This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Elevator-Assisted Sensor Data Collection for Structural Health Monitoring

Tao Zhang, Dan Wang, *Member, IEEE,* Jiannong Cao, *Senior Member, IEEE,* Yi Qing Ni, Li-Jun Chen, and Daoxu Chen, *Member, IEEE*

Abstract—Sensor networks nowadays are widely used for structural health monitoring; for example, the sensor monitoring system deployed on the Guangzhou New TV Tower, China. While wired systems still dominate, it is commonly believed that wireless sensors will play a key role in the near future. One key difficulty for such systems is the data transmission from the sensor nodes to the base station. Given the long span of the civil structures, neither a strategy of long-range one-hop data transmission nor short-range hop-by-hop communication is cost-efficient.

In this paper, we propose a novel scheme of using the elevators to assist data collection. A base station is attached to an elevator. A representative node on each floor collects and transmits the data to the base station using short range communication when the elevator stops at or passes by this floor. As such, communication distance can be minimized. To validate the feasibility of the idea, we first conduct an experiment in an elevator of the Guangzhou New TV Tower. We observe steady transmission when elevator is in movement. To maximize the gain, we formulate the problem as an optimization problem where the data traffic should be transmitted on time and the lifetime of the sensors should be maximized. We show that if we know the movement pattern of the elevator in advance, this problem can be solved optimally. We then study the online version of the problem and show that no online algorithm has a constant competitive ratio against the offline algorithm. We show that knowledge of the future elevator movement will intrinsically improve the data collection performance. We discuss how the information could be collected and develop online algorithms based on different level of knowledge of the elevator movement patterns. Theoretically, given that the links capacity assumptions we made, we can prove that our online algorithm can guarantee data delivery on time. In practice, we may set a buffer zone to minimize the possible data delivery violation. A comprehensive set of simulations and MicaZ testbed experiments have demonstrated that our algorithm substantially outperforms conventional multi-hop routing and naive waiting for elevator scheme. The performance of our online algorithm is close to the optimal offline solution.

Index Terms—Wireless sensor networks, data collection, mobile sink.

1 Introduction

Sensor networks nowadays have been widely used for structural health monitoring (SHM) applications. For example, the Ting Kau Bridge in Hong Kong [1] is equipped with a large number of accelerometers, thermometers, and strain sensors to monitor its working conditions. Another recent project that we are working on is the monitoring system for the current inbuilt Guangzhou New TV Tower (GNTVT) [2]. From these projects, there is a gradual yet clear transition from the wired sensor systems to partially wireless sensor systems.

In SHM applications, the sensors are deployed on critical locations that are of civil importance and periodically sample the data. A commonly adopted data collection strategy is to assign a representative node on each floor which collects all the data from the sensors on this floor. These representative nodes then transmit the data back to the base station located at the foot of the structure. Conventionally, the data transmission is carried out by wires. For a life-long monitoring system, a wire-dominated system is still a reasonable choice. For a short term (weeks or months) evaluation of the structure, a wired system introduces huge deployment cost. Another major headache is the in-construction monitoring; the wire can be very easily damaged during the hammers and drills of structure construction. Obviously, wireless sensor systems can play a more important role for these applications.

There are two possible data collection strategies for the wireless communication system, namely short-range hop-by-hop routing and long-range single-hop transmission. Long range single hop transmission (partially adopted by the GNTVT project) is costly both in communication devices and suffers greatly if the energy supply is limited or difficult to obtain. Hop-by-hop routing will put high burden to the sensors that are close to the base station as these sensors need to relay large amount of data (note that in civil applications, the data are not aggregated in the intermediate nodes). A careful design

- *T. Zhang, L. Chen, and D. Chen are with the State Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, China. This work was done while Tao Zhang was a research assistant in Department of Computing, The Hong Kong Polytechnic University.*
  Email: zt@dislab.nju.edu.cn, {chenlj, cdx}@nju.edu.cn
- *D. Wang and J. Cao are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong.*
  Email: {csdwang, csjcao}@comp.polyu.edu.hk
- *Y. Q. Ni is with the Department of Civil and Structural Engineering, The Hong Kong Polytechnic University, Hong Kong.*
  Email: ceyqni@polyu.edu.hk

Digital Object Indentifier 10.1109/TMC.2011.191 1536-1233/11/$26.00 © 2011 IEEE

of a mixed scheme combining these two can lead to a more efficient system; however, the intrinsic difficulty remains, that is, the larger the structure, the longer the distance from the sensors to the base station.

In this paper, we propose a novel scheme which uses the elevators to assist data collection. We install a base station on an elevator. When the elevator stops at or passes by a certain floor, the representative sensors on this floor will forward the data to the base station using short range communications. This scheme provides a scalable solution as the communication range and the load of the sensors closer to the base station will not increase when the number of floors increases. Definitely, a prior is that the building should be elevator equipped. This is true for high-rise buildings (e.g., in China, the regulation requests all seven-floor buildings or higher to be elevator-equipped); and for smaller buildings, the burden of both long-range one-hop communication and hop-by-hop routing are more likely to be acceptable.

We have conducted preliminary experiments in an elevator of the Guangzhou New TV Tower, and observed that steady data transmission is possible while an elevator is in movement. More importantly, when an elevator stops at certain floors, there is a long period of time for communication and a significantly larger amount of data can be transmitted. Such throughput, after careful scheduling, can also be used to assist transmission of data from the sensors on neighboring floors.

To fully explore the benefit of this idea, many challenges need to be addressed. First, we need to maximize the amount of data transmitted. While we can improve the data transmission rate by using more expensive hardware/antenna, this can also be achieved by careful routing schedule of data transmission among sensors. Second, the data should be collected on time. Though SHM applications are delay tolerant to some extent, there is a limit. If the data are entirely delay insensitive (can be collected in the infinite remote future), there is no need to collect data at all. Third, for energy constrained sensors, we need to balance the load and maximize system lifetime. Fourth, the purpose of the elevator is to carry passengers. We cannot control the frequency of the elevators movement nor the floors that the elevators stop at. As such, the routing schedule must be adaptive to the online elevator movement.

In this paper, we provide a systematic study to the aforementioned problems. We formulate an elevator-assisted sensor data collection problem which we need to maximize the lifetime of the sensors, as well as to guarantee that the periodically generated data can be transmitted on time. We show that if the movement of the elevator is known in advance, this problem can be reduced to some maximum flow problems and solved optimally. We show that the online version is significantly more difficult and no online algorithm can achieve a constant competitive ratio as opposed to an optimal offline algorithm.

We illustrate that some movement sequences lead to the intrinsic gap between the online algorithm and offline algorithm. We show that the knowledge of an absolute elevator movement sequence is not necessary. This is of great practical importance as it makes certain level of prediction a reasonable assumption. We then develop an online algorithm with different level of knowledge of elevator movement. We evaluate our scheme by a comprehensive set of simulations and MicaZ testbed experiments. We compare our scheme with multi-hop transmission and naive waiting for elevator schemes. Significant performance improvement is observed.

The remaining part of the paper proceeds as follows. The background, our preliminary experimental validation in an elevator of GNTVT, and problem formulation are presented in Section 2. In Section 3, we develop optimal algorithms for three different offline problems. Section 4 is devoted to the difficulty and the solutions for the online elevator-assisted algorithms. We evaluate our algorithms with a comprehensive set of the simulation in Section 5 and MicaZ testbed experiments in Section 6. The related work is discussed in Section 7. Section 8 concludes the paper.

2 Background, Experiments and The Problem

As the first work to use elevators to assist data collection for SHM applications, we first clarify the necessary details of the application; and outline our problem. We study the offline scenario where the elevator movement is known in advance and the online scenario in the next sections.

Without loss of generality, we assume that there is a structure with N floors and there is one representative sensor on each floor. This representative sensor can collect the data from all the sensors on this floor. To have our study more focused, in the rest of the paper, we restrict ourselves on these representative sensors. We call the representative sensors the *source sensors* or just the *sensors*. We assume that the sensors on adjacent floors can communicate with each other; the elevator hoistway could be used to provide better link quality; note that this is also a necessary assumption for hop-by-hop routing in any wireless systems. In this paper, we study the scenario that the sensors only communicate with the sensors on the two neighboring floors. Our results, however, can be easily extended to the scenario where the sensors can communicate with the sensors far away.

