based only on empirical power acquisition patterns in sensor networks. We apply acquisition patterns from actual solar energy harvesting systems and build a framework to maximize the utilization of solar energy in general sensor networks. To achieve this goal, we develop a cross-layer optimization-based scheduling scheme called binding optimization of duty cycling and networking through energy tracking (BUCKET), which is formulated in four-stages: 1) prediction of energy harvesting and arriving traffic; 2) internode optimization at the transport and network layers; 3) intranode optimization at the medium access control layer; and 4) flow control of generated communication task sets using a token-bucket algorithm. Monitoring of the structural health of bridges is shown to be a potential application of an energy-harvesting sensor network. The example network deploys five sensor types: 1) temperature; 2) strain gauge; 3) accelerometer; 4) pressure; and 5) humidity. In the simulations, the BUCKET algorithm displays performance enhancements of ∼12–15% over those of conventional methods in terms of the average service rate.**

***Index Terms*— Binding optimization of duty cycling and networking through energy tracking (BUCKET), cross-layer optimization, equal duty cycle allocation (EDCA), energy-harvesting sensor network, inter- and intra-node optimization, maximization of the sum of the duty cycle (MSDC).**

I. Introduction

HEIGHTENED interest in green technologies that consider environmental and energy consumption is accelerating the growth of industries promoting and standardizing such technologies. Among renewable energy sources, solar energy is of particular interest owing to its low carbon profile and low maintenance requirement. Solar energy-harvesting systems are of particular interest for deployment in sensor networks, where they can replace conventional batteries. The solar solution is particularly appealing given interest in large-area deployment of sensor networks.

One problem complicating the use of solar energy in sensor networks is that the power acquisition pattern of the network is often irregular over time. The energy available from the solar panel at a given time may be lower than the energy required for network operation, or vice versa. Energy loss occurs when solar energy is applied to a resident battery; therefore, it is sensible to design harvesting power-aware protocols that enable a maximum volume of information at the sink node, using efficient energy utilization over the sensor network. Toward this end, we have designed an energy-efficient algorithm that we term BUCKET (Binding optimization of dUty Cycling and networKing through Energy Tracking), which uses a cross-layer optimization approach to maximize energy utilization in the presence of time-varying power acquisition patterns.

Several optimization frameworks have been proposed for energy-harvesting sensor networks. In [1] and [2], performance enhancement, expressed in terms of duty cycle, is investigated using Heliomote sensor-mote devices. The authors sought to maximize the total duty cycle assuming uniform traffic and point-to-point communication; however, these assumptions are nontrivial when extended to more general sensor networks that are characterized by variable traffic patterns, and that allow inter-networking among the sensor nodes.

Much of the current research holds limited opportunity for optimizing the efficiency of power acquisition from solar energy sources as a function of hardware architecture. Little attention appears to have been paid to the problem of inter-networking over transport, network, and medium-access-control (MAC) layers [3]–[5]. Here, we investigate an optimal methodology for maximizing energy utilization over solar energy-harvesting sensor networks. Toward this end, we apply and deploy a realistic model for harvesting power and data filtering and perform a numerical analysis under statistical traffic pattern and inter-networking assumptions.

Our primary contributions are expressed according to the following protocol layers:

- Transport Layer: The optimal source rate in the transport layer is determined, and a proportional fairness scheduling criterion is implemented, taking into account the power acquisition pattern.
- Network Layer: To maximize the network throughput under a limited harvesting energy constraint, we develop a multi-path routing optimization process and calculate an optimal flow rate for each multi-path route.
- MAC Layer: By estimating the harvesting power and arrival traffic pattern, we develop a duty cycle scheduling scheme based on intra-node optimization.

Manuscript received September 2, 2014; revised October 15, 2014; accepted October 15, 2014. Date of publication October 20, 2014; date of current version December 18, 2014. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2A10011764). This research was supported by the MSIP (Ministry of Science, ICT&Future Planning), Korea, under the C-ITRC (Convergence Information Technology Research Center) support program (NIPA-2014-H0401-14-1006) supervised by the NIPA (National IT Industry Promotion Agency). The associate editor coordinating the review of this paper and approving it for publication was Prof. K. Kim. *(Corresponding author: Sanghoon Lee.)*

S. Lee, B. Kwon, and S. Lee are with the Department of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea (e-mail: elflee7@yonsei.ac.kr; hsm260@yonsei.ac.kr; slee@yonsei.ac.kr).

A. C. Bovik is with the Laboratory for Image and Video Engineering, Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX 78712 USA (e-mail: bovik@ece.utexas.edu).

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/JSEN.2014.2363900

- Physical and Power Management Layer: We use an authentic solar panel harvesting power model and its acquisition pattern model.

The remainder of this paper is organized as follows. In Section II, we introduce previous related work on energy-harvesting sensor node networks. In Section III, the overall operation of our solar-powered cross-layer sensor network model is explained. Implementation of the BUCKET algorithm for real-time scheduling is discussed in Section IV. The core processes of the BUCKET algorithm: inter- and intra-node optimization are presented in Sections V and VI, respectively. Section VII demonstrates the performance enhancement attained through extensive simulation. Finally, we conclude the paper in Section VIII.

II. Related Work

A. Optimization of Energy-Harvesting Sensor Nodes

Previous studies have addressed the maximization of the efficiency of solar energy use via optimized hardware design, including the efforts in [5]–[7]. In each of these studies, a simple routing or scheduling algorithm for resource allocation is developed over a single layer, rather than over multiple layers [8], [9]. The authors in [1] and [2] explore the design of an optimal duty cycle implemented at the MAC layer based on energy-harvesting power measurements, restricting optimization to point-to-point communication at one harvesting sensor node, without considering inter-networking across the overall sensor network. The framework they developed seeks to maximize the simple total duty cycle over a uniform traffic environment, ignoring the important practical problem of multiple traffic types. We attempt to ameliorate the shortcomings of prior approaches by deriving a utility function that expresses the harvesting power and traffic over a solar energy-harvesting sensor network.

B. Optimization of the Energy-Harvesting Sensor Network

The topic of resource allocation in energy-limited sensor networks is vigorously studied via optimization theory. For example, to address battery limits at a sensor node, the authors in [10]–[12] deploy suboptimal algorithms that seek to maximize the network lifetime of all nodes based on load balancing; however, little substantial effort is made toward cross-layer optimization of energy-harvesting sensor networks from the application to physical layers. The algorithms explained in [10]–[12] are based on centralized control schemes operated by the cluster head, which collect all necessary related information from each node. This leads to an exponential increase in computational overhead with the number of nodes, making it difficult to find a practical, optimal solution.

To alleviate the concentration of energy consumption in a sensor node, the authors in [13]–[17] employ a distributed control mechanism in which, an optimal solution is found through layering of optimization decomposition or by allowing the exchange of information between nodes. The authors of [13] consider a flow conservation constraint without including the network lifetime based on the available energy of nodes; therefore, it is difficult to appropriately apply the distributed mechanism in [13] to an energy-limited sensor network. The authors in [14] propose a distributed flow control algorithm for information on given routes, which exhibits limited adaptability for performing dynamic networking based on available energy resources. In [15], a distributed routing and rate control algorithm is studied to maximize the sum of the log of source rates in rechargeable sensor networks. Here, the authors assume that the recharging energy of each node can be estimated with high accuracy without including environmental factors such as weather conditions, the historical information of recharging energy, and prediction error, as well as the duty cycle of each node.

The authors in [16] develop a distributed flow control and energy allocation algorithms in solar-powered sensor networks; however, the energy allocation algorithm is not tightly coupled with the flow control algorithm. The energy allocated to each node is determined simply based on the residual energy in the battery. The expectation of harvested energy is formed without considering the arrival rate of each node. In [17], a power allocation policy is studied to maximize the total system utility as a function of the average bit rate per link. The proposed policy is limited in that the asymptotic optimal solution can only be calculated when the ratio of the battery capacity to maximum transmission power is sufficiently large.

To overcome this, in [18], a joint-flow control, routing and energy management algorithm is proposed to maximize the sum of the utility functions in terms of the data rate of each node. In [19], the authors propose routing and scheduling algorithms that can dynamically adapt to variations in energy replenishment and arrival traffic as a function of changes in the network environment. The authors also provide bounds on node capacity using prior knowledge of the energy replenishment and packet arrival rates. However, the authors in [18] and [19] did not fully address the statistical information from the energy-harvesting or arrival traffic pattern; this information is vital in addressing the real-time arrival traffic and the wrong energy scheduling problem according to the prediction error of energy-harvesting. In addition, in the case of [19], it is difficult to adaptively determine the routing path because the flow rate is not controlled adaptively according to the channel state of a link.

Although some of these works [14]–[17] find solutions using message exchanges, the energy consumption needed for such message exchange is not taken into account. In addition, in [15], [16], [18], and [19], the prediction procedure for rechargeable energy is not presented, while the historical information of rechargeable energy is utilized without including environmental factors. To alleviate such limitations as the concentration of energy consumption to specific nodes, the unavailability of energy required to exchange control messages and failure to exploit the useful meteorological, geographical and platform related information that is required to predict the amount of rechargeable energy, we construct a distributed optimal control algorithm termed BUCKET that addresses these limitations using cross-layer optimization of energy-harvesting power in solar-based sensor networks.

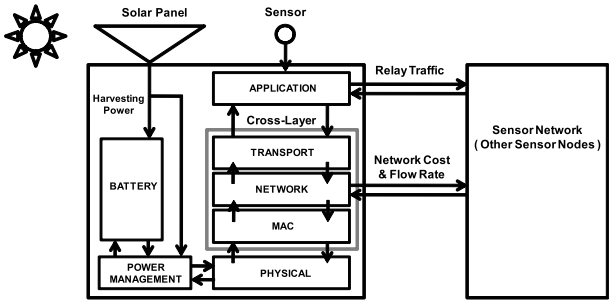


Fig. 1. Schematic diagram of cross-layer operation in a sensor node.

III. System Environment

A. Overview of Cross-Layer Operation

For multiple access control, the carrier sense multiple access with collision avoidance (CSMA/CA) protocol is employed for distributed networking. In addition, a synchronous duty cycle MAC protocol that is based on the CSMA/CA protocol for communication between nodes is deployed. Since the proposed scheme is directed toward energy scheduling in sensor networks, it is formulated using the superstructure of IEEE 802.15.4 WPAN.

Figure 1 depicts the signal flow over the protocol layers in a sensor node powered by the energy-harvesting system. Two traffic types are taken into account in the application layer: the traffic sensed by the sensor and the relay traffic transferred from neighboring nodes over the physical layer. This traffic is filtered according to the energy availability. To reduce redundancy, part of the filtered data may be dropped, thereby reducing the transmitted data volume and saving energy. In the scheme proposed here, the amount of traffic required to be transmitted per unit time is calculated in the application layer, and the traffic is transferred to the transport, network and MAC layers. In the MAC layer, an optimal duty cycle is determined based on both the amount of traffic transferred from the application layer and the harvesting power attained from the power management module during a unit time period. Using the duty cycle obtained from the MAC layer, the transport and network layers determine the source rate, flow rate and routing path. This process is fulfilled via exchange of network costs and flow rates among neighboring nodes. In this way, load balancing can be implemented as a function of the dynamic behavior of the harvesting power. Notice that these operations are tightly coupled functions of the harvesting power as viewed from a cross-layer design perspective. We therefore design a cross-layer optimization structure among the three layers: source rate control at the transport layer, flow rate control at the transport layer, and optimal duty cycling at the MAC layer.

B. Overall Problem Formulation

In the sensor network, it is assumed that sensor nodes transmit data to a sink node via a multi-hop strategy in order to achieve efficient conservation of energy. It is noted that sensor nodes near the sink node are severely constrained by energy, because they are to more frequently relay data received from other neighboring nodes to the sink node. Therefore, in order to maximize the sum of the source rates for given energy constraints of sensor nodes, it is necessary to provide more opportunities for sensor nodes near the sink node to transmit data to the sink node, due to its high throughput per energy. Conversely, sensor nodes located farther from the sink node have less opportunity to transmit data to the sink node. To ameliorate this unfairness, we employ the notion of proportional fairness [20]–[22] by using the sum of the log as an objective function of (1) for the optimization formulation, where the optimal flow and source rates are sought in a solar energy-harvesting sensor network in the form of

$$\begin{aligned}&\{\text{flow and source rates}\}^* \\ &= \arg\ \max \sum_{i \in network} \log(\text{source rate for node } i) \qquad (1)\\ &\text{subject to}\quad \text{network constraints for all links,}\\ &\qquad\qquad\ \ \text{a harvesting energy constraint for all nodes,}\\ &\qquad\qquad\ \ \text{an arrival traffic constraint for all nodes.}\end{aligned}$$

In (1), the network constraints reflect the flow and energy conservation between the receive and transmit data flows at each node; the harvesting energy constraint is obtained from the total usable energy recharged from the harvesting panel, and the arrival traffic constraint is related to the service rate needed according to the amount of data relayed from previous nodes.