We install a base station d on an elevator; we call it the *sink* or the *mobile sink*.[1] To make communication possible, the sink can be installed on top of the elevator, or a slim antenna can be used so that it can stretch outside the elevator compartment through ventilation ports. Note that there can be multiple elevators and multiple sinks. The throughput between the source sensors and the mobile

1. If there is no ambiguity, we sometimes also use the elevator to refer the base station sensor attached to it.

(a) Sensor mote (b) Sink in elevator

Fig. 1. Experimental environment in GNTVT

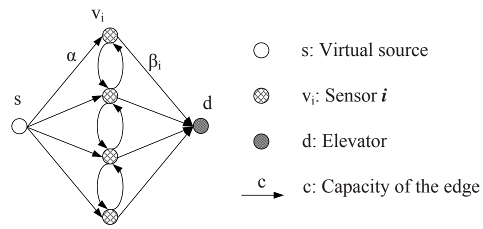


Fig. 2. Graph $G(V, E)$ and the network flow model.

sink depends on the total length of their encountering time. The most important factor is whether the elevator stops at one floor or simply passes by.

We further assume that the throughput between two source sensors on neighboring floors is large enough (i.e., there is no throughput constraint). This is because the rate of data generation is around $O(1)$Mbits per hour (the heaviest load comes from the accelerometers which has a rate of 100Hz). The elevators movement frequency is in the order of tens of minutes and there is a reasonable amount of time for the adjacent sensors to exchange data of such loads.

To verify that the communication is realistic, we conducted a preliminary experiment in GNTVT. A sensor mote was placed on one floor of the tower and a sink was placed in the No. 2 elevator of GNTVT (see Fig.1 (a) and (b) respectively). Our sensor motes were equipped with 7dBi antenna to enhance the signal strength in the poor construction conditions[2]. The speed of the elevator was around $1.5m/s$. We observed the transmission could easily reach $55Kbps$ or above. Note that we used the MAC layer self-equipped with the sensor. We also tested in our experiments and observed that steady communication was possible between adjacent floors.

In SHM applications, the data are periodically sampled. We call every period a *round*. Let the data generated at each sensor be α bits per round. Generally speaking, the SHM applications are delay tolerant. Note however, that each round may be a few hours and the data could become less useful if the delivery delay is very long. In the extreme case, if the data are entirely delay insensitive, there is no need to collect data as we only need the data in the infinite remote future. Consequently, we assume that the data have to be delivered to the base station no later than $\mathcal{L}$ rounds after they are generated.

We define the system lifetime to be the lifetime of the first depleted sensor. This conforms to the requirement of civil applications, where missing data can greatly affect the accuracy of evaluation. Let $\mathcal{E}$ denote the energy reserve of each sensor. We use e_t and e_r to denote the energy consumption for transmitting and receiving one bit of data. In this paper, we focus on the data traffic and ignore the energy consumed for the control traffic. As the mobile sinks are attached to the elevators, we assume that they have no energy constraint.

In this paper, we focus on traffic scheduling and consider MAC layer as a black box. This follows a modular design and favors system complexity reduction. We do understand that a cross layer optimization may further improve the performance. We did some initial study as shown in our poster [3]. In our experiments in Section 6, we chose a low duty-cycle X-MAC [4].

With the above model, the objectives of the elevator assisted data collection are: 1) to guarantee that the data generated can be collected on time, and 2) to maximize the system lifetime.

2. The GNTVT is in construction and we can put the base station in the elevator compartment as the door is hollow. In real deployment, as explained, the base station can be installed on top of the elevator outside the compartment or the antenna can be extended out from the ventilation port.

3 Elevator-Assisted Data Collection: The Offline Scenario

In this section, we study three different offline versions of the problem and develop associated algorithms. Here, *offline* scenario means the entire elevator movements are completely known in advance. We discuss *online* scenario where part of the elevator movements could be predicted in Section 4.

3.1 Maximizing the Throughput in a Single Round (MT)

We first consider the case where the data should be reported every round, i.e., $\mathcal{L} = 1$. We emphasize that each round may represent a few hours and $\mathcal{L} = 1$ reflects the requirement of some real-time monitoring systems. In this subsection, we focus on the first objective. We maximize system throughput so that we can examine whether the system can guarantee all the data generated to be collected. This also paves the way for our understanding of the overall problem.

Let v_i represent the sensor on the ith floor where $i \in [1 \ldots N]$. Let $S = \{v_1, v_2, ..., v_N\}$. We construct a directed virtual graph $G(V, E)$, where $V = \{s, d\} \cup S$. Here s is a virtual data generator and d is the virtual mobile sink. E is composed of three parts: 1) $E_s = \{(s, v_i) | v_i \in S\}$; 2) $E_d = \{(v_i, d) | v_i \in S\}$; and 3) $E_v = \{(v_i, v_j) | v_i, v_j \in S \wedge v_i \text{ and } v_j \text{ are on adjacent floors}\}$. The capacity $c(u, w)$ of link (u, w) is

$$c(u,w) = \begin{cases} \alpha, & (u,w) \in E_s, \\ \beta_i, & (u,w) \in E_d, \\ \infty, & \text{otherwise} \end{cases}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

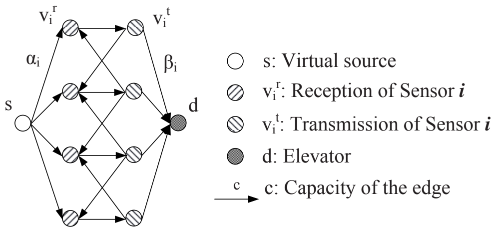


Fig. 3. Graph $G'(V', E')$ and the network flow model.

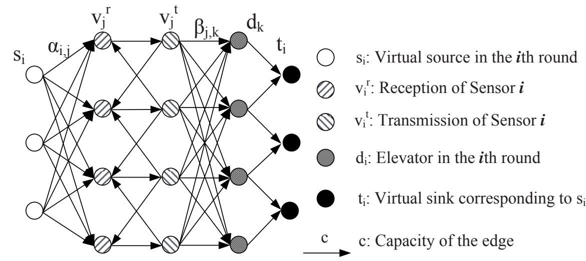


Fig. 4. Graph $G''(V'', E'')$ and the multi-commodity flow model.

Here, β_i is the aggregate throughput between sensor i and the mobile sinks during this round. This is the total throughput capacity of all the mobile sinks pass by and stop at floor i and $\beta_i = 0$ if no mobile sink is encountered during this round. As discussed, since the link capacity between adjacent floors is not bottleneck, in our model we assign infinite link capacity for these links. We also abuse the notation a little and use d to denote the virtual aggregate mobile sinks of all elevators during this round. An illustrative example of graph G is shown in Fig.2.

The problem can be transforms to a max-flow problem where s generates α bits of data for each source sensor v_i and all the data need to be pushed to d. Consequently,

Observation 1: Optimal solution exists for the problem of maximizing throughput for a single round.

If the total capacity between s and S is greater than the total capacity between S and d, there is no solution. Otherwise, we can use standard augmenting path algorithm [5] and solve the max-flow problem optimally. We call this MT algorithm in the rest of the paper. In what follows, we only study the case where the total capacity between S and d is greater than that of s and S.

3.2 Maximizing the Minimum Residual Energy in a Single Round (MMRE)

We next incorporate the second objective of the problem, to maximize system lifetime. In a single round, this reduces to maximizing the minimum residual energy of the sensors. Similarly, we build a graph $G'(V', E')$, by splitting the nodes v_i of G into two nodes v_i^r and v_i^t representing sensor receiving and sensor transmission. Formally, let $S^r = \{v_i^r | 1 \leqslant i \leqslant N\}$, $S^t = \{v_i^t | 1 \leqslant i \leqslant N\}$, we have $V' = \{s, d\} \cup S^r \cup S^t$. E' is composed of four parts: 1) $E_s = \{(s, v_i^r) | v_i^r \in S^r)\}$, 2) $E_e = \{(v_i^r, v_i^t) | 1 \leqslant i \leqslant N\}$, 3) $E_v = \{(v_i^t, v_j^r) | \text{sensor } i \text{ and } j \text{ are on adjacent floors}\}$, and 4) $E_d = \{(v_i^t, d) | v_i^t \in S^t\}$. The link capacity of G' is:

$$c'(u, w) = \begin{cases} \alpha_i, & (u, w) \in E_s, \\ \beta_i, & (u, w) \in E_d, \\ \dfrac{\mathcal{E}_i + \alpha_i e_r}{e_t + e_r}, & (u, w) \in E_e, \\ \infty, & \text{otherwise} \end{cases}$$

Here α_i denotes the amount of data to be transmitted by sensor i. $\mathcal{E}_i$ denotes the residual energy of sensor i. The total throughput capacity of the links in E_e can be calculated as the total residual energy of sensor i, divided by the energy requirement of receiving and transmission of one bit. Here we need to compensate an additional $\alpha_i e_r$ for receiving the α_i bits from the virtual generator s. An illustrative example is shown in Fig.3.

Observation 2: Optimal solution exists for the problem of maximizing the minimum residual energy in a single round; where the data generated from all source sensors are guaranteed to be collected.

The problem can be transformed into a decision problem such that given $\mathcal{E}_r$, by allowing each sensor to have the energy of $\mathcal{E}_i - \mathcal{E}_r$, whether the data flow generated by s can be successfully pushed to d. This decision problem can also be answered by max-flow augment path algorithm. Thus, the maximum of all $\mathcal{E}_r$ is the residual energy. In the rest of the paper, we call this the MMRE algorithm.

Note that unlike in Section 3.1, we on purposely choose to solve the problem with a non-uniform α_i and $\mathcal{E}_i$. The MMRE will serve as a building block for our online algorithms; as after a few rounds, the data to be transmitted and the residual energy on each source will become different.

3.3 The General Case (EA Offline)

Finally, we consider the general case where $\mathcal{L} > 1$. We transform it into a multi-commodity flow problem as follows. Again, we construct a directed graph $G''(V'', E'')$, where $V'' = \widetilde{S} \cup S^r \cup S^t \cup D \cup T$. Let the total rounds be R (the total number of commodity). Let $\widetilde{S} = \{s_1, s_2, \ldots, s_R\}$ be a set of virtual data generators where s_i represents the data generator in the ith round. Similarly, we have S^r and S^t representing the sensor receiving and sensor transmission on each floor. Let set $D = \{d_1, d_2, \ldots, d_{R+\mathcal{L}-1}\}$ be a set of virtual aggregate mobile sinks where d_i denote the aggregated mobile sink for the ith round. Let $T = \{t_1, t_2, \ldots, t_R\}$ be a set of virtual destination for the data generated by $\widetilde{S}$.

E'' is composed of five parts: 1) $E_s = \{(s_i, v_j^r) | s_i \in \widetilde{S} \wedge v_j^r \in S^r\}$ connecting the virtual data generators to each sensor; 2) $E_v = \{(v_i^t, v_j^r) | \text{sensor } i \text{ and } j \text{ are on adjacent floors}\}$ denotes the transmission between adjacent sensors; 3) $E_e = \{(v_i^r, v_i^t) | 1 \leqslant i \leqslant N\}$ denotes the throughput passing sensor i; 4) $E_d = \{(v_i^t, d_j) | v_i^t \in S^t \wedge d_j \in D\}$, denotes the transmission between sensor i and the virtual aggregated mobile sink in the jth round; and 5) $E_t = \{(d_i, t_j) | j \leqslant i < j + \mathcal{L}\}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Let α_{ij} be the data flow generated in the ith round by sensor j.[3] Let β_{jk} be the total throughput between sensor j and the mobile sink during the kth round. The capacity of edges is:

$$c''(u,w) = \begin{cases} \alpha_{ij}, & (u,w) \in E_s, \\ \beta_{jk}, & (u,w) \in E_d, \\ \dfrac{\mathcal{E} + \alpha e_r R}{e_t + e_r}, & (u,w) \in E_e, \\ \infty, & \text{otherwise} \end{cases}$$

This is a multi-commodity flow problem where the flows $f_i = \sum_j \alpha_{ij}$ need to be delivered from a specific s_i to a specific t_i. In other word, for a graph $G''(V'', E'')$, we solve the decision problem of whether the total flow αR for each source sensor can be delivered. Note that our case is a linear programming version; and a feasible solution is easy to find. To maximize system lifetime, we need to maximize the number of source-destination pairs, i.e., R. A binary search can be applied until a feasible solution cannot be found.

Observation 3: Optimal solution exists for the offline elevator-assisted data collection problem.

In the rest of the paper, we call this the EAF algorithm.

Discussion: In our analysis, we did not consider the storage limitation of the sensors. The amount of data in SHM application is moderate. In practice, we implement the system by allowing the data to be queued in neighboring nodes; making the storage a minor problem. The storage can also be seen as a constraint in the problem modeling; and we leave this to our future work.

4 Elevator-Assisted Data Collection: The Online Scenario

4.1 The Difficulty

Though we have successfully solved the offline problem, in the following theorem, we see that developing efficient online algorithm is much more difficult. The intuition behind the theorem is that the online algorithm performs poorly when there are unbalanced elevator movements followed by infrequent elevator movements and then balanced elevator movements. An *unbalanced movement* is that the elevator only stops at a few floors. An *infrequent movement* is that the elevator rarely moves, i.e., stops at certain floors for long periods of time. We formalize this intuition as follows.

Theorem 1: No online algorithm can have a transmission schedule that achieves a constant competitive ratio to the offline algorithm in term of the optimal system lifetime.

Proof: We have N sensors $(v_1, v_2, ..., v_N)$ and one elevator. Assume each sensor generates one packet per round. Also assume that the lifetime of the packets is one round ($\mathcal{L} = 1$); that is the packets have to be transmitted to the ground floor within the same round.

3. Note that in offline scenario, the data generated in each round are the same. We use α_{ij} to unify the notations with the rest of the paper.

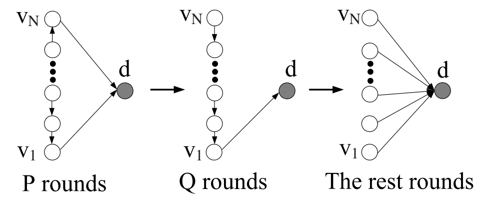


Fig. 5. Three phases of the elevator movement.

Let the throughput between a source sensor and the elevator passes by the floor be zero; the throughput between a source sensor and the elevator stops at the floor be ∞; and the throughput between the source sensors be ∞. We assume one unit of energy is consumed for either packet transmission or receiving. Assume the total energy reserve at each sensor node be $\mathcal{E}$ units and $\mathcal{E} = \theta N$ where $\theta > 6$.

The elevator movement is divided into three different phases. The first phase is composed of P rounds where $2 < P < \frac{\theta}{2}$; the second phase is composed of Q rounds where $Q = \lceil \frac{\mathcal{E}-(N-1)P}{N-1+N} \rceil$; and the third phase is composed of the rest rounds until the energy $\mathcal{E}$ is used up. In the first phase, the elevator will stop at both v_1 and v_N in each round. In the second phase, the elevator will stop at either v_1 or v_N. In the third phase, the elevator will stop at each floor in each round. An illustration of these three phases is shown in Fig.5.

With any online algorithm, in the first phase, either v_1 or v_N will receive at least $\frac{(N-2)P}{2}$ packets and transmit $\frac{(N-2)P}{2} + P$ packets. In total, $(N-1)P$ units of energy will be consumed in these P rounds. Without loss of generality, we assume v_1 spends such amount of energy.

In the second phase, we assume the elevator will stop at v_1.[4] As v_1 needs to undertake all the traffic, the total energy consumed in this round is at least $Q \times ((N-1)+N)$. The total energy consumed in the first phase and the second phase is $(N-1)P + \lceil \frac{\mathcal{E}-(N-1)P}{N-1+N} \rceil \times (N-1+N) \geq \mathcal{E}$.