C. Parameter Description

1) Timing Parameters: Figure 2 shows the four types of sensor nodes: a *source node* senses the event and transmits sensed data, a *relay node* receives the data and relays it to the next node, a *source relay node* plays the role of both sensing and relaying, while a *sleep node* does not participate in processing. In addition, Figure 2 depicts the hierarchical timing structure for energy scheduling based on periodic sensing. The timing parameters are defined as

- t: the time index for scheduling energy allocation.
- Δt: the time duration of each scheduling ("scheduling period").
- $\Delta\tau$: the fixed duration of sensing time ("sensing period").
- N_{sen}: the number of $\Delta\tau$ over Δt. ($N_{sen} = \Delta t/\Delta\tau$), which is equivalent to the number of sensing tasks being performed by node i over Δt from t.
- $\tau_u\langle i,t\rangle$: the active duration over $\Delta\tau$ from t in node i. The active duration relies on the set of tasks performed at each node on a multi-hop routing path. As shown in Figure 2, the sensing task is performed periodically during each active period. On the other hand, the tasks of receiving and transmitting communication can then be performed selectively according to the type of node.[1] With the exception of the sensing task, the set of tasks that can be composed of either a transmitting task or both receiving and transmitting tasks is termed a "comm task set" and is denoted by $\mathcal{T}_{com}$.
- $\tau_s\langle i,t\rangle$: the sensing duration in node i during $\Delta\tau$ at t.
- $\tau_r\langle i,t\rangle$: the duration for receiving data D in node i over $\Delta\tau$ at t. ($\tau_r\langle i,t\rangle = \frac{D}{R_{bw}} + c_r$, where R_{bw} is

[1] $\tau_u\langle i,t\rangle$ according to the type of node is derived in Eq. (3).

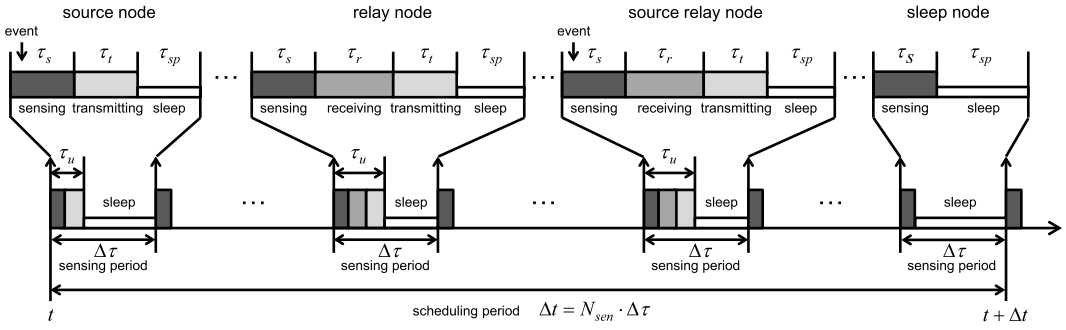


Fig. 2. Description of hierarchical timing management for real-time scheduling.

a bit rate[2] and c_r is a constant value for the state transition).

- $\tau_t\langle i,t\rangle$: the duration for transmitting data D in node i over $\Delta\tau$ at t. ($\tau_t\langle i,t\rangle = \frac{D}{R_{bw}} + c_t$, where c_t is a constant value for the state transition).
- $\tau_{sp}\langle i,t\rangle$: the sleep duration in node i over $\Delta\tau$ at t. ($\tau_{sp}\langle i,t\rangle = \Delta\tau - \tau_u\langle i,t\rangle$).
- $t_{act}\langle i,t\rangle$: the active duration in node i over Δt at t. ($t_{act}\langle i,t\rangle = \sum_{k=1}^{N_{sen}} \tau_u\langle i, t+(k-1)\cdot\Delta\tau\rangle$).

In Figure 2, each sensor node performs a sensing task at each $\Delta\tau$. When an event occurs during the sensing duration and sensor nodes on the routing path have enough energy to transmit sensed data to the sink node, the data is transmitted to neighboring nodes. Thus, even if the sensing task is periodically accomplished, data transmission is irregularly performed according to event detection and available energy of sensor nodes.

2) Parameters for Data-Flow Modeling: From the perspective of each node i, the transmitted data can be modeled as the sum of the sensed data and the data relayed from other nodes over the routing path using the following notations:

- $\mathcal{P}\langle i\rangle$: the set of nodes on the routing path during $\Delta\tau$ until node i is reached.
- $D_s\langle i,\tau\rangle$: the amount of data units sensed at node i during $\Delta\tau$ at τ, which is assumed to be a constant.
- $D_r\langle i,\tau\rangle$: the amount of data relayed from other nodes to node i during $\Delta\tau$ at τ through $\mathcal{P}\langle i\rangle\backslash\{i\}$, representing route nodes, except for node i.
- $D_t\langle i,\tau\rangle$: the amount of data transmitted from node i to the next node during $\Delta\tau$ at τ on the routing path.
- $\alpha\langle i,\tau\rangle$: a flag for whether node i acts as a *source node* during $\Delta\tau$ at time τ. For example, if node i is a *source node*, $\alpha\langle i,\tau\rangle = 1$. Otherwise, $\alpha\langle i,\tau\rangle = 0$.
- $\beta\langle i,\tau\rangle$: a flag for whether node i acts as a *relay node* during $\Delta\tau$ at time τ. For example, if node i is a *relay node*, $\beta\langle i,\tau\rangle = 1$. Otherwise, $\beta\langle i,\tau\rangle = 0$.

Using the parameters above, $D_r\langle i,\tau\rangle$ and $D_t\langle i,\tau\rangle$ can be respectively expressed as

$$D_r\langle i,\tau\rangle = \sum_{j\in\mathcal{P}\langle i\rangle\backslash\{i\}} \alpha\langle j,\tau\rangle \cdot D_s\langle j,\tau\rangle,$$

$$D_t\langle i,\tau\rangle = \sum_{j\in\mathcal{P}\langle i\rangle} \alpha\langle j,\tau\rangle \cdot D_s\langle j,\tau\rangle = D_r\langle i,\tau\rangle + D_s\langle i,\tau\rangle.$$

[2] R_{bw} is derived in Appendix A.

3) Parameters for Event Handling: To model the mechanism of event handling according to the type of node, the parameters are defined as

- $n_{ed}\langle i,t\rangle$: the number of events detected at node i over Δt from t, called the "event detection rate."
- $n_s\langle i,t\rangle$: the number of tasks for transmitting the events detected only at node i during Δt from t, called the "self-transmission rate." This rate is equivalent to the number of $\mathcal{T}_{com}$ over Δt, which depends on the available energy of node i.
- $n_r\langle i,t\rangle$: the number of tasks for transmitting the events received at node i from other nodes during Δt at t through $\mathcal{P}\langle i\rangle\backslash\{i\}$, called the "relay-transmission rate."
- $n_t\langle i,t\rangle$: the transmission rate being performed over Δt from t at node i. This rate is equivalent to the number of tasks for the previously mentioned self-transmission and relay-transmission rates.

The parameters $n_s\langle i,t\rangle$, $n_r\langle i,t\rangle$ and $n_t\langle i,t\rangle$ can be respectively represented by

$$\begin{aligned} n_s\langle i,t\rangle &= \sum_{k=1}^{N_{sen}} \alpha\langle i, t+(k-1)\cdot\Delta\tau\rangle, \\ n_r\langle i,t\rangle &= \sum_{j\in\mathcal{P}\langle i\rangle\backslash\{i\}} \sum_{k=1}^{N_{sen}} \alpha\langle j, t+(k-1)\cdot\Delta\tau\rangle \\ &= \sum_{j\in\mathcal{P}\langle i\rangle\backslash\{i\}} n_s\langle i,t\rangle, \\ n_t\langle i,t\rangle &= \sum_{j\in\mathcal{P}\langle i\rangle} \sum_{\tau=1}^{N_{sen}} \alpha\langle j, t+(k-1)\cdot\tau\rangle \\ &= \sum_{j\in\mathcal{P}\langle i\rangle} n_s\langle i,t\rangle = n_s\langle i,t\rangle + n_r\langle i,t\rangle. \end{aligned} \tag{2}$$

Figure 3 depicts the relationship among N_{sen}, $n_{ed}\langle i,t\rangle$ and $n_s\langle i,t\rangle$ according to the event detection and available energy of *source node* i.

4) Parameters for Energy Consumption Modeling: Each node consumes energy differently according to the combination of sensing, receiving and transmitting tasks. To derive the energy consumption, the parameters are denoted as

- $E_u\langle i,\tau\rangle$: the energy consumption of node i during the active duration at τ.
- $E_{act}(n_s\langle i,t\rangle)$: the energy consumption of node i for performing a sensing task, and the $\mathcal{T}_{com}$ over the active duration during Δt of t.

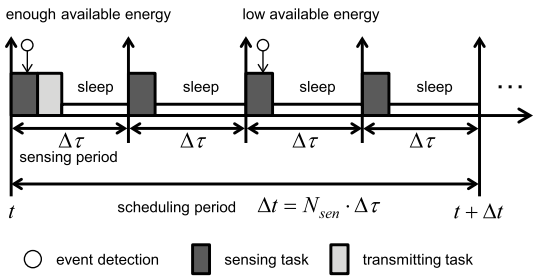


Fig. 3. An example showing the relationship among N_{sen}, $n_{ed}\langle i,t\rangle$ and $n_s\langle i,t\rangle$ according to the event detection and available energy of *source node* i. In this example, $N_{sen} = 4$, $n_{ed}\langle i,t\rangle = 2$ and $n_s\langle i,t\rangle = 1$.

- $E_{slot}\langle i,t\rangle$: the energy consumption including the sleep state of node i during Δt of t.

According to the value of $\beta\langle i,t\rangle$, the parameters $\tau_u\langle i,\tau\rangle$, $E_u\langle i,\tau\rangle$, $t_{act}\langle i,t\rangle$ and $E_{act}(n_s\langle i,t\rangle)$ are derived as follows.

$$\begin{aligned}
&\text{if } \beta\langle i,t\rangle = 0,\\
&\quad \tau_u\langle i,\tau\rangle = \tau_s + \alpha\langle i,\tau\rangle \cdot \tau_t,\\
&\quad E_u\langle i,\tau\rangle = P_s \cdot \tau_s + \alpha\langle i,\tau\rangle \cdot P_t \cdot \tau_t,\\
&\quad t_{act}\langle i,t\rangle = N_{sen} \cdot \tau_s + n_s\langle i,t\rangle \cdot \tau_t,\\
&E_{act}(n_s\langle i,t\rangle) = N_{sen} \cdot P_s \cdot \tau_s + n_s\langle i,t\rangle \cdot P_t \cdot \tau_t,\\
&\text{if } \beta\langle i,t\rangle = 1,\\
&\quad \tau_u\langle i,\tau\rangle = \tau_s + \tau_r + \tau_t,\\
&\quad E_u\langle i,\tau\rangle = P_s \cdot \tau_s + P_r \cdot \tau_r + P_t \cdot \tau_t,\\
&\quad t_{act}\langle i,t\rangle = N_{sen} \cdot \tau_s + n_r\langle i,t\rangle \cdot \tau_r + n_t\langle i,t\rangle \cdot \tau_t,\\
&E_{act}(n_s\langle i,t\rangle) = N_{sen} \cdot P_s \cdot \tau_s + n_r\langle i,t\rangle \cdot P_r \cdot \tau_r\\
&\qquad + n_t\langle i,t\rangle \cdot P_t \cdot \tau_t \qquad (3)
\end{aligned}$$

where P_s, P_r, P_t and P_{sp} are the sense, receive, transmit and sleep powers, respectively. In (3), for brevity, we use the simplified timing parameters $(\tau_s, \tau_r, \tau_t, \tau_{sp})$ after omitting the node and time indexes, $\langle i,t\rangle$, from $(\tau_s\langle i,\tau\rangle, \tau_r\langle i,\tau\rangle, \tau_t\langle i,\tau\rangle$ and $\tau_{sp}\langle i,\tau\rangle)$. In addition, $E_{slot}\langle i,t\rangle$ is modeled as

$$\begin{aligned}
E_{slot}\langle i,t\rangle &= \sum_{\tau=1}^{N_{sen}} \{E_u\langle i,\tau\rangle + \tau_{sp} \cdot P_{sp}\}\\
&= E_{act}(n_s\langle i,t\rangle) + \sum_{\tau=1}^{N_{sen}} \tau_{sp} \cdot P_{sp}.
\end{aligned}$$

Finally, from (3), the duty cycle in node i over Δt at t is defined as $\Phi\langle i,t\rangle = \frac{t_{act}\langle i,t\rangle}{\Delta t}$.