Clearly, the energy will be exhausted after the first two phases in the online algorithm and the system lifetime is (in terms of rounds; constrained by v_1)

$$L_{on} \leqslant P + \frac{\mathcal{E} - (N-1)P}{(N-1)+N} \leqslant \frac{\mathcal{E} + NP}{2N-1} = \frac{(\theta + P)N}{2N-1}$$

For an offline algorithm, as we can predict the entire movement of elevator, the schedule can be that we use v_N to undertake all packets for the first phase and use v_1 to undertake all packets for the second phase. As such, in the first $P + Q$ rounds, the total energy consumed by v_1 and v_N should be $\mathcal{E}_1 = P + (2N-1)Q$ and $\mathcal{E}_N = (2N-3)P + Q$. In the third phase, both v_1 and v_N spend one unit of energy for transmission its own packet until the battery drains out. The system lifetime with optimal schedule is (in terms of rounds; constrained by either v_1 or v_N)

4. Note that we do not assume that the elevator will choose v_1 or v_N based on the decision of the online algorithm in the first phase. Our argument is that a constant competitive ratio cannot be achieved.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

$$\begin{aligned} L_{opt} &\geqslant min\{P+Q+(\mathcal{E}-\mathcal{E}_1), P+Q+(\mathcal{E}-\mathcal{E}_N)\} \\ &\geqslant min\{(\theta-2P)N+4P, (P-2)N-P+1\} \end{aligned}$$

When N is large enough, we have

$$\lim_{N\to\infty} \frac{L_{on}}{L_{opt}} = \frac{O(1)}{O(N)} = 0$$

This completes the proof. □

4.2 The Online Algorithms

From previous observations and the theorem, we see that with knowledge of future movement information, system can efficiently figure out the optimal solution. As a sharp contrast, without future information, system performance can become arbitrarily bad. Intrinsically, there can be some very bad situation (unbalanced elevator movements followed by infrequent elevator movements and balanced elevator movements). An optimal offline schedule could only be generated if the elevator movement in the entire data collection is known. Note that the entire collection period would last for hundreds of rounds and the data might have a deadline of more than one round. As such, the more knowledge of future movement information, the higher the chance to adjust the schedule.

As can be seen in our offline algorithms, we do not need to predict exactly the movement sequence and the exact time the elevator will stops at/passes by a certain floor. To effectively apply these algorithms, we only need to know the aggregated throughput β_i within a period of time; which is much easier to approximate. This observation is of great practical importance; as it makes the prediction of the elevator movement to certain extent a reasonable assumption.

In reality, the movement of the elevators may even be partially available in some scenario; for example, when passengers input their destination floors on the control pad, the elevator system can immediately obtain this information. Some elevators in sightseeing towers may even have fixed schedules.

In this paper, however, we do not study how to learn the movement of the elevators. We confine our focus to the question of the intrinsic benefit of some information of the future movement; and to what extent we can optimize system performance given such information.

In our model, we assume the link quality is ideal so as to focus the study on the traffic scheduling problem. In our testbed experiments, we handle this problem by reserving extra throughput for error tolerance. This simple reservation actually showed quite satisfactory results. Clearly, there can be optimization to further squeeze better performance out of this extra throughput.

Let the length of each round be T_R; the lifetime of data be $T_{\mathcal{L}} = \mathcal{L} \times T_R$; the current time be $current\_time$; and the sensors can predict the elevator movement schedule for a length of $\mathcal{T}$ ($\mathcal{T} = 0$ if no schedule is known). The prediction is beyond the scope of the paper, which can be based on history traces. In some scenarios, the elevator movement is even fixed. For example the elevator of Space Needle Tower in Seattle has a schedule of 10 minutes in daytime. For the data routing within $\mathcal{T}$, we cannot directly apply the offline algorithms. Within $\mathcal{T}$, there is a mixture of data flows with different deadlines, and some will not reach their deadlines in this period. Let K be the total number of data flows in $\mathcal{T}$. Let the remaining time for flow i before reaching its deadline be T_i^e, $1 \leq i \leq K$.

In this section, we will first develop an algorithm given that at any point of time, we know the elevator movement information for the future period of $\mathcal{T}$. We will further develop an algorithm with dynamic adjustment given the movement of the elevator changes. This is useful when the prediction within $\mathcal{T}$ is not entirely accurate. We will prove that given that the links capacity assumptions we made, our algorithm guarantees that all the data flows can be delivered on time.

4.2.1 Online Schedule Given Prediction of $\mathcal{T}$

Recall that T_R represents the length of a round; there are two different cases, $\mathcal{T} > T_R$ and $\mathcal{T} \leq T_R$. For $\mathcal{T} > T_R$, the system will generate an *online schedule* (or, in short, a *schedule*) at the beginning of every round. Each schedule has a length of $\mathcal{T}$. For $\mathcal{T} \leq T_R$, a round can be divided into multiple full $\mathcal{T}$ periods and a small *leftover* period at the end of T_R. The system will generate an online schedule at the beginning of every $\mathcal{T}$, and the beginning of the small leftover period. Each schedule has a length of $\mathcal{T}$ (except for the schedule in the leftover period, which has a length the same as the leftover period). All the schedule will only apply to the data flows that have been already generated at the time. The pseudo-code of the algorithm can be found in Algorithm 1.

Algorithm 1 Elevator Assisted Online Algorithm (EAO)

```
current_time ← 0.
while TRUE do
  if 𝒯 ⩽ T_R then
    if (current_time mod T_R) + 𝒯 > T_R then
      t = T_R mod 𝒯
    else
      t = 𝒯
    end if
    online-schedule(current_time, current_time + t)
    current_time ← current_time + t
  else
    t = 𝒯
    online-schedule(current_time, current_time + t)
    subtract system throughput allocated to this schedule
    current_time ← current_time + T_R
  end if
end while
```

In each schedule the data flow will be divided into three different priorities. First, the data with deadline within this schedule will be delivered. Second, the remaining data will be delivered with joint optimization of their deadline and the energy consumption. Third, if there is still throughput capacity that the source sensors

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

can push the data to the mobile sink when the elevator passes by; data flows with the earliest deadline will be delivered. The pseudo-code of the schedule can be found in Algorithm 2.

Algorithm 2 online-schedule($current\_time$, $current\_time$+t)

Generate a transmission schedule for a period of t within the prediction period, i.e., $t \leq \mathcal{T}$.

```
Step 1:
Deliver the flow with deadline in t by MMRE()
Step 2:
current_round ← ⌊current_time / T_R⌋
for i = current_round − L + 1 to current_round do
    F_i ← residual data generated in the ith round.
    γ_i ← = F_i − (T_i^e − t)/T_L × αN
    γ_i ← = max(γ_i, 0)
end for
Construct G°
Deliver the data flows by MMRE_ext().
Step 3:
Source sensor j transmits data directly to the mobile sink
    of the data flow with the earliest deadline.
```

In the first step, since the schedule is generated no longer than a round, at most one data flow will reach its deadline. Thus, MMRE() is used to deliver these data.

In the second step, there are data flows with the deadline beyond $\mathcal{T}$. Two decisions are needed: first, what amount of data should be delivered and second, how to route these data.

Let the amount of data generated in the ith round that have not been delivered be $\mathcal{F}_i$. The amount of data that need to be transmitted in this schedule is $\gamma_i = \mathcal{F}_i - \frac{T_i^e - t}{T_{\mathcal{L}}} \times \alpha N$. Notice that αN is the total data flow generated in every round (including the ith round). The data flow generated in the ith round would expire $T_i^e - t$ after the end of this schedule period and the remaining proportion of the lifetime is $\frac{T_i^e - t}{T_{\mathcal{L}}}$. Intuitively, this is a balance with the amount of residual data and the deadline of the data making sure all data can be transmitted to mobile sink on time.

For the routing of these data flows (i.e., γ_i), we maximize the minimum residual energy of the source sensors. This problem is a little different from the MMRE problem as the data flows have different deadlines. Consequently, we develop an extended MMRE_ext() by modification of graph $G'(V', E')$ to $G^\circ(V^\circ, E^\circ)$ where an additional $\overline{s}$ is introduced to distribute γ_i. The development of MMRE_ext() is similar to the techniques used in Section 3 and a formal description is shown in Appendix.

We prove in the next theorem that all data can be transmitted to mobile sink on time if the system throughput capacity $\mathcal{B} \geq \frac{\alpha N}{T_R}$; this is the basic requirement for any feasible schedule.

Theorem 2: If the throughput capacity from the source sensors to the mobile sink $\mathcal{B} \geq \frac{\alpha N}{T_R}$, using Algorithm 1 all the data can be transmitted to the sink on time.

Proof: This is a case by case proof. To smooth the exposition, the proof is delayed to the Appendix. □

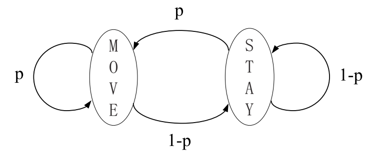


Fig. 6. Elevator movement model used in our simulation.

4.2.2 Dynamic Schedule Adjustment

We develop a simple adaptive algorithm given that the movement of the elevator is changed from the prediction. Assume that the schedule of $[t', t' + \mathcal{T}]$ period has already been generated, and a change is made at the moment $current\_time$ where $t' < current\_time < t' + \mathcal{T}$. We split the $\mathcal{T}$ into two sections and change a subpart of $\mathcal{T}$. The pseudo-code of the algorithm is in Algorithm 3. Note that the reschedule does not increase the amount of data that have to be transmitted. Therefore, within the same duration, there is enough throughput capacity for the sensors to transmit the data to the mobile sink.

Algorithm 3 Elevator Assisted Data Collection (EADC)

1: Split the remaining part of $\mathcal{T}$ into two sections $T_A = [t', current\_time]$ and $T_B = [current\_time, t' + \mathcal{T}]$.
2: Maintain the schedule in T_A.
3: Reschedule T_B by Algorithm 1 (EAO) with the prediction period be T_B (the same as $\mathcal{T} = t' + \mathcal{T} - current\_time$).

5 Performance Evaluation and Discussion

5.1 Simulation Setup

First, we evaluate our algorithms by simulation. While our algorithms are general, we adopt the parameters of the sensors in our simulation similar to MicaZ [6]. The energy reserve for each source sensor is $\mathcal{E} = 1200\text{mAh} \times 1.2\text{V} \times 2$. The energy consumption for packet reception and transmission is $e_r = 19.7\text{mA} \times 1.8\text{V}$ per 250Kbps and $e_t = 17.4mA \times 1.8V$ per 250Kbps, respectively. The data generation period is $T_R = 2$hours and the default amount of data generated is $\alpha = 2$Mbits per round. This reflects to the data rate of accelerometers, one of the most important sensors used in SHM. According to the experiment results mentioned in Section 2, we assign the transmission rate to be $55Kbps$. We assume that the total time that an elevator stops at a floor would be no less than $T_s = 30$s, and the time that an elevator passes by a floor to be $T_p = 2$s. Thus, the total amount of data that can be transmitted is 1650Kbits and 110Kbits in these two situations. The default value for the deadline of data is $\mathcal{L} = 2$. The default number of floors is $N = 50$.

We assume the elevator movement follows a Markov Chain, see Fig.6. There are two states, MOVE and STAY. After staying at a certain floor for T_s, the elevator will move to an other floor with a probability of p, and might stay at the same floor with a probability of $1 - p$. If the elevator moves, two different movement models are considered: 1) the *OFFICE* model, the elevator would randomly move to a certain floor and all the other

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

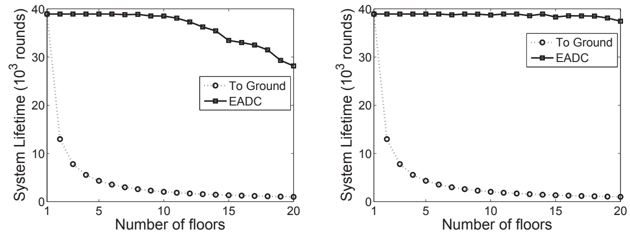
floors have an equal probability to be the destination; and 2) the *TOWER* model, the elevator would only move to the top floor or the ground floor. Different movement probability p reflects the frequency of the elevator movement. In addition, note that setting the time $T_s = 30$s may be a small number for certain elevators (e.g., more crowded elevators carrying a larger number of passengers). By having a small p, we can increase the time for the elevator stops. We note that some elevators are busy in the day time but only stay at the ground floor in the night. We consequently consider two different working time models: 1) the *D&N* model, the elevator would be in movement both in the day time and in the night time; and 2) the *DAY* model, the elevator would be in movement in the day time but stay at the ground floor during the night time (from $11:00$pm to $7:00$am). We admit that this elevator movement model is a simplified model. Nevertheless, we believe it captures certain characteristics of some elevators.

We compare our algorithm (system lifetime in terms of number of rounds) with hop-by-hop data routing and with the optimal offline algorithm. We also implement a simple Waiting for Elevator (WE) algorithm. WE does not predict the movement of elevator. The sensors wait for the elevator passes by or stops. Then the sensors transmit the data directly to the elevator. If the data are about to expire, the sensors transmit the data towards the elevator using hop-by-hop routing.

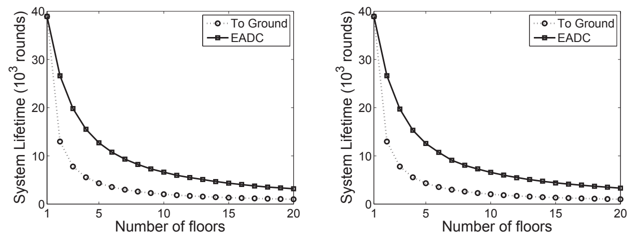
5.2 Simulation Results

We first compare our algorithm EADC with the hop-by-hop routing scheme (See Fig.7). Not surprisingly, the performance of our algorithm is significantly better. To make a comparison even possible, we have to work on some extreme parameters. We set the movement probability to be $p = 10\%$ and compare with number of floors from one to twenty. In D&N model, EADC decreases very modestly; the lifetime is maintained at a high level when the number of floors increases from one to twenty. As a comparison, system lifetime using hop-by-hop routing with twenty floors is only 3% as that of the performance to the one floor case. In the DAY model, when the number of floors increases, the lifetime of both schemes decreases. But EADC decreases slower and always outperforms hop-by-hop routing. As the sensors close to the ground have to relay traffic for others, their load increases fast. For example, when $N = 20$, EADC is four times better than hop-by-hop routing. These results have shown the intrinsic scalability of the elevator-assisted data collection scheme. As the performance of hop-by-hop routing hardly compares to that of EADC, in the rest of our simulation, we only focus on the performance of EADC under various configurations.

We next study the effect of the amount of data generated in each round. We can see from Fig.8 that the more data, the smaller the system lifetime. In Fig.8 (a) and (b), the decrease of EADC is proportional to the increase of the data volume. For example, when the rate

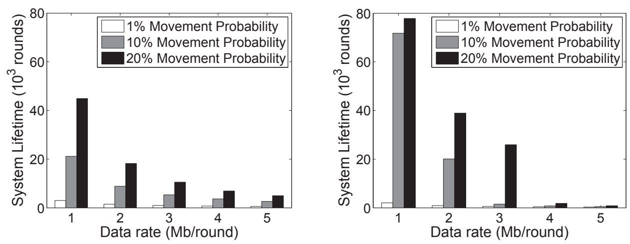


(a) D&N OFFICE model (b) D&N TOWER model

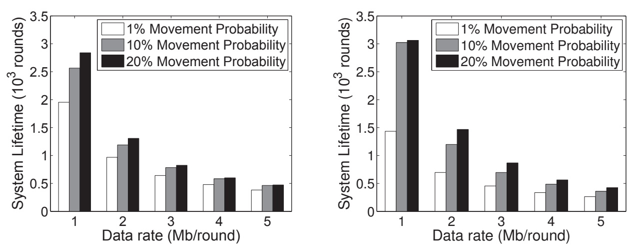


(c) DAY OFFICE model (d) DAY TOWER model

Fig. 7. System lifetime as a function of the number of floors.