5) Energy Scheduling Based on Prediction: The energy consumption of a node depends on the tasks performed during the active duration. Therefore, it is necessary to schedule the performance of $\mathcal{T}_{com}$ based on the available energy of the node. Let $P_h\langle i,t\rangle$ be the harvesting power in node i at time t and $n_a\langle i,t\rangle$ be the arrival rate as the number of inputted $\mathcal{T}_{com}$ during Δt at t in node i.

$$n_a\langle i,t\rangle = n_{ed}\langle i,t\rangle + \sum_{j \in \mathcal{P}\langle i\rangle \setminus \{i\}} n_s\langle j,t\rangle.$$

At this time, only a few of the inputted $\mathcal{T}_{com}$ may be serviced according to energy availability, which becomes $n_t\langle i,t\rangle$. For scheduling $n_t\langle i,t\rangle$ at each scheduling time, t, $n_a\langle i,t\rangle$ should be predicted at each scheduling period, Δt, and $n_t\langle i,t\rangle$ should be determined based on the energy supplied from the energy-harvesting module.

Because events occur randomly, accurate prediction of $n_a\langle i,t\rangle$ is difficult. Accurate prediction of $P_h\langle i,t\rangle$ is also difficult since it depends on environmental conditions such as the weather, season and latitude. Let $\tilde{n}_{ed}\langle i,t\rangle$, $\tilde{n}_a\langle i,t\rangle$ and $\tilde{P}_h\langle i,t\rangle$ be the predicted version of $n_{ed}\langle i,t\rangle$, $n_a\langle i,t\rangle$ and $P_h\langle i,t\rangle$, respectively. Then, $\tilde{n}_{ed}\langle i,t\rangle$ is modeled using a statistical approach, $\tilde{n}_a\langle i,t\rangle$ is calculated using $\tilde{n}_{ed}\langle i,t\rangle$, and $\tilde{P}_h\langle i,t\rangle$ is modeled including significant environmental parameters. The detailed process for predicting $\tilde{P}_h\langle i,t\rangle$ is included in our prior work [23]. In addition, the event detection rate, $n_{ed}\langle i,t\rangle$, over a certain time interval is estimated using the probability mass function (PMF) and cumulative distribution function (CDF) of the event detection rate, and $\Delta n_d\langle i,t\rangle$ is the prediction error. The event detection rate can then be represented as

$$n_{ed}\langle i,t\rangle = \tilde{n}_{ed}\langle i,t\rangle + \Delta n_{ed}\langle i,t\rangle.$$

Let $n_{ed}\langle i,t,k\rangle$ be the event detection rate of node i in node set V at each scheduling time, t, in the k^{th} previous day and K be the set of measured days, i.e., $\forall k \in K$. By analyzing the frequency of occurrence of $n_{ed}\langle i,t,k\rangle$ at each time, t, over a large amount of experimental data, we can estimate the PMF, $f_{n_{ed}\langle i,t\rangle}(x)$, and the CDF, $F_{n_{ed}\langle i,t\rangle}(x)$, using

$$\begin{aligned}
f_{n_{ed}\langle i,t\rangle}(x) &= \Pr[n_{ed}\langle i,t\rangle = x] \approx \lim_{|K|\to\infty} \frac{N(n_{ed}\langle i,t,k\rangle = x)}{|K| \cdot |V|},\\
F_{n_{ed}\langle i,t\rangle}(x) &= \Pr[n_{ed}\langle i,t\rangle \le x] \approx \lim_{|K|\to\infty} \frac{N(n_{ed}\langle i,t,k\rangle \le x)}{|K| \cdot |V|},\\
&\qquad \forall x \in \mathbb{Z}_+, \forall k \in K, \forall i \in V \qquad (4)
\end{aligned}$$

where $n_{ed}\langle i,t\rangle$ is a random variable, x is a value of $n_{ed}\langle i,t\rangle$, $\mathbb{Z}_+$ is a non-negative integer, and $N(n_{ed}\langle i,t,k\rangle = x)$ is the counting function that indicates the number of $n_{ed}\langle i,t,k\rangle$ satisfying $n_{ed}\langle i,t,k\rangle = x$ for $\forall i, \forall k$. Similarly, $N(n_{ed}\langle i,t,k\rangle \le x)$ indicates the number of $n_a\langle i,t,k\rangle$ satisfying $n_{ed}\langle i,t,k\rangle \le x$ for $\forall i, \forall k$. Because $n_{ed}\langle i,t\rangle$ is a random variable for the number of occurrences, the domain of the PMF, $f_{n_{ed}\langle i,t\rangle}(x)$, is discrete with non-negative integers. In addition, because the PMF, $f_{n_{ed}\langle i,t\rangle}(x)$, is calculated by averaging the values aggregated from each node in node set V, the PMF, $f_{n_{ed}\langle i,t\rangle}(x)$, becomes identical over the node domain, i.e., $f_{n_{ed}\langle i,t\rangle}(x) = f_{n_{ed}\langle j,t\rangle}(x)$, $i, j, i \neq j \in V$.

The predicted event detection rate, $\tilde{n}_{ed}\langle i,t\rangle$, can then be obtained from the PMF, $F_{n_{ed}\langle i,t\rangle}(x)$, and its expectation obtained by

$$\begin{aligned}
E[n_{ed}\langle i,t\rangle] &= \sum_{x \in \mathbb{Z}_+} x \cdot f_{n_{ed}\langle i,t\rangle}(x)\\
&\approx \frac{1}{|K||V|} \sum_{j \in V} \sum_{k \in K} n_{ed}\langle j,t,k\rangle.
\end{aligned}$$

Some prediction errors may persist even with accurate prediction of $\tilde{n}_{ed}\langle i,t\rangle$ and $\tilde{P}_h\langle i,t\rangle$, complicating the allocation of harvesting energy in proportion to $\tilde{n}_a\langle i,t\rangle$ and leading to energy exhaustion in nodes. We propose a token-bucket-based scheduling algorithm to absorb such prediction errors. The token-bucket algorithm has been widely used for scheduling

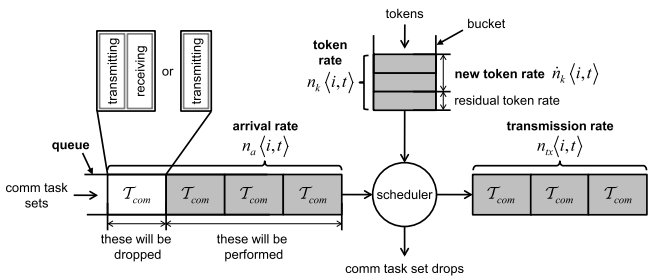


Fig. 4. Token-bucket schedule of the BUCKET algorithm.

in the areas of packet switched computing and telecommunication networking and uses the following steps.

- A token is added to the bucket at each time unit.
- The bucket can hold a limited number of tokens due to size constraints. If a token arrives when the bucket is full, it is discarded.
- When n_1 comm task sets arrive, n_1 tokens are removed from the bucket and their tasks are performed.
- If n_2 tokens are available ($n_2 < n_1$), only n_2 comm task sets among n_1 comm task sets are performed.

To facilitate the description of the token-bucket algorithm, let $n_k\langle i,t\rangle$ be the token rate determined based on $\tilde{n}_a\langle i,t\rangle$ and $\tilde{P}_h\langle i,t\rangle$ in each node. As shown in Figure 4, based on the three parameters, $n_a\langle i,t\rangle$, $n_k\langle i,t\rangle$ and $n_t\langle i,t\rangle$, the scheduling of the generated $\mathcal{T}_{com}$ is performed using the token-bucket algorithm.

In the token-bucket algorithm, when $n_a\langle i,t\rangle$ exceeds $n_k\langle i,t\rangle$, the generated $\mathcal{T}_{com}$ are serviced up to $n_k\langle i,t\rangle$, while the remaining comm task sets, $n_a\langle i,t\rangle - n_k\langle i,t\rangle$, are discarded. The sensor node then enters the sleep state and returns to the active state at the next sensing time. When $n_k\langle i,t\rangle$ exceeds $n_a\langle i,t\rangle$, the remaining tokens, $n_k\langle i,t\rangle - n_a\langle i,t\rangle$, are utilized at the next scheduling time, $t + \Delta t$. Hence, two separate token flows exist in the bucket: newly generated tokens at time t, named *new token rate* $\dot{n}_k\langle i,t\rangle$, and previous tokens before time t, named *residual token rate* $[n_k\langle i,t-\Delta t\rangle - n_a\langle i,t-\Delta t\rangle]^+$,

$$n_k\langle i,t\rangle = \overbrace{\dot{n}_k\langle i,t\rangle}^{\text{new token rate}} + \overbrace{[n_k\langle i,t-\Delta t\rangle - n_a\langle i,t-\Delta t\rangle]^+}^{\text{residual token rate}}. \tag{5}$$

When $\mathcal{T}_{com}$ are generated and placed in the queue, the scheduler services them with reference to the determined token rate, $n_k\langle i,t\rangle$, over t. Therefore, the transmission rate, $n_t\langle i,t\rangle$, can be derived as

$$n_t\langle i,t\rangle = \min\{n_a\langle i,t\rangle,\ n_k\langle i,t\rangle\}. \tag{6}$$

IV. Proposed Bucket Algorithm for Real-Time Scheduling

For brevity, we hereafter omit the time index, t, from the notations. In the token-bucket algorithm, an inappropriate determination of $n_k\langle i\rangle$ causes network performance degradation. In addition, in (1), the source rate is a function of $n_s\langle i\rangle$ while being controlled by $n_k\langle i\rangle$ in (2) and (6). Finding an optimal token rate, $n_k\langle i\rangle^*$, is therefore of vital importance.

To determine the optimal token, flow, and source rates, we propose the BUCKET algorithm composed of four steps, as shown in Figure 5, and using the following parameters.

Description of the BUCKET Algorithm Procedure

Step 1: Prediction
1: Predict the arrival rate, $n_a\langle i\rangle$.
2: Predict the harvesting power, $P_h\langle i\rangle$.
Step 2: Inter-node Optimization
1: Determine the optimal initial source and flow rates $(g^{(0)}\langle i\rangle^*,\ f_{(i,j)}^{(0)\,*})$, subject to network constraints.
2: Determine an optimal initial token rate, $\dot{n}_k^{(0)}\langle i\rangle^*$.
Step 3: Intra-node Optimization
1: Determine an optimal energy-scale factor, $\Gamma\langle i\rangle^*$, subject to the harvesting energy constraint.
2: Determine optimal scaled source and flow rates $(g^{(1)}\langle i\rangle^*,\ f_{(i,j)}^{(1)\,*})$.
3: Determine an optimal new token rate, $\dot{n}_k\langle i\rangle^*$.
Step 4: Token-bucket Schedule
1: Determine a transmission rate, $n_t\langle i\rangle$.
2: Determine an adaptation factor, $\Upsilon\langle i\rangle$.
3: Determine adapted source and flow rates $(g\langle i\rangle,\ f_{(i,j)})$ subject to the arrival traffic constraint.

Fig. 5. The pseudo code of the BUCKET algorithm.

- $\mathcal{I}(i)$: the set of neighboring nodes of node i for incoming flow rates.
- $\mathcal{O}(i)$: the set of neighboring nodes of node i for outgoing flow rates.
- $f_{(i,j)}\langle t\rangle$: the flow rate from node i to node j at t. The incoming rate of node i is $\sum_{j\in\mathcal{I}(i)} f_{(j,i)}\langle t\rangle = \sum_{j\in\mathcal{P}\langle i\rangle\setminus\{i\}} n_s\langle j,t\rangle \cdot D_s\langle j,\tau\rangle$, and the outgoing rate of node i is $\sum_{j\in\mathcal{O}(i)} f_{(i,j)}\langle t\rangle = \sum_{j\in\mathcal{P}\langle i\rangle} n_s\langle j,t\rangle \cdot D_s\langle j,\tau\rangle$.
- $g\langle i,t\rangle$: the source rate of node i at t ($g\langle i,t\rangle = n_s\langle i,t\rangle \cdot D_s\langle i,\tau\rangle$).
- $\Gamma\langle i,t\rangle$: the energy-scale factor of node i at t, which reflects the available energy of the node.
- $\Upsilon\langle i,t\rangle$: the adaptation factor of node i at t; a ratio of the transmission rate to the minimum token rate, reflecting the transmission rate bound by the minimum token rate.

As shown in Figure 5, the procedure of the BUCKET algorithm is as follows:

- Step 1 - Prediction: In order to optimize the harvesting sensor network, it is necessary to eventually predict $n_a\langle i\rangle$ and $P_h\langle i\rangle$. The prediction processes are explained in detail in Section III-C.5.
- Step 2 - Inter-node optimization: Initially, using the optimal source and flow rates ($g^{(0)}\langle i\rangle^*$, $f_{(i,j)}^{(0)\,*}$), the sum of the log of the source rates is maximized for all nodes under the network constraints. The optimal new token rate ($\dot{n}_k^{(0)}\langle i\rangle$) is also initially calculated based on the optimal initial source rate, $g^{(0)}\langle i\rangle^*$; however, the harvesting energy and arrival traffic constraints are ignored.
- Step 3 - Intra-node optimization: The energy-scale factor, $\Gamma\langle i\rangle$, is utilized to adjust the scale of the initial optimal

source, flow, and new token rates in accordance with the harvesting energy constraint. The optimal scaled source, flow and new token rates $g^{(1)}\langle i\rangle^*$, $f_{(i,j)}^{(1)\,*}$, and $\dot{n}_k\langle i\rangle^*$ are then updated from $g^{(0)}\langle i\rangle^*$, $f_{(i,j)}^{(0)\,*}$, and $\dot{n}_k^{(0)}\langle i\rangle^*$, respectively.