(a) D&N OFFICE model (b) D&N TOWER model



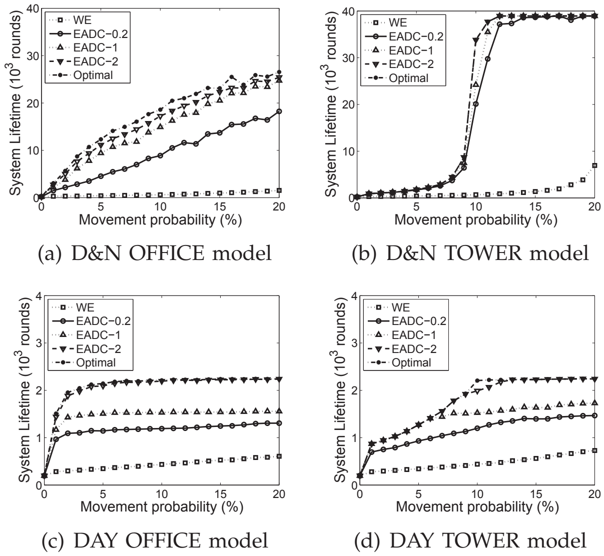
(c) DAY OFFICE model (d) DAY TOWER model

Fig. 8. System lifetime as a function of the amount of data generated in a round.

of data generation is 4Mbits per round, the lifetime is roughly 25% to the case when the rate of data generation is 1Mbits per round. The system lifetime is also fairly proportional to the elevator movement probability p. These are partially because the D&N model has the most evenly distributed movement of the elevator. For Fig.8 (c) and (d), since the elevator stays at the ground floor in the night, the system lifetime is shorter; and the impact of p decreases.

We next study the effect of different prediction period. Here EADC-2, EADC-1, and EADC-0.2 denote the EADC algorithm with a prediction of the elevator movement for the future $\mathcal{T} = 2$ rounds, 1 round and 0.2 round. We compare EADC with the optimal offline solution where the entire movement can be predicted. It is also compared with WE algorithm (Waiting for Elevator) which is

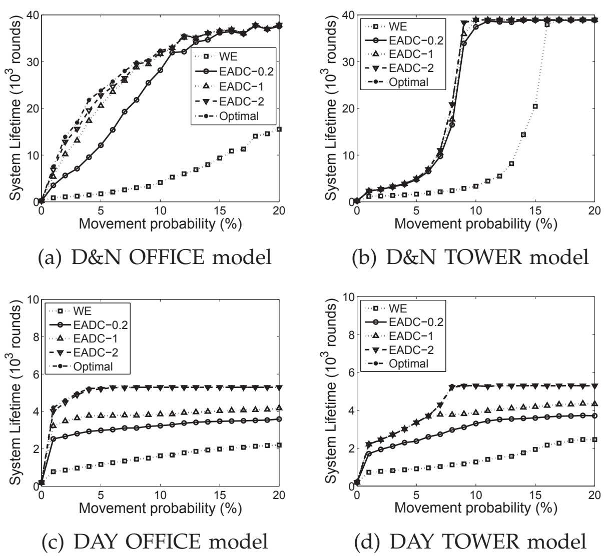
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.



(a) D&N OFFICE model (b) D&N TOWER model

(c) DAY OFFICE model (d) DAY TOWER model

Fig. 9. System lifetime as a function of the elevator movement probability, i.e., p in the building with 50 floors.

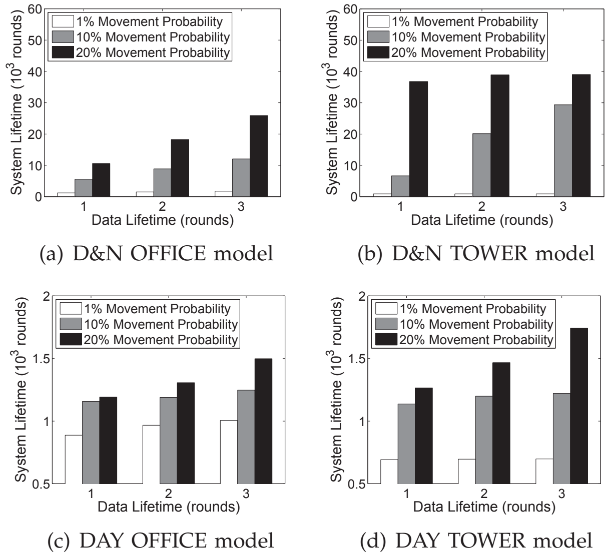


(a) D&N OFFICE model (b) D&N TOWER model

(c) DAY OFFICE model (d) DAY TOWER model

Fig. 10. System lifetime as a function of the elevator movement probability, i.e., p in the building with 20 floors.

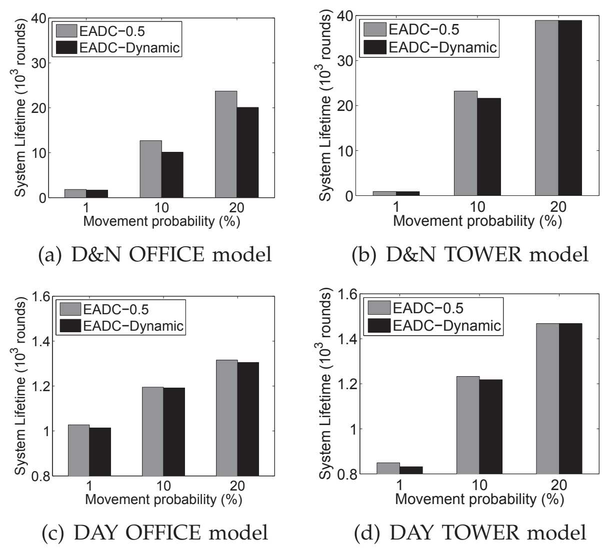
a naive elevator-assisted data collection scheme without data scheduling and prediction. The results are shown in Fig.9 and Fig.10. It is obvious that prediction would greatly improve the system lifetime and the longer prediction time the better the system performance. In Fig.9 (a), the performance of $\mathcal{T} = 0.2$ round is 30% to 70% to that of the optimal offline solution, where the larger the movement probability, the closer the gap. If the prediction period is 2 rounds, the performance is almost the same to the optimal offline solution. In Fig.9 (c) and (d), we have two observations. First, the system lifetime flattened when the elevator movement probability is larger than $5\% \sim 10\%$. Again, this is because the effect of increasing elevator movement probability is reduced by the elevator inactivity in the night time. Second, there is a



(a) D&N OFFICE model (b) D&N TOWER model

(c) DAY OFFICE model (d) DAY TOWER model

Fig. 11. System lifetime as a function of data lifetime.



(a) D&N OFFICE model (b) D&N TOWER model

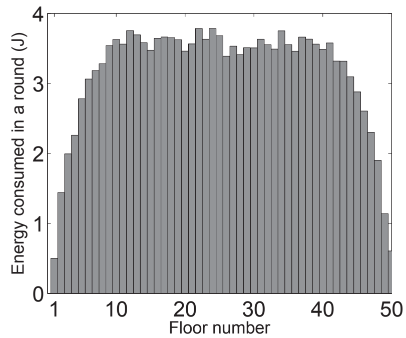
(c) DAY OFFICE model (d) DAY TOWER model

Fig. 12. Performance of dynamic schedule adjustment under imprecise prediction scenario.

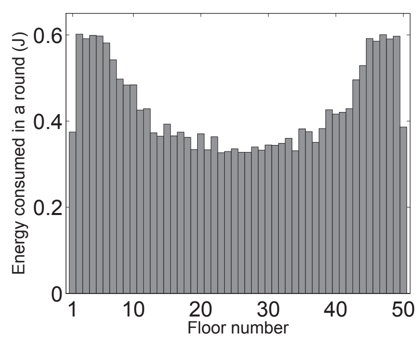
larger gap if the prediction time is short. This is because during the switch from day time to night time, a short prediction time may result in less optimal scheduling. In Fig.10, the system lifetime is longer as there is less number of floors. We have observed the same trend as that of Fig.9.

Fig.11 shows the impact of data lifetime. Clearly, if the SHM applications can allow a longer data lifetime, a better performance can be achieved. A longer data lifetime can allow the algorithm to schedule the data transmission more efficiently. We argue that, however, this is not applicable for the hop-by-hop routing. Generally speaking, the performance of hop-by-hop routing is affected by the data volume only. A unique feature of the elevator-assisted data collection is that we have a chance to adapt to the movement of the elevators, which is different from time to time. Therefore, a larger deadline

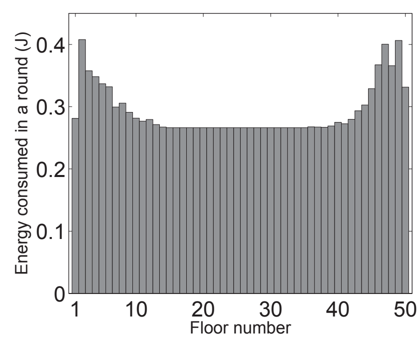
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.