- Step 4 - Token-bucket schedule: As shown in Figure 4, when $\mathcal{T}_{com}$ are generated, they are addressed only up to $n_k\langle i\rangle$ in (5) determined at Steps 2 and 3. If $n_a\langle i\rangle \le n_k\langle i\rangle$, then $n_t\langle i\rangle = n_a\langle i\rangle$, and the remaining tokens, $n_k\langle i\rangle - n_a\langle i\rangle$, are utilized at the next scheduling time, $t + \Delta t$. If $n_a\langle i\rangle > n_k\langle i\rangle$, then $n_t\langle i\rangle = n_k\langle i\rangle$, and the unserviced comm task sets, $n_a\langle i\rangle - n_s\langle i\rangle$, are dropped. If tokens $[n_k\langle i\rangle - n_a\langle i\rangle]^+$ remain, they are transferred to the token-bucket to be utilized at the next scheduling time, $t + \Delta t$. This process has the effect of absorbing the prediction error of the arrival rate, $n_a\langle i\rangle$. In addition, to adaptively calculate the source and flow rates ($g\langle i\rangle$, $f_{(i,j)}$), the adaptation factor, $\Upsilon\langle i\rangle$, is determined using the minimum optimal token rates of the nodes. By multiplying $\Upsilon\langle i\rangle$ by $g^{(1)}\langle i\rangle^*$ and $f_{(i,j)}^{(1)\,*}$, the respective source and flow rates ($g\langle i\rangle$, $f_{(i,j)}$) are updated.

V. Inter-Node Optimization at the Transport and Network Layers

As mentioned above, when attempting to maximize the sum of the source rates, energy exhaustion may occur in nodes near the sink node. It is therefore desirable to evenly balance the energy consumption over nodes in the network. Toward this end, the sum of the log of the source rates is employed as the objective function rather than using the sum of the source rates. Through this inter-node optimization, we obtain the optimal source and flow rates initially. Using the optimal initial source rate, the optimal initial new token rate is found. The formulation of the optimization problem is conducted under the following conditions.

Flow Conservation: The outgoing rate from a sensor node is the same as the sum of the incoming and self-generated rates. Therefore, the flow conservation during a scheduling period, Δt, can be expressed as

$$\overbrace{\sum_{j\in\mathcal{P}\langle i\rangle\setminus\{i\}} \dot{n}_k^{(0)}\langle j\rangle \cdot D_s\langle j\rangle}^{\text{incoming rate}} + \overbrace{\dot{n}_k^{(0)}\langle i\rangle \cdot D_s\langle i\rangle}^{\text{self-generated rate}} = \overbrace{\sum_{j\in\mathcal{P}\langle i\rangle} \dot{n}_k^{(0)}\langle j\rangle \cdot D_s\langle j\rangle}^{\text{outgoing rate}}, \; i, j \in V, \; (i,j) \in E \tag{7}$$

where V is the set of nodes (or vertices), E is the set of directed links (or edges) in a directed graph that describes a sensor network and $\dot{n}_k^{(0)}\langle i\rangle$ is the initial new token rate of $\dot{n}_k\langle i\rangle$. For flow conservation, the harvesting energy and arrival traffic constraints are not involved in inter-node optimization. Therefore, the incoming, outgoing and self-generated rates are represented via $\dot{n}_k^{(0)}\langle i\rangle$ instead of $n_s\langle i\rangle$. In addition, (7) can be rewritten as

$$\overbrace{\sum_{j\in\mathcal{I}(i)} f_{(j,i)}^{(0)}}^{\text{incoming rate}} + \overbrace{g^{(0)}\langle i\rangle}^{\text{self generated rate}} = \overbrace{\sum_{j\in\mathcal{O}(i)} f_{(i,j)}^{(0)}}^{\text{outgoing rate}}, \tag{8}$$

where $i, j \in V$, $(i, j) \in E$.

Energy Conservation: The energy consumed by a sensor node in scheduling time t should be less than or equal to the allocated energy. Let $E_{act}(\dot{n}_k^{(0)}\langle i\rangle)$ be the allocated energy to support $\dot{n}_k^{(0)}\langle i\rangle$. The incoming and outgoing flow rates at node i are then bound by the allocated energy, $E_{act}(\dot{n}_k^{(0)}\langle i\rangle)$, from (3).

$$E_{act}(\dot{n}_k^{(0)}\langle i\rangle) \ge (P_r\cdot\tau_r + P_d\cdot\tau_d) \cdot \sum_{j\in\mathcal{I}(i)} f_{(j,i)}^{(0)} + (P_c\cdot\tau_c + P_t\cdot\tau_t)\cdot \sum_{j\in\mathcal{O}(i)} f_{(i,j)}^{(0)} + \varepsilon \tag{9}$$

where $\varepsilon = N_{sen}\cdot P_s\cdot\tau_s$. For ease of description, (9) is rewritten as

$$\mathcal{R}\langle i\rangle \ge \sum_{j\in\mathcal{I}(i)} f_{(j,i)}^{(0)} + \gamma \sum_{j\in\mathcal{O}(i)} f_{(i,j)}^{(0)}, \tag{10}$$

where $\mathcal{R}\langle i\rangle = \left(\frac{E_{act}(\dot{n}_k^{(0)}\langle i\rangle)-\varepsilon}{P_r\cdot\tau_r}\right)$, $\gamma = \frac{P_t\cdot\tau_t}{P_r\cdot\tau_r}$.

Based on proportional fairness, flow conservation and energy conservation, the optimization problem of maximizing the source rates can be formulated as

$$\begin{aligned} \max_{\mathbf{F}^{(0)},\mathbf{G}^{(0)}} \quad & \sum_{i\in V}\log(g^{(0)}\langle i\rangle) \\ \text{subject to} \quad & \sum_{j\in\mathcal{I}(i)} f_{(j,i)}^{(0)} + g^{(0)}\langle i\rangle = \sum_{j\in\mathcal{O}(i)} f_{(i,j)}^{(0)}, \\ & \sum_{j\in\mathcal{I}(i)} f_{(j,i)}^{(0)} + \gamma \sum_{j\in\mathcal{O}(i)} f_{(i,j)}^{(0)} \le \mathcal{R}\langle i\rangle, \\ & g^{(0)}\langle i\rangle, f_{(i,j)}^{(0)}, f_{(j,i)}^{(0)} \ge 0 \end{aligned} \tag{11}$$

where $\mathbf{G}^{(0)} = [g^{(0)}\langle 1\rangle, \ldots, g^{(0)}\langle |V|\rangle]$ is a vector expression of the initial source rates, and $\mathbf{F}^{(0)}$ is a matrix expression of the initial flow rates.

$$\mathbf{F}^{(0)} = \begin{bmatrix} f_{(1,1)}^{(0)} & \cdots & f_{(1,|V|)}^{(0)} \\ \vdots & \ddots & \vdots \\ f_{(|V|,1)}^{(0)} & \cdots & f_{(|V|,|V|)}^{(0)} \end{bmatrix}.$$

In (11), to calculate $\mathbf{F}^{(0)*}$ and $\mathbf{G}^{(0)*}$ in order to maximize the sum of the log of the source rates, the control parameters $g^{(0)}\langle i\rangle$, $f_{(i,j)}^{(0)}$ and $f_{(j,i)}^{(0)}$ are utilized for $i, j \in V, (i, j) \in E$. The parameters $\mathcal{R}\langle i\rangle$ and γ and the two sets $\mathcal{I}(i)$ and $\mathcal{O}(i)$ are given for two conservation constraints, (8) and (10).

Unfortunately, the dual problem of (11) is not strictly concave relative to $\mathbf{F}^{(0)}$. To overcome this difficulty, regularized approximation is used as in [24] (see Chapter 6.3 Regularized approximation), whereby a regularization term is added: $\epsilon\sum_{(i,j)\in E}(f_{(i,j)}^{(0)})^2$, for $\epsilon > 0$. This term is a small value and converges to 0.

The regularization method is an effective means to solve the dual problem when a sufficient degree of strict convexity or concavity is lacking. This method facilitates determination

of an optimal solution by adding the norm of a control variable or the square of the norm to the objective function as a regularization term when differentiating the Lagrangian function. This regularization term is added to the objective function (11). The updated objective function becomes strictly concave in terms of the control variable, $f_{(i,j)}^{(0)}$. The Lagrangian function, including the regularization term, is written as

$$\begin{aligned} &L(\mathbf{F}^{(0)}, \mathbf{G}^{(0)}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \\ &= \sum_{i \in V} \log(g^{(0)}\langle i \rangle) - \epsilon \sum_{(i,j) \in E} (f_{(i,j)}^{(0)})^2 \\ &\quad - \sum_{i \in V} \lambda\langle i \rangle \left[\sum_{j \in \mathcal{I}(i)} f_{(j,i)}^{(0)} - \sum_{j \in \mathcal{O}(i)} f_{(i,j)}^{(0)} + g^{(0)}\langle i \rangle \right] \\ &\quad - \sum_{i \in V} \mu\langle i \rangle \left[\sum_{j \in \mathcal{I}(i)} f_{(j,i)}^{(0)} + \gamma \sum_{j \in \mathcal{O}(i)} f_{(i,j)}^{(0)} - \mathcal{R}\langle i \rangle \right], \end{aligned} \tag{12}$$

where $\boldsymbol{\lambda} = [\lambda\langle 1 \rangle, \ldots, \lambda\langle |V| \rangle]$ and $\boldsymbol{\mu} = [\mu\langle 1 \rangle, \ldots, \mu\langle |V| \rangle]$ are vector expressions of Lagrange multipliers. The value of (12) is almost unchanged in the regularization term since $0 < \epsilon \ll 1$.

The dual problem then becomes

$$\min \; D(\boldsymbol{\lambda}, \boldsymbol{\mu}), \quad \text{subject to} \;\; \boldsymbol{\mu} \succeq 0,$$

where $\succeq$ denotes a component-wise inequality, and the objective function can be expressed as

$$D(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \max_{\mathbf{F}^{(0)}, \mathbf{G}^{(0)}} \; L(\mathbf{F}^{(0)}, \mathbf{G}^{(0)}, \boldsymbol{\lambda}, \boldsymbol{\mu}). \tag{13}$$

Since the Lagrangian is a function of two primal variables $(\mathbf{F}^{(0)}, \mathbf{G}^{(0)})$ and two Lagrange multipliers $(\boldsymbol{\lambda}, \boldsymbol{\mu})$, the dual function can be decomposed into three sub-problems, $D_{\text{tra}}(\boldsymbol{\lambda})$, $D_{\text{net}}(\boldsymbol{\lambda}, \boldsymbol{\mu})$ and $D_{\text{mac}}(\boldsymbol{\mu})$:

$$\begin{aligned} D(\boldsymbol{\lambda}, \boldsymbol{\mu}) &= D_{\text{tra}}(\boldsymbol{\lambda}) + D_{\text{net}}(\boldsymbol{\lambda}, \boldsymbol{\mu}) + D_{\text{mac}}(\boldsymbol{\mu}), \\ \text{P1}: D_{\text{tra}}(\boldsymbol{\lambda}) &= \max \left\{ \sum_{i \in V} \log(g^{(0)}\langle i \rangle) - \sum_{i \in V} \lambda\langle i \rangle g^{(0)}\langle i \rangle \right\}, \\ \text{P2}: D_{\text{net}}(\boldsymbol{\lambda}, \boldsymbol{\mu}) &= \max \Bigg\{ -\epsilon \sum_{(i,j) \in E} (f_{(i,j)}^{(0)})^2 \sum_{i \in V} \lambda\langle i \rangle \\ &\qquad \left[\sum_{j \in \mathcal{I}(i)} f_{(j,i)}^{(0)} - \sum_{j \in \mathcal{O}(i)} f_{(i,j)}^{(0)} \right] - \sum_{i \in V} \mu\langle i \rangle \\ &\qquad \left[\sum_{j \in \mathcal{I}(i)} f_{(j,i)}^{(0)} + \gamma \sum_{j \in \mathcal{O}(i)} f_{(i,j)}^{(0)} \right] \Bigg\}, \\ \text{P3}: D_{\text{mac}}(\boldsymbol{\mu}) &= \max \left\{ \sum_{i \in V} \mu\langle i \rangle \mathcal{R}\langle i \rangle \right\}. \end{aligned}$$

From the layering perspective, P1, P2 and P3 depict the functional behavior of the transport, network, and MAC layers in terms of the source rate, $\mathbf{G}^{(0)}$, the link-flow rate, $\mathbf{F}^{(0)}$, and the initial new token rate, $\dot{n}_k^{(0)}\langle i \rangle$, respectively. The sub-problems, P1 and P2, are related to congestion and flow control and are bound by the duration of the duty cycle.