(a) Elevator movement frequency $p = 1\%$.

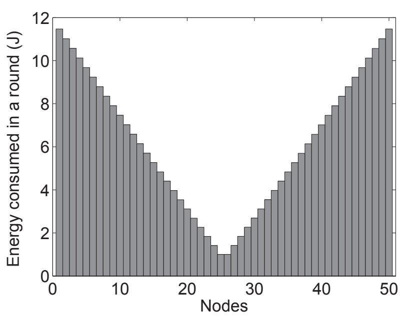


(b) Elevator movement frequency $p = 10\%$.

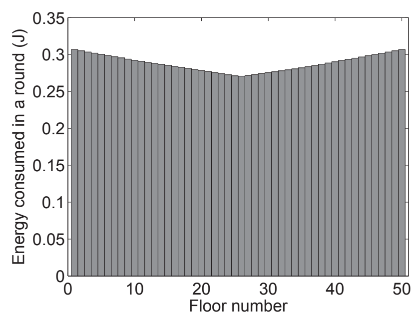


(c) Elevator movement frequency $p = 20\%$.

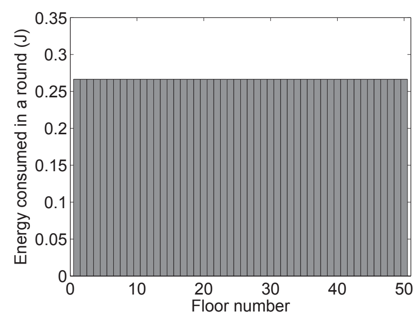
Fig. 13. Energy consumption of each sensor with D&N OFFICE model



(a) Elevator movement frequency $p = 1\%$.

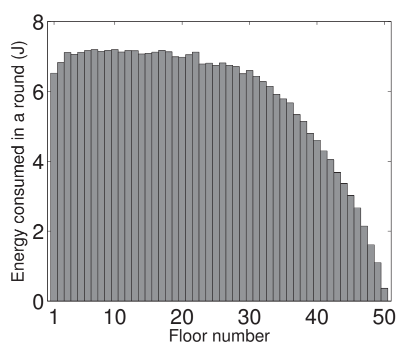


(b) Elevator movement frequency $p = 10\%$.

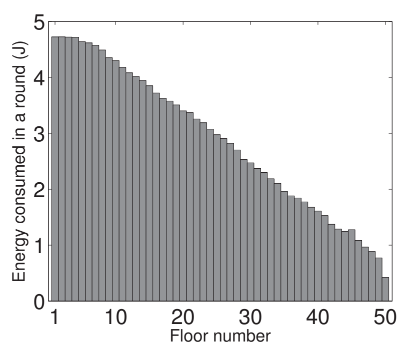


(c) Elevator movement frequency $p = 20\%$.

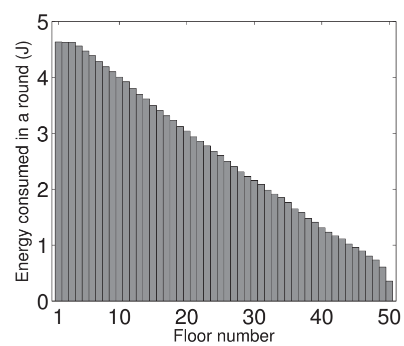
Fig. 14. Energy consumption of each sensor with D&N TOWER model



(a) Elevator movement frequency $p = 1\%$.

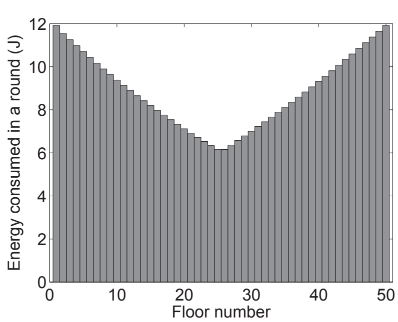


(b) Elevator movement frequency $p = 10\%$.

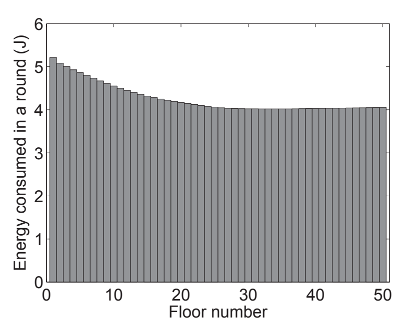


(c) Elevator movement frequency $p = 20\%$.

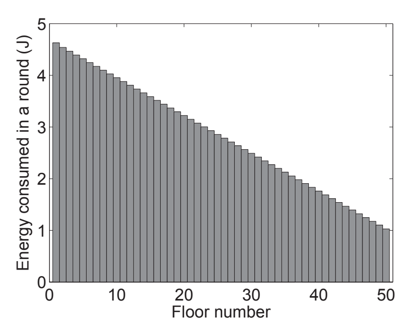
Fig. 15. Energy consumption of each sensor with DAY OFFICE model



(a) Elevator movement frequency $p = 1\%$.



(b) Elevator movement frequency $p = 10\%$.



(c) Elevator movement frequency $p = 20\%$.

Fig. 16. Energy consumption of each sensor with DAY TOWER model

can give room to more optimized data scheduling.

We next examine the performance of dynamic schedule adjustment of EADC. Our simulation is based on a prediction period of half a round. To simulate the dynamic, a new set of movement is generated every quarter of a round using the same elevator movement model. The algorithm will adapt to such dynamics by producing a new schedule accordingly. The results are shown in Fig.12. We can see that the performance of our algorithm is only slightly affected by such dynamics.

We further look at the distribution of the energy consumed of each sensor at the end of system lifetime; averaged to each round. We can see three distinct distributions with different movement frequencies p. In D&N OFFICE model, if $p = 10\%$ (Fig.13 (b)), the sensors around floor [3-10] and floor [40-47] have the highest load. Based on our elevator movement model, the elevator will more likely to pass or stop at the floors in the middle of the building; and the sensors on these floors have to relay more traffic. For a good strategy to minimize energy consumption, sensors around floor 5 and floor 45 have a shorter path to the sensors on both sides, thus, they have the highest traffic load. As a sharp contrast, when $p = 1\%$ (Fig.13 (a)), the sensors in the middle of the building have a much higher load. This is because, if the movement is very infrequent (where the elevator may stay at a certain floor for a long time), the data have to be transmit, even though this requires a much longer path, due to data deadline constraint.

In the DAY movement model, the sensor on the ground always consumes the most power for the elevator stops at ground most of the time. This can be seen clearly from Fig.15 and 16. In the TOWER model, the elevator only stops at the ground floor or the top floor. Therefore, the sensors on the top floor and ground floor consumes more energy than others, as can be seen in Fig.14 and 16.

As a summary, the elevator-assisted data collection is much more efficient than the hop-by-hop routing. Generally, we would notice that lifetime of sensor network would benefit from more frequently movement of the elevator, a longer data lifetime, and a better prediction.

6 MicaZ Testbed Experiments

6.1 Testbed Setup

We implement our algorithm in TinyOS with MicaZ testbed to test the performance in real system. Considering the difficulty of going through specific regulations of GNTVT, we modified our experiments as follows.

We use real data trace of the sensors of GNTVT to serve as traffic intensity model. There are 2-4 accelerometers deployed on the monitored floor. Six elevators could be used to deliver the data. Each elevator would afford about 0.1-0.2KB/s data rate. We track the elevator movement of Chemistry Building with 11 floors and Meng Minwei Building with 27 floors in Nanjing University to serve as the elevator movement model. We record the elevator movement as a sequence of two-tuple <floor, time>, and map the elevator movement tuples to the robot car experiment scenario by adjusting the stopping time of the car. A sink node is tied to a robot car EXKJ-ZN02X to simulate an elevator. The stopping position was specified with the assistance of the infrared sensor of the robot car. The testbed is shown in Fig.17.