Using this formulation, it is possible to achieve distributed network optimization by finding an optimal solution for the corresponding layer of each node. These sub-problems are mutually coupled, so the optimal solution can be obtained via cross-layer optimization. Since the third sub-problem, $D_{\text{mac}}(\boldsymbol{\mu})$, is studied in Section VI, we here focus on the two sub-problems, $D_{\text{tra}}(\boldsymbol{\lambda})$ and $D_{\text{net}}(\boldsymbol{\lambda}, \boldsymbol{\mu})$.

Using the supergradient method, we perform the following four steps to determine the solution to the dual problem:

- Step 1: *Initialization* - Start with any point, $\lambda_1\langle i \rangle$, $\mu_1\langle i \rangle$, where $\lambda_1\langle i \rangle$ and $\mu_1\langle i \rangle$ are the initial values of $\lambda_n\langle i \rangle$ and $\mu_n\langle i \rangle$ in the n^{th} iteration. Choose an infinite sequence of positive step-size values, $\{a_n\}_{n=1}^{\infty}$ and $\{b_n\}_{n=1}^{\infty}$ for $\lambda\langle i \rangle$ and $\mu\langle i \rangle$. Set $n = 1$.
- Step 2: *Supergradient* - Determine the optimal solutions $(\mathbf{F}^{(0)*}, \mathbf{G}^{(0)*})$, such that

$$\begin{aligned} &\mathbf{F}^{(0)*}, \mathbf{G}^{(0)*} \\ &= \arg \max_{\mathbf{F}^{(0)}, \mathbf{G}^{(0)}} \Bigg\{ \sum_{i \in V} \log(g^{(0)}\langle i \rangle) - \epsilon \sum_{(i,j) \in E} (f_{(i,j)}^{(0)})^2 \\ &\quad - \sum_{i \in V} \lambda_n\langle i \rangle \cdot s_1(\mathbf{F}^{(0)}, \mathbf{G}^{(0)}) - \sum_{i \in V} \mu_n\langle i \rangle \cdot s_2(\mathbf{F}^{(0)}) \Bigg\} \end{aligned}$$

  where $s_1(\mathbf{F}^{(0)}, \mathbf{G}^{(0)}) = \sum_{j \in \mathcal{I}(i)} f_{(j,i)}^{(0)} - \sum_{j \in \mathcal{O}(i)} f_{(i,j)}^{(0)} + g^{(0)}\langle i \rangle$, $s_2(\mathbf{F}^{(0)}) = \sum_{j \in \mathcal{I}(i)} f_{(j,i)}^{(0)} + \gamma \sum_{j \in \mathcal{O}(i)} f_{(i,j)}^{(0)} - \mathcal{R}\langle i \rangle$. Since $\mathbf{G}^{(0)*}$ and $\mathbf{F}^{(0)*}$ are also optimal solutions of $D_{\text{tra}}(\boldsymbol{\lambda})$ and $D_{\text{net}}(\boldsymbol{\lambda}, \boldsymbol{\mu})$, update them via

$$\begin{aligned} g^{(0)}\langle i \rangle^* &= \left[\frac{1}{\lambda_n\langle i \rangle} \right]^+, \\ f_{(i,j)}^{(0)\,*} &= \left[\frac{\lambda_n\langle j \rangle - \lambda_n\langle i \rangle + \gamma \cdot \mu_n\langle i \rangle + \mu_n\langle j \rangle}{-2\epsilon} \right]^+. \end{aligned} \tag{14}$$

  Set $s_1^* := s_1(\mathbf{F}^{(0)*}, \mathbf{G}^{(0)*})$ and $s_2^* := s_2(\mathbf{F}^{(0)*})$. If $\|s_1^*\| \leq \delta^{th}$ and $\|s_2^*\| \leq \delta^{th}$, where δ^{th} is a small stopping threshold, stop the search procedure and obtain $\mathbf{F}^{(0)*}$ and $\mathbf{G}^{(0)*}$.
- Step 3: *Step-size* - The initial step sizes, a_1 and b_1, are random numbers in the range of $[0, U_b]$, where $U_b > 0$ is the upper bound. For the n^{th} iteration, the step-sizes, a_n and b_n, are updated by

$$a_n = \frac{a_{n-1}}{\sqrt{n}}, \quad b_n = \frac{b_{n-1}}{\sqrt{n}}, \quad n \geq 2.$$

- Step 4: *Lagrange Multipliers* - Using the gradient-descent method [24] (see Chapter 9.3 Gradient descent method), the Lagrange multipliers are updated by

$$\begin{aligned} \lambda_{n+1}\langle i \rangle &= \left[\; \lambda_n\langle i \rangle + a_n \frac{s_1^*}{\|s_1^*\|} \; \right], \\ \mu_{n+1}\langle i \rangle &= \left[\; \mu_n\langle i \rangle + b_n \frac{s_2^*}{\|s_2^*\|} \; \right]^+. \end{aligned} \tag{15}$$

  Set $n = n + 1$ and proceed to Step 2.

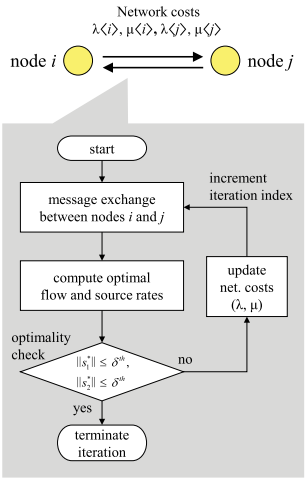


Fig. 6. Flow chart of inter-node optimization.

Finally, using $g^{(0)}\langle i\rangle^*$, the optimal value of $\dot{n}_k^{(0)}\langle i\rangle$ at each node can be obtained as follows:

$$\dot{n}_k^{(0)}\langle i\rangle^* = \frac{g^{(0)}\langle i\rangle^*}{D_s\langle i\rangle}. \tag{16}$$

In Step 2, the supergradient method algorithm is known to guarantee convergence when the objective function is convex/concave. Therefore, using the regularization method guarantees convergence of this iterative procedure, since (13) is strictly concave. Owing to this property, an optimal solution is uniquely found as a global optimum.

Figure 6 is a flow chart that shows how the various network parameters, including the source and flow rates and the routing paths are determined in the BUCKET algorithm. Nodes i and j exchange network costs (Lagrange multipliers, i.e., $\boldsymbol{\lambda}, \boldsymbol{\mu}$) and the optimal flow and source rates are computed at each iteration. Optimality of the computed flow and source rates is determined at each node using the gradient method. The optimality condition is satisfied when the computed source and flow rates fall within a suitably narrow range around the optimal value, i.e., the computed values become near-optimal values. If the optimality condition is passed, the nodes terminate the iteration. If the computed values do not pass the optimality condition, then each node updates its network costs and continues the iteration until the optimality condition is satisfied. A detailed analysis in terms of the number of exchanges for convergence, the energy required for the exchanges, and the associated protocol is presented in Appendix B.

VI. Intra-Node Optimization at the MAC Layer

A. Measurement of Residual Energy

For the inter-node optimization in Section V, the harvesting energy constraint of each node is not taken into account. Thus, intra-node optimization is performed to update the optimal solutions, $g^{(0)}\langle i\rangle^*$, $f_{(i,j)}^{(0)*}$ and $\dot{n}_k^{(0)}\langle i\rangle^*$, according to the harvesting and available energies of each node. Toward this end, an energy-scale factor, $\Gamma\langle i,t\rangle$, at each node, i, is flexibly utilized at scheduling time t. Since each node has a unique availability, the bottleneck relies on the minimum value of $\Gamma\langle i\rangle$ for $\forall i \in V$. Because the nodes near the sink node generally conduct more task requests, the bottleneck node often exists around the sink node. In general, the minimum energy-scale factor of the network, $\Gamma_{min}\langle t\rangle$, is expressed as

$$\Gamma_{min}\langle t\rangle = \min\{\Gamma\langle i,t\rangle\},\ \forall i \in V. \tag{17}$$

Hence, via intra-node optimization, we find the optimal minimum energy-scale factor, $\Gamma^*_{min}\langle t\rangle$, and incorporate the harvesting energy into allocated energy $E_{act}(\Gamma^*_{min}\langle t\rangle \cdot \dot{n}_k^{(0)}\langle i\rangle^*)$ by employing the $\Gamma^*_{min}\langle t\rangle$. It is assumed that the energy source is limited to solar power using Heliomote, that the energy-harvesting system obtains energy from the solar panel only, and that this energy can be consumed with full efficiency or recharged with an efficiency of η. In addition, when the energy is consumed directly from the solar panel, no leakage occurs, i.e., $\eta = 1$. On the other hand, if the energy is recharged into the battery, the efficiency is in the range, $0 < \eta < 1$. Thus, higher energy utilization can be achieved through direct energy consumption.

For simplicity, daytime and nighttime are respectively defined as $T_D = \{t|P_h\langle i,t\rangle > 0\}$ and $T_N = \{t|P_h\langle i,t\rangle = 0\}$. During daytime, both the recharging and energy expenditures can occur simultaneously, while only energy expenditure occurs during nighttime. Then, let N_{sch} be the number of scheduling periods during a day, let $N_{sch} \cdot \Delta t$ be 24 h and let $t_{N_{sch}}$ be $t_0 + (N_{sch} - 1) \cdot \Delta t$. Based on $E_{act}(n_s\langle i\rangle)$ and previous work [1], [2], the residual energy at time $t_{N_{sch}}$ at node i for both time intervals, T_D and T_N, can then be modeled as

$$\begin{aligned} E_{res}\langle i, t_{sch}\rangle = E_{res}\langle i,t_0\rangle &+ \eta \sum_{t\in T_D} [\mathcal{A} - \mathcal{B}\cdot\Gamma\langle i,t\rangle]^+ \\ &- \sum_{t\in T_D} [\mathcal{B}\cdot\Gamma\langle i,t\rangle - \mathcal{A}]^+ \\ &+ \eta \sum_{t\in T_D} [(\mathcal{C} - \mathcal{D}\cdot\Gamma\langle i,t\rangle)\cdot\mathcal{G}]^+ \\ &- \sum_{t\in T_N} (\mathcal{E} - \mathcal{F}\cdot\Gamma\langle i,t\rangle), \end{aligned} \tag{18}$$

where $E_{res}\langle i,t_0\rangle$ is the residual energy at the initial time t_0, $\mathcal{A} = N_{sen}\cdot\tau_s\cdot(\tilde{P}_h\langle i,t\rangle - P_s)$, $\mathcal{B} = E_{act}(n_s\langle i,t\rangle) - N_{sen}\cdot P_s\cdot\tau_s$, $\mathcal{C} = \Delta t - N_{sen}\cdot\tau_s$, $\mathcal{D} = t_{act}\langle i,t\rangle - N_{sen}\cdot\tau_s$, $\mathcal{E} = P_{sp}\cdot(\Delta t - N_{sen}\cdot\tau_s) + N_{sen}\cdot P_s\cdot\tau_s$, $\mathcal{F} = \mathcal{D}\cdot P_{sp} + \mathcal{B}$ and $\mathcal{G} = \tilde{P}_h\langle i,t\rangle - P_{sp}$, $t \in [t_0, t_{sch}]$.

In (18), the first term is the initial energy at the battery, the second term is the energy recharged with efficiency η after expending $E_{act}(n_s\langle i\rangle)$ during the active state, the third term is the energy consumption of $E_{act}(n_s\langle i\rangle)$ beyond the instantaneous harvesting power, the fourth term is the energy recharged with efficiency η in the sleep state and the fifth term is the energy allocated at T_N and consumed during the sleep state. In addition, by setting $E_{res}\langle i,t_{sch}\rangle - E_{res}\langle i,t_0\rangle = \Delta E_{res}(\Gamma\langle i,t\rangle)$ and by assuming $P_h\langle i,t\rangle \geq P_{sp}$, for $\forall t$, the variation in residual energy for $[t_0, t_{sch}]$ is derived as

follows:

$$\begin{aligned}
&\Delta E_{res}(\Gamma\langle i,t\rangle)\\
&\quad=\eta\sum_{t\in T_D}[\mathcal{A}-\mathcal{B}\cdot\Gamma\langle i,t\rangle]^+-\sum_{t\in T_D}[\mathcal{B}\cdot\Gamma\langle i,t\rangle-\mathcal{A}]^+\\
&\qquad+\eta\sum_{t\in T_D}\{(\mathcal{C}-\mathcal{D}\cdot\Gamma\langle i,t\rangle)\cdot\mathcal{G}\}-\sum_{t\in T_N}(\mathcal{E}-\mathcal{F}\cdot\Gamma\langle i,t\rangle)\\
&\quad=\begin{cases}\text{if } \mathcal{A}-\mathcal{B}\cdot\Gamma\langle i,t\rangle^*\geq 0,\quad \forall t\in T_D,\\ \eta\cdot\sum\limits_{t\in T_D}(\mathcal{A}+\mathcal{C}\cdot\mathcal{G})-\eta\sum\limits_{t\in T_D}(\mathcal{B}-\mathcal{D}\cdot\mathcal{G})\cdot\Gamma\langle i,t\rangle,\\ \text{if } \mathcal{A}-\mathcal{B}\cdot\Gamma\langle i,t\rangle^*<0,\quad \forall t\in T_D,\\ \sum\limits_{t\in T_D}(\mathcal{A}+\mathcal{C}\cdot\mathcal{G})-\sum\limits_{t\in T_D}(\mathcal{B}-\eta\cdot\mathcal{D}\cdot\mathcal{G})\cdot\Gamma\langle i,t\rangle,\\ \text{if } \forall t\in T_N,\quad \sum\limits_{t\in T_N}\mathcal{E}-\sum\limits_{t\in T_D}\mathcal{F}\cdot\Gamma\langle i,t\rangle.\end{cases}
\end{aligned}\tag{19}$$

B. Intra-Node Optimization Calculation

The residual energy, $E_{res}\langle i\rangle$, in each node and the variation of the residual energy, $\Delta E_{res}\langle i\rangle$, over the scheduling interval can be derived for the optimal solution. Since $E_{res}\langle i\rangle$ and $\Delta E_{res}\langle i\rangle$ are functions of $n_s\langle i,t\rangle$, by (18) and (19), energy scheduling is performed for each scheduling interval by controlling the minimum energy-scale factor, $\Gamma_{min}\langle t\rangle$. The new token rate, $\dot{n}_k\langle i,t\rangle$, is then determined based on the variation in the residual energy by solving the convex optimization,

$$\begin{aligned}\dot{n}_k\langle i,t\rangle&=\Gamma_{min}\langle t\rangle\cdot\dot{n}_k^{(0)}\langle i,t\rangle^*\\&=\min\{\Gamma\langle i,t\rangle\}\cdot\dot{n}_k^{(0)}\langle i,t\rangle^*,\quad\forall i\in V,\end{aligned}$$

where $\dot{n}_k^{(0)}\langle i,t\rangle$ is obtained via (16).

To obtain the optimal new token rate, $\dot{n}\langle i,t\rangle^*$, it is necessary to determine the optimal energy-scale factor, $\Gamma\langle i,t\rangle^*$. Toward this end, intra-node optimization is performed, and the utility function in terms of $\Gamma\langle i,t\rangle$ is defined as

$$\begin{aligned}U\langle i,t\rangle&=\log\left(1+e_a\langle i,t\rangle\ \dot{n}_k\langle i,t\rangle\right)\\&=\log\left(1+e_a\langle i,t\rangle\ \Gamma\langle i,t\rangle\ \dot{n}_k^{(0)}\langle i,t\rangle^*\right)\end{aligned}\tag{20}$$

where $e_a\langle i,t\rangle$ is a weight factor reflecting the arriving traffic of node i at scheduling time t. Eq. (20) implies that $\Gamma\langle i,t\rangle$ should be determined based on the value of $e_a\langle i,t\rangle$. Parameter $e_a\langle i,t\rangle$ is assumed to be obtained in advance by estimating the arrival rate. In addition, $\dot{n}_k^{(0)}\langle i,t\rangle^*$ is also obtained in advance via (16). Thus, the utility function $U\langle i,t\rangle$ can be expressed as a function of $\Gamma\langle i,t\rangle$. Using $\Gamma\langle i,t\rangle$, set $\Delta E_{res}(\Gamma\langle i,t\rangle)=M_h+N_{sch}\cdot N_{sen}\cdot E_{cal}\langle i\rangle$ for $\forall t$, where M_h is the margin on the estimation error, and $E_{cal}\langle i\rangle$ is the energy consumption required for the exchange of network prices $(\boldsymbol{\lambda},\boldsymbol{\mu})$ in (14) for inter-node optimization. The margin, M_h, is bound by the maximum error between the harvested and allocated energies. The detailed derivation of $E_{cal}\langle i\rangle$ is described in Appendix B. Therefore, intra-node optimization can be performed by controlling the variable cost, $\Gamma\langle i,t\rangle$, in order to achieve maximum profit under the given supply $\tilde{P}_h\langle i,t\rangle$ with margin M_h at each node. The optimal value of $\Gamma\langle i,t\rangle$ is obtained via the following optimization:

$$\begin{aligned}&\max_{\boldsymbol{\Gamma}}\sum_{t\in[t_0,t_{sch}]}U\langle i,t\rangle\\&\text{subject to }\Delta E_{res}(\Gamma\langle i,t\rangle)=M_h+N_{sch}\cdot N_{sen}\cdot E_{cal}\langle i\rangle,\\&\qquad\forall\ \Gamma\langle i,t\rangle\geq 0,\end{aligned}\tag{21}$$

where $\boldsymbol{\Gamma}=[\Gamma\langle i,t_0\rangle,...,\Gamma\langle i,t_{sch}\rangle]$ is a vector expression of energy-scale factors, and t is in the range of $[t_0,t_{sch}]$, so that the optimal utility is obtained during the time duration.

The Lagrangian function of (21) becomes

$$\begin{aligned}L(\boldsymbol{\Gamma},\boldsymbol{\nu},\boldsymbol{\omega})=&-\sum_{t\in[t_0,t_{sch}]}U\langle i,t\rangle\\&+\sum_{t\in[t_0,t_{sch}]}\nu\langle t\rangle\{\Delta E_{res}(\Gamma\langle i,t\rangle)-M_h\}\\&-\sum_{t\in[t_0,t_{sch}]}\omega\langle t\rangle\Gamma\langle i,t\rangle,\end{aligned}$$

where $\nu\langle t\rangle$ and $\omega\langle t\rangle$ are Lagrange multipliers of the first and second constraints, respectively, in (21).

Problem (21) is convex with respect to $\Gamma\langle i,t\rangle$; while the first constraint is an affine function of $\Gamma\langle i,t\rangle$, the second constraint is a half-space with respect to $\Gamma\langle i,t\rangle$, and the objective function is the sum of concave functions. Hence, the optimization problem in (21) can be solved using the Karush-Kuhn-Tucker (KKT) conditions, as follows:

$$\begin{aligned}&1.\ \Delta E_{res}(\Gamma\langle i,t\rangle)-M_h-N_{sch}\cdot N_{sen}\cdot E_{cal}\langle i\rangle=0,\\&\qquad\Gamma\langle i,t\rangle\geq 0,\ t\in[t_0,t_{sch}]\\&\qquad 2.\ \omega\langle t\rangle\geq 0,\ t\in[t_0,t_{sch}]\\&\quad 3.\ \omega\langle t\rangle\cdot\Gamma\langle i,t\rangle=0,\ t\in[t_0,t_{sch}]\\&4.\ \frac{\partial L(\boldsymbol{\Gamma},\boldsymbol{\nu},\boldsymbol{\omega})}{\partial\Gamma\langle i,t\rangle}=-\frac{\partial U\langle i,t\rangle}{\partial\Gamma\langle i,t\rangle}+\rho\langle t\rangle\cdot\nu\langle t\rangle-\omega\langle t\rangle=0.\end{aligned}$$

$\rho\langle t\rangle$ is a coefficient of $\Gamma\langle i,t\rangle$ in (19) and can be written as

$$\rho\langle t\rangle=\begin{cases}-\eta\cdot\mathcal{B}-\eta\cdot\mathcal{D}\cdot\mathcal{G}, & \text{if }\ \mathcal{A}-\mathcal{B}\cdot\Gamma\langle i,t\rangle^*\geq 0,\ \forall t\in T_D,\\ -\mathcal{B}-\eta\cdot\mathcal{D}\cdot\mathcal{G}, & \text{if }\ \mathcal{A}-\mathcal{B}\cdot\Gamma\langle i,t\rangle^*<0,\ \forall t\in T_D,\\ \mathcal{F}, & \text{if }\ \forall t\in T_N,\end{cases}$$

where $\mathcal{A},\mathcal{B},\mathcal{D},\mathcal{F}$ and $\mathcal{G}$ are derived in (18). Therefore, by manipulating the KKT conditions, the optimal solution of (21) becomes

$$\Gamma\langle i,t\rangle^*=\left[\frac{1}{\rho\langle t\rangle\cdot\nu\langle t\rangle^*}-\frac{1}{e_a\langle i,t\rangle\cdot\dot{n}_k^{(0)}\langle i\rangle^*}\right]^+.\tag{22}$$

From (19), the energy allocated to node i during $[t_0,t_{sch}]$ can be derived using $\rho\langle t\rangle$ and (22) as

$$\begin{aligned}\sum_{t\in[t_0,t_{sch}]}\rho\langle t\rangle\Gamma\langle i,t\rangle^*&=\sum_{t\in[t_0,t_{sch}]}\left[\frac{1}{\nu\langle t\rangle^*}-\frac{\rho\langle t\rangle}{e_a\langle i,t\rangle\cdot\dot{n}_k^{(0)}\langle i\rangle^*}\right]^+\\&=\sum_{t\in[t_0,t_{sch}]}P_h\langle i,t\rangle\\&\quad-M_h-N_{sch}\cdot N_{sen}\cdot E_{cal}\langle i\rangle.\end{aligned}\tag{23}$$

In (23), the allocated energy is affected by $\frac{1}{\nu\langle t\rangle^*}$ and the total harvesting energy, so it is important to determine $\frac{1}{\nu\langle t\rangle^*}$ based

on the total harvesting energy for one day. Equation (23) indicates that the difference $\left[\frac{1}{\nu\langle t\rangle^*} - \frac{\rho\langle t\rangle}{e_a\langle i,t\rangle \cdot \dot{n}_k^{(0)}\langle i\rangle^*}\right]^+$ between the water-level, $\frac{1}{\nu\langle t\rangle^*}$, and the inverse of the required energy, $e_a\langle i,t\rangle \cdot \dot{n}_k^{(0)}\langle i\rangle^*/\rho\langle t\rangle$, should be allocated to each scheduling time; scheduling times with greater $e_a\langle i,t\rangle \cdot \dot{n}_k^{(0)}\langle t\rangle^*/\rho\langle t\rangle$ occupy more token rates. Since this sum is approximately equal to the sum of $P_h\langle i,t\rangle$ in the active and sleep states, the problem is the same as determining the allocation of $(\sum_{t\in[t_0,t_{sch}]} P_h\langle i,t\rangle - M_h - N_{sch} \cdot N_{sen} \cdot E_{cal}\langle i\rangle)$ at each scheduling time. Thus, the water-level, $\frac{1}{\nu\langle t\rangle^*}$, should be determined to satisfy (23) over the total time duration $[t_0, t_{sch}]$ for one day. Notice that the weight vector, $e_a\langle i,t\rangle$, allows for preferential energy allocation to scheduling times with a large $e_a\langle i,t\rangle \cdot \dot{n}_k^{(0)}\langle i\rangle^*/\rho\langle t\rangle$.

If the amount of harvesting power is sufficient to support the total arrival rate, then $\dot{n}_k\langle i\rangle^*$ can be assigned more than $n_a\langle i,t\rangle$. In such a case, the optimal solution may be meaningless; however, when the amount of harvesting power is similar to or less than that of the required power, the proposed scheme outperforms conventional schemes in terms of data throughput, as demonstrated in the simulations.