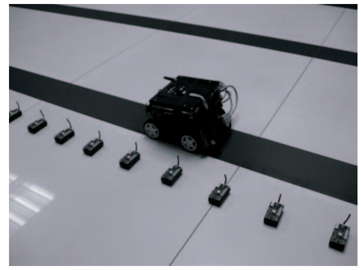


Fig. 17. MicaZ testbed

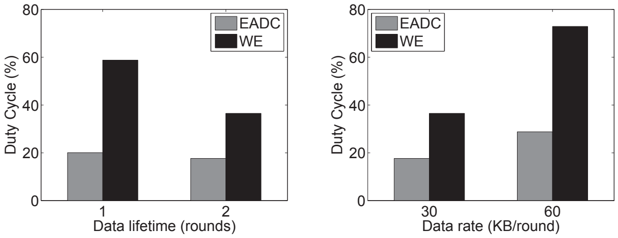
We configure MicaZ motes to transmit data with minimum power. The transmission range is set to $20cm$ to simulate the transmission between floors in the limited laboratory space. The motes farther than 20 cm could also receive data sometimes. So, the sink might be within range of more than one sensor. In our experiments, the sink knows its own position and pulls data from the specified sensors to reduce the conflict between sensors.

In our experiments, a round is 5 minutes and the lifetime of data is 1 or 2 rounds. We configure each sensor to generate 30-60KB data per round. This setting satisfies the data rate requirement in real application of GNTVT. We run the experiment 2 hours each time and 4 times for each setting. Then we get the average results of the experiments.

In the EADC algorithm, we evaluate the energy of each node. In our experiment, we cannot deplete our sensor nodes. As such, we apply the EADC algorithm in a low duty cycle system. The computation of schedule is centralized at the powerful sink node and the schedule is transmitted to the sensors hop-by-hop. The control data consists of items $[DIRECTION, DATA\_AMOUNT]$. Each item consists of 4 bytes (1 bit direction flag, 31 bits data amount). The ith item presents the data flow direction and data amount between sensor i and $i+1$. In our experiments, a sensor only needs to deliver $40 \sim 104$ bytes control data in a schedule period.

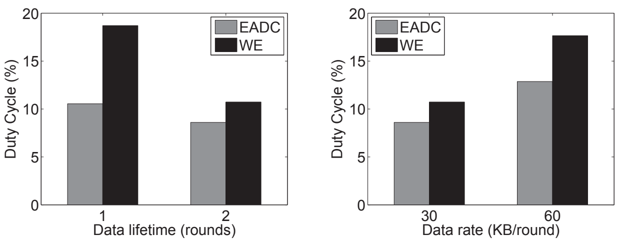
We would transmit the data to the proper sensors waiting for elevator. In our system, the cache size is limited to 500KB. The target sensors or the sensors in the route might have not enough memory to store the data. In that case, we could cache the data in the previous nodes in the route. When the target node has transmitted the data to the elevator or other nodes, it would signal the neighbors. Then the neighbors would continue to transmit the cached data.

We assume that we could only predict one way movement of elevator. For example, assuming that the elevator would stop at floors 1, 5, 10, 8, 2, 1,... sequentially, we could predict the movement 1, 5, 10. When the



(a) Duty cycle as a function of data lifetime (b) Duty cycle as a function of amount of data

Fig. 18. Maximum duty cycle of sensors in the building with 27 floors.



(a) Duty cycle as a function of data lifetime (b) Duty cycle as a function of amount of data

Fig. 19. Maximum duty cycle of sensors in the building with 11 floors.

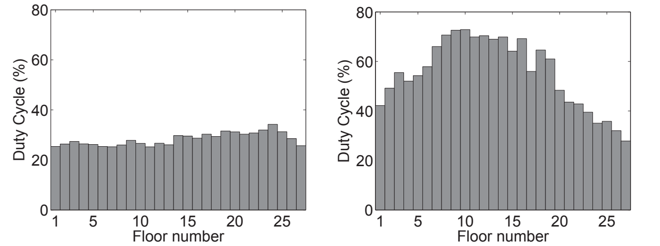
elevator arrives at 10th floor, we could further predict the movement 10, 8, 2, 1. This prediction is not a fixed period but measured by steps. That is applicable, for we could get that information according to the button pressed by the passengers.

We chose low duty cycle MAC protocol X-MAC [4] in our implementation. The sensors keep asleep most of the time when there is no data to be transmitted. In practice, this brings some extra latency introduced from the MAC, which results in a slight reduction of the link capacity between sensors and the elevator. In our implementation, we adjusted the capacity between sensors and elevator. We tried different sleeping period settings of 100ms, 200ms and 500ms and observed that 200ms period is more efficient with little extra latency. So, we set the period to be 200ms in our experiments.

As a comparison, we also implement WE scheme. We evaluate the energy consumption according to the duty cycle of sensors. The results show that our algorithm can reduce the duty cycle efficiently.

6.2 Testbed results

First we set the data rate to be 30KB/round and study the effect of the data lifetime. Then, we fix the data lifetime to be 2 rounds and evaluate the effect of data rate. The results in 27-floor building are show in Fig.18. It is obvious that EADC always outperforms WE. The duty cycles of both EADC and WE are decreasing with data lifetime and increasing with data rate. That is corresponding to the simulation results. We could also find out that, in Fig.18(a), the difference between EADC and WE is larger when the data lifetime is shorter. With WE, the sensors are purely waiting for the elevator. When the



(a) EADC (b) WE

Fig. 20. Duty cycle of each sensor in the building with 27 floors.

lifetime is short, WE has little chance to transmit data to elevator directly, and most of the energy is wasted by hop-by-hop transmitting to elevator; EADC could properly schedule the data transmission and maintain the duty cycle at a low level. In Fig.18(b), when the data rate increases, the increasing rate of WE's duty cycle is much higher than EADC. That also shows the efficiency of EADC algorithm. Similar results could be found in 11-floor building experiments in Fig.19.

Fig.20 shows the duty cycle of each sensor in 27-floor building while the data rate is 60KB/round and data lifetime is 2 rounds. We could find out that EADC algorithm could balance the energy consumption much better than WE algorithm. With WE, the sensors in the middle consumes much more energy than the others. That is because when the data are about to expire, the sensors have to transmit data to the elevator hop-by-hop; and the sensors in the middle are likely to relay more traffic. That is similar to the scenario where the elevator seldom moves as in Fig.13 (a).

We had observed some packet delivery miss in our prototype system. The data delivery ratio for 60Kbytes for 27 nodes is 93% while it is 99% for 30Kbytes or for 11 nodes. We think further parameter adjustment may achieve some better delivery ratio. Based on our experience, in practice it is impossible to achieve 100% packet delivery. A common strategy used in wireless sensor network today is to put some application layer coding to mask the failure.

7 Related Work

Sensor networks have long been used for structural health monitoring in civil engineering [7][8]. Many of such applications adopt wired systems; for example, the monitoring system on Ting Kau Bridge, Hong Kong [1]. The development of wireless sensors and the associated networking and communication techniques have shown great impact on the SHM systems. Experiment systems using the Mica-like sensors are developed [9][10]. The experiments both in-door and on the Golden Gate Bridge of San Francisco have shown that current wireless smart sensors are promising to be applied on SHM systems in the future. The BriMon project [11] shows promising experiments on railway bridge monitoring. We are currently in the development of a SHM system for the Guangzhou New TV Tower [2] for life-long monitoring.

Though the system is still dominated by the wired sensors, wireless substations are used for data transmission. We have a full scale of experiments focusing on high quality sensor placement using wireless sensors for in-construction monitoring in [12].

Using mobile sinks to assist data collection have been proposed previously in literature. Luo et al. use mobile sinks to balance the sensor load and maximize network lifetime; both theoretical analysis [13] and routing protocol implementation [14] are introduced. In [15], single hop transmission is applied, and the sink is expected to visit all the sensors. In [16], some nodes serve as rendezvous points to aggregate data and the mobile sink visits these nodes to collect data. A theoretical study of mobile sink with network lifetime bound is recently studied in [17]. Nevertheless, in all these works, the mobility of the mobile sink can be controlled. That is, the mobile sinks