C. Derivation of Adapted Source and Flow Rates

At time $t \in [t_0, t_{sch}]$, $\Gamma\langle i,t\rangle^*$ is determined for node i through the intra-node optimization. After obtaining $\Gamma\langle t\rangle^*$ among $\Gamma\langle i,t\rangle^*, i \in V$ in (17), the optimal scaled source and flow rates are obtained by

$$g^{(1)}\langle i,t\rangle^* = g^{(0)}\langle i,t\rangle^* \cdot \Gamma\langle t\rangle^*, \quad f_{(i,j)}^{(1)\,*}\langle t\rangle = f_{(i,j)}^{(0)\,*}\langle t\rangle \cdot \Gamma\langle t\rangle^*.$$

Using $\Gamma\langle t\rangle^*$, an optimal new token rate is obtained via

$$\dot{n}_k\langle i,t\rangle^* = \hat{n}_k^{(0)}\langle i,t\rangle^* \cdot \Gamma\langle t\rangle^*. \tag{24}$$

From (24), the transmission rate is calculated using

$$n_t\langle i,t\rangle = \min\{n_a\langle i,t\rangle, n_k\langle i,t\rangle^*\}. \tag{25}$$

To adaptively adjust the source and flow rates to the arrival rate, an adaptation factor, $\Upsilon\langle i,t\rangle$, is defined using (25) for the token-bucket algorithm:

$$\Upsilon\langle i,t\rangle = \frac{\min\{n_a\langle i,t\rangle,\ n_k\langle \tilde{i},t\rangle^*\}}{n_k\langle \tilde{i},t\rangle^*} = \frac{n_t\langle i,t\rangle}{n_k\langle \tilde{i},t\rangle^*} \tag{26}$$

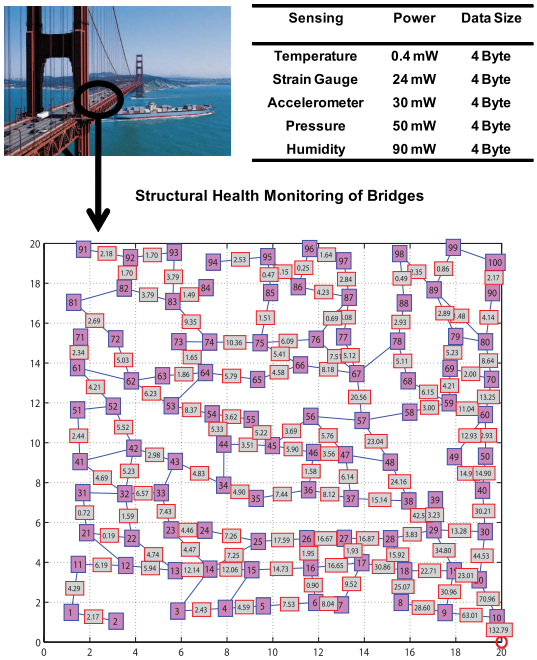
where $\tilde{i} = \arg\min_i\{\Gamma\langle i,t\rangle\},\ \forall i \in V$. Using (26), the adapted source and flow rates can be respectively written as

$$g\langle i,t\rangle = g^{(1)}\langle i,t\rangle^* \cdot \Upsilon\langle i,t\rangle, \quad f_{(i,j)}\langle i,t\rangle = f_{(i,j)}^{(1)\,*}\langle t\rangle \cdot \Upsilon\langle i,t\rangle.$$

VII. Simulation Results

A. Simulation Environment

To evaluate the performance of the BUCKET algorithm, the structural health monitoring (SHM) system of a bridge is used as a representative and useful example of a wireless sensor network [25]. The purpose of the SHM system is to obtain quantitative data on the safety of a bridge structure [26]. By measuring structural behavior, it is possible to confirm design assumptions, to provide real-time feed-back during construction, and to perform a controlled lifetime extension of the bridge. We focus on the behavior of data acquisition via sensor networking.



Sensing	Power	Data Size
Temperature	0.4 mW	4 Byte
Strain Gauge	24 mW	4 Byte
Accelerometer	30 mW	4 Byte
Pressure	50 mW	4 Byte
Humidity	90 mW	4 Byte

Fig. 7. Deployment of an energy-harvesting sensor network on a bridge in order to monitor structural health.

The performance of the BUCKET algorithm is evaluated in a C program that employs statistics derived from real data measured in Seoul [27]. We designed the simulation environment to be as authentic as possible by using parameters drawn from actual hardware [7] and by modeling realistic solar power acquisition patterns [23]. Figure 7 depicts the simulation environment. The specific simulation environments are arranged as follows:

1) 100 harvesting sensor nodes are quasi-uniformly distributed over a $20m \times 20m$ space.
2) One of the 100 nodes is a sink node. The other nodes forward information to the sink node via the multi-path routing method described in Section V.
3) Five types of sensors are installed at each node in order to perform structural health monitoring: a thermometer, a strain gauge, an accelerometer, and pressure and humidity seasons, using a data size of 4 bytes and sensing powers of $0.4mW$, $24mW$, $30mW$, $50mW$ and $90mW$, respectively.
4) The power values for the transmit, receive and sleep operations are determined from the parameters of the Heliomote sensor device (e.g. $P_r = 38mW$, $P_t = 35mW$ and $P_{sp} = 30\mu W$).
5) Each node operates using the equal harvesting pattern of [23] with the following harvesting-related parameters: the maximum power recharged from the solar panel per unit area is $18mW/cm^2$, the area of the solar panel is $225cm^2$, the local latitude is 38.5^o, and the month is assumed to be May.
6) The arrival rate that is used follows the temporal distribution shown in Figure 8. This arrival rate was

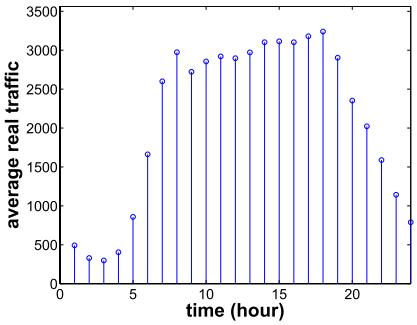


Fig. 8. Average arrival rate used for structural health monitoring.

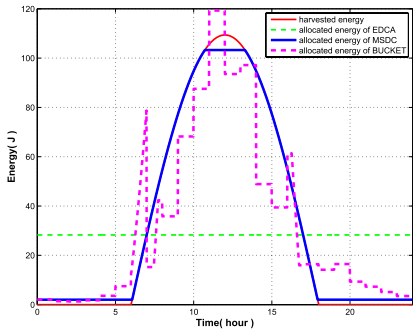


Fig. 9. Comparison of energy allocation at node 10 for BUCKET, MSDC and EDCA.

estimated from the traffic volume measured over bridges in Seoul [27]. The traffic volume statistics are modeled using the PMF and CDF in (4).

B. Performance Evaluation

To demonstrate the efficiency of the BUCKET algorithm, we programmed two existing basic algorithms for a performance comparison: Maximization of the sum of the duty cycle (MSDC) [1], [2] and equal duty cycle allocation (EDCA), as shown in Figure 9.

Figure 9 shows the energy allocations using the MSDC scheme, the BUCKET algorithm and the EDCA scheme with the aforementioned parameters. In the MSDC scheme, a greater amount of energy is allocated during the daytime T_D than at nighttime T_N in order to attain the maximum sum of duty cycles given an energy constraint. This does not, however, guarantee that the amount of serviced $\mathcal{T}_{com}$ is maximized. In particular, relay traffic is not accounted for in the optimization; the allocated energies of all nodes are equal, as shown in Figure 9; thus, it may be inefficient to apply such a scheme to multi-hop transmission, where severe traffic flow can occur at the sink node. As shown in Figure 9, the EDCA scheme can achieve fair energy allocation over all scheduling times, but inefficiency occurs during the daytime, T_D, since the recharging efficiency over time is not included. Figures 9 and 10(a) show the energy allocation curve and the energy evolution at each node using BUCKET, based on the simulation environments of Section VII-A. Via inter- and intra-node optimization, node 10 near the sink node requires a larger token rate for data relay, while the other nodes require smaller

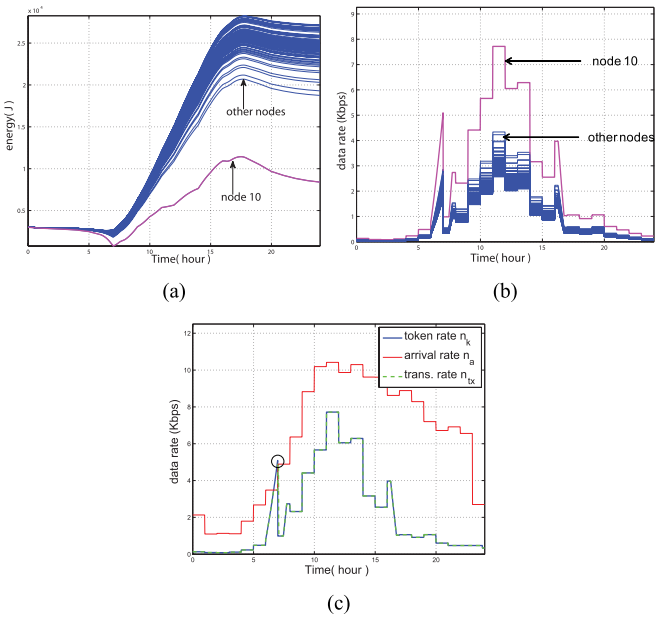


Fig. 10. A performance analysis of the BUCKET algorithm. (a) Energy evolution at each node. (b) New token rate, $\dot{n}_k$ at each node. (c) Token rate, n_k, arrival rate, n_a and transmission rate, n_t.

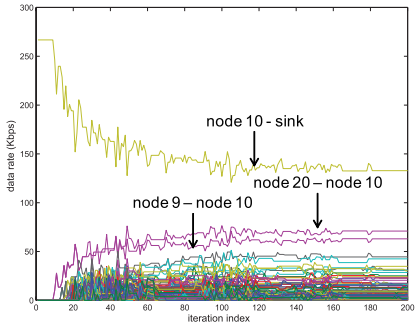


Fig. 11. The flow rate convergence at each link ($\epsilon = 0.1$).

token rates, as shown in Figure 10(b). In other words, the optimal token rates far from the sink (nodes 91 and 92, for example) are controlled by feedback from nodes near the sink (as in node 10), thereby avoiding traffic congestion at nodes near the sink. Figure 10(c) shows adaptation of the token rate into the bucket. The circle in Figure 10(c) is the transferred portion of the token rate, n_k, as explained in Figure 4, which has the effect of overcoming the estimation error.

Figure 11 shows the convergence of flow rates as a function of the iteration index in inter-node optimization, when the traffic is delivered from the sensor nodes. Since node 10 is a neighboring node of the sink node, the source traffic from all of the sensor nodes is delivered to the sink node via node 10. Since node 10 is a bottleneck in terms of utility maximization, packet-drops may occur there due to a scarcity of resources (time or power) during a multi-hop transmission; therefore, when the reserved energy at node 10 is insufficient for delivering all of the data, the other nodes can reduce their source rates via feedback from node 10. In Figure 11, it is apparent that the flow rate increases with the order of nodes

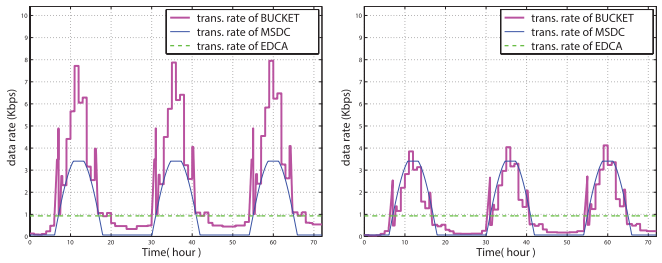


Fig. 12. Comparison of the three schemes in terms of the transmission rate. (a) Transmission rates of the three schemes at node 10. (b) Transmission rates of the three schemes at node 91.

due to data accumulation. Since the flow rate is bound by rate $\mathcal{R}\langle i\rangle$ in (10), the source and flow rates converge.

Suppose that the other schemes, MSDC and EDCA, are assumed to utilize shortest single-path routing with equal source rates over the node index. Figure 12 shows the transmission rates of the three schemes for this case at nodes 10 and 91, which are the most distant and nearest nodes to the sink node, respectively. While the MSDC and EDCA schemes have the same transmission rates over the node index, the BUCKET algorithm exhibits superior performance at node 10, owing to the proportional fairness imposed by inter-node optimization. Though MSDC at node 91 yields a better transmission rate than BUCKET during the daytime, T_D (Figure 12(b)), it exhibits a transmission rate of nearly zero at night, T_N, although the arrival rate is quite large at $18 \sim 22$h, as shown in Figure 10(c). Conversely, the BUCKET algorithm demonstrates balanced performance over time by accounting for both the recharging efficiency and arrival rate. Compared to the average transmission rate, the BUCKET algorithm delivers performance enhancements of approximately 12% and 15% over MSDC and EDCA, respectively.

To summarize, since the BUCKET algorithm considers the limited harvesting energy, the arrival rate and the amount of relay traffic assuming optimal networking, it generally achieves a higher transmission rate than the other two schemes.

VIII. Conclusion

There is considerable interest in the design of renewable solar energy-harvesting systems capable of replacing conventional chemical batteries. This important issue is timely for sensor networks. Prior research has focused on maximizing data throughput and network lifetime extension using conventional chemical batteries, with less emphasis on the efficient use of solar energy for sensor networks. Since there are inefficiencies in solar panel recharge systems, it is difficult to optimize the overall system based only on empirical power acquisition patterns. Here, we considered the acquisition pattern of the solar energy-harvesting system and constructed a framework for maximizing the utilization of solar energy-harvesting in general sensor networks. Toward this goal, cross-layer optimization was proposed as a best strategy via inter-node cooperation among sensor nodes, a unique feature relative to prior approaches. The optimization problem was formulated into inter- and intra-node optimization sub-problems, resulting in a holistic solution that demonstrated excellent performance. For practical performance evaluation, the energy-harvesting sensor network and SHM system of a bridge were evaluated by modeling the actual hardware-related parameters and the solar-power acquisition system.

Appendix A
Derivation of Bit Rate

The signal-to-interference plus noise ratio (SINR) at receiver j from transmitter i under the assumption of the static channel is

$$\gamma_{(i,j)} = \frac{P_t \cdot l_{(i,j)}}{N_0 + I\langle j\rangle}$$

where $l_{(i,j)}$ is the path-loss between nodes i and j and is calculated as $l_{(i,j)} = d_{(i,j)}^{-2} \cdot \zeta_{(i,j)}$, where $d_{(i,j)}$ is the distance between nodes i and j, and $\zeta_{(i,j)}$ is the random variable representing Rayleigh fading with $E[\zeta_{(i,j)}^2] = 1$. N_0 is the noise power, $I\langle j\rangle$ is the interference at node j and $I\langle j\rangle = \frac{1}{RF}\sum_{k\in n\setminus m(j)\setminus\{i,j\}} P_t \cdot l_{(k,j)}$. In addition, RF is the spatial reuse factor for the CSMA/CA channel access (in this grid topology, RF is approximately nine), $m(j)$ are neighboring contenders to node j for CSMA/CA channel access, and $m(j) = \{l | P_t \cdot l_{(l,j)} > \nu_{sen}, l \neq i, l \neq j\}$, where ν_{sen} is the receiver sensitivity. Therefore, the bit rate becomes

$$R_{bw}(\gamma_{(i,j)}) = BW \cdot \frac{1}{RF} \cdot \log_2(\gamma_{(i,j)}),$$

where BW is the bandwidth.

Appendix B
Practical Issues of the BUCKET Algorithm

Here, we address practical issues of the BUCKET algorithm in terms of distributed computation.

A. Cost Required for Distributed Computation

Distributed computation affects the energy utilization for message exchanges, as in network prices ($\boldsymbol{\lambda}$, $\boldsymbol{\mu}$), for example. In a distributed computation, each participating node consumes energy at every iteration, so the nodal energy consumption increases linearly as the iterations proceed. The energy consumption required for the message exchanges of (14) in inter-node optimization then becomes

$$E_{cal}\langle i\rangle = N_{iter} \cdot \Bigg\{ P_t \sum_{j\in\mathcal{I}(i)\cup\mathcal{O}(i)} \tau_t \cdot |\lambda\langle j\rangle + \mu\langle j\rangle| + P_r \sum_{j\in\mathcal{I}(i)\cup\mathcal{O}(i)} \tau_r \cdot |\lambda\langle j\rangle + \mu\langle j\rangle| \Bigg\}, \quad (27)$$

where N_{iter} is the number of iterations performed for the distributed optimization, $|\lambda\langle j\rangle + \mu\langle j\rangle|$ and is the amount of bits required for representing $\lambda\langle j\rangle + \mu\langle j\rangle$. Here, assuming equal $|\lambda\langle j\rangle + \mu\langle j\rangle|$ for all nodes, (27) can be simplified using $E_{cal}\langle i\rangle = N_{iter} \cdot |\mathcal{I}(i)\cup\mathcal{O}(i)| \cdot |\lambda\langle i\rangle + \mu\langle i\rangle| \cdot \{P_t \cdot \tau_t + P_r \cdot \tau_r\}$, and the computational complexity, $O(E_{cal}\langle i\rangle)$, can be obtained via

$$O(E_{cal}\langle i\rangle) \cong N_{iter} \cdot |\mathcal{I}(i) \cup \mathcal{O}(i)| \cdot |\lambda\langle i\rangle + \mu\langle i\rangle|.$$

N_{iter} is a function of the step size, the number of sensor nodes, the number of connected links among nodes, and the generated $\mathcal{T}_{com}$. Figure 11 depicts the flow convergence according to N_{iter} where 100 nodes are deployed. At least 120 iterations are required for convergence of the iterative procedure. As the number of sensor nodes increases, the value of N_{iter} also increases. As the number of nodes is increased, the number of links between nodes also becomes greater on average, which leads to an increment in the number of messages required for convergence. However, message exchange is performed only for links between neighboring nodes, which are not significant in number. Most importantly, the distributed computation distributes the energy consumption of a cluster head or representative node over all nodes participating in the routing, which extends the network lifetime. In this respect, the BUCKET algorithm is well-suited for sensor networks operating over very large nodes.

REFERENCES

[1] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embedded Comput. Syst.*, vol. 6, no. 4, p. 32, Sep. 2007.

[2] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, "Adaptive duty cycling for energy harvesting systems," in *Proc. Int. Symp. Low Power Electron. Design*, Oct. 2006, pp. 180–185.

[3] C. Moser, L. Thiele, D. Brunelli, and L. Benini, "Adaptive power management in energy harvesting systems," in *Proc. Conf. Design, Automat. Test Eur. Conf. Exhibit.*, Apr. 2007, pp. 773–778.

[4] H. K. Le, D. Henriksson, and T. Abdelzaher, "A practical multi-channel media access control protocol for wireless sensor networks," in *Proc. 7th Int. Conf. Inform. Process. Sensor Netw.*, Apr. 2008, pp. 70–81.

[5] V. Raghunathan and P. H. Chou, "Design and power management of energy harvesting embedded systems," in *Proc. Int. Symp. Low Power Electron. Design*, 2006, pp. 369–374.

[6] A. Hande, T. Polk, W. Walker, and D. Bhatia, "Indoor solar energy harvesting for sensor network router nodes," *Microprocessors Microsyst.*, vol. 31, no. 6, pp. 420–432, Sep. 2007.

[7] K. Lin *et al.*, "Heliomote: Enabling long-lived sensor networks through solar energy harvesting," in *Proc. 3rd Int. Conf. Embedded Netw. Sensor Syst.*, Nov. 2005, p. 309.

[8] D. Noh, D. Lee, and H. Shin, "QoS-aware geographic routing for solar-powered wireless sensor networks," *IEICE Trans. Commun.*, vol. 90, no. 12, pp. 3373–3382, Dec. 2007.

[9] D. Niyato, E. Hossain, and A. Fallahi, "Sleep and wakeup strategies in solar-powered wireless sensor/mesh networks: Performance analysis and optimization," *IEEE Trans. Mobile Comput.*, vol. 6, no. 2, pp. 221–236, Feb. 2007.

[10] S. Cui, A. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," *IEEE Trans. Wireless Commun.*, vol. 4, no. 5, pp. 2349–2360, Sep. 2005.

[11] R. Banner and A. Orda, "Bottleneck routing games in communication networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 6, pp. 1173–1179, Aug. 2007.

[12] N. Abughalieh, Y. A. Le Borgne, K. Steenhaut, and A. Nowe, "Lifetime optimization for sensor networks with correlated data gathering," in *Proc. IEEE 8th Int. Symp. Modeling Optim. Mobile, Ad Hoc Wireless Netw. (WiOpt)*, May 2010, pp. 266–272.

[13] M. Chiang, S. H. Low, R. A. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[14] K. W. Fan, Z. Zheng, and P. Sinha, "Steady and fair rate allocation for rechargeable sensors in perpetual sensor networks," in *Proc. 6th ACM Conf. Embedded Netw. Sensor Syst.*, 2008, pp. 239–252.

[15] R. Liu, P. Sinha, and C. E. Koksal, "Joint energy management and resource allocation in rechargeable sensor networks," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[16] D. Noh and T. Abdelzaher, "Efficient flow-control algorithm cooperating with energy allocation scheme for solar-powered WSNs," *Wireless Commun. Mobile Comput.*, vol. 12, no. 5, pp. 379–392, Apr. 2012.

[17] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Control of wireless networks with rechargeable batteries [transactions papers]," *IEEE Trans. Wireless Commun.*, vol. 9, no. 2, pp. 581–593, Feb. 2010.

[18] L. Huang and M. J. Neely, "Utility optimal scheduling in energy-harvesting networks," in *Proc. 12th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2011.

[19] S. Sarkar, M. H. R. Khouzani, and K. Kar, "Optimal routing and scheduling in multihop wireless renewable energy networks," *IEEE Trans. Autom. Control*, vol. 58, no. 7, pp. 1792–1798, Jul. 2013.

[20] F. Kelly, "Charging and rate control for elastic traffic," *Eur. Trans. Telecommun.*, vol. 8, no. 1, pp. 33–37, 1997.

[21] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[22] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[23] S. Lee, B. Kwon, and S. Lee, "Joint energy management system of electric supply and demand in houses and buildings," *IEEE Trans. Power Syst.*, vol. 29, no. 6, pp. 2804–2812, Nov. 2014.

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

[25] J. Lynch, "An overview of wireless structural health monitoring for civil structures," *Philosoph. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 365, no. 1851, pp. 345–372, Feb. 2007.

[26] D. Inaudi, "Overview of 40 bridge structural health monitoring projects," in *Proc. Int. Bridge Conf. (IBC)*, Jun. 2010.

Sungjin Lee was born in Korea in 1980. He received the B.S. degree from Soongsil University, Seoul, Korea, in 2005, and the M.S. and Ph.D. degrees from Yonsei University, Seoul, in 2007 and 2011, respectively, all in electrical engineering. He is currently with Samsung Electronics Company, Ltd., Suwon, Korea, where he is working on 5G communication development. His research interests are in the area of wireless communication networks, in particular, sensor network, small-cell network, and wireless resource management issues.

Beom Kwon was born in Korea in 1989. He received the B.S. degree in electrical engineering from Soongsil University, Seoul, Korea, in 2012. He is currently pursuing the M.S. and Ph.D. degrees in multidimensional insight laboratory at Yonsei University, Seoul. His research interests are in the areas of small cell networks and cooperative communications.

Sanghoon Lee (SM'12) received the B.S. degree in electrical engineering from Yonsei University, Seoul, Korea, in 1989, and the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 1991. From 1991 to 1996, he was with Korea Telecom, Seongnam, Korea. He received the Ph.D. degree in electrical engineering from the University of Texas at Austin, Austin, TX, USA, in 2000. From 1999 to 2002, he was with Lucent Technologies, where he worked on 3G wireless and multimedia networks. In 2003, he joined the faculty of the Department of Electrical and Electronics Engineering at Yonsei University, where he is currently a Full Professor. He has been an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING (2010–2014), an Editor of the *Journal of Communications and Networks* (since 2009), and the Chair of the IEEE P3333.1 Quality Assessment Working Group (since 2011). He served as a Technical Committee Member of the IEEE IVMSP (since 2014), the Technical Program Co-Chair of the International Conference on Information Networking in 2014, the 2012 Global 3D Forum in 2013, the General Chair of the 2013 IEEE IVMSP Workshop, and a Guest Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING in 2013. He was a recipient of the 2012 Special Service Award from the IEEE Broadcast Technology Society and the 2013 Special Service Award from the IEEE Signal Processing Society. His research interests include image/video quality assessments, medical image processing, cloud computing, wireless multimedia communications, and wireless networks.

Alan Conrad Bovik (F'96) is currently the Curry/Cullen Trust Endowed Chair Professor with the University of Texas at Austin, Austin, TX, USA, where he is also the Director of the Laboratory for Image and Video Engineering. He is a faculty member with the Department of Electrical and Computer Engineering and the Center for Perceptual Systems, Institute for Neuroscience, University of Texas at Austin. His research interests include image and video processing, computational vision, and visual perception. He has authored over 650 technical articles in these areas, and holds two U.S. patents. His several books include the recent companion volumes entitled *The Essential Guides to Image and Video Processing* (Academic Press, 2009).

He has received a number of major awards from the IEEE Signal Processing Society, including the Best Paper Award (2009), the Education Award (2007), the Technical Achievement Award (2005), and the Meritorious Service Award (1998). He was also a recipient of the Honorary Member Award from the Society for Imaging Science and Technology (2013), the Society of Photo-Optical and Instrumentation Engineers (SPIE) Technology Achievement Award (2012), the IS&T/SPIE Imaging Scientist of the Year (2011), the Hocott Award for Distinguished Engineering Research at the University of Texas at Austin, the Distinguished Alumni Award from the University of Illinois at Urbana-Champaign (2008), the IEEE Third Millennium Medal (2000), and two Journal Paper Awards from the International Pattern Recognition Society (1988 and 1993). He is a Fellow of the Optical Society of America, SPIE, and the American Institute of Medical and Biomedical Engineering. He has been involved in numerous professional society activities, including the Board of Governors of the IEEE Signal Processing Society (1996–1998), the Co-Founder and Editor-in-Chief of the IEEE TRANSACTIONS ON IMAGE PROCESSING (1996–2002), an Editorial Board Member of the PROCEEDINGS OF THE IEEE (1998–2004), a Series Editor of *Image, Video, and Multimedia Processing* (Morgan and Claypool Publishing Company, 2003–present), and the Founding General Chairman of the First IEEE International Conference on Image Processing, held in Austin (1994).

Dr. Bovik is a registered Professional Engineer in the state of Texas, and a Frequent Consultant to legal, industrial, and academic institutions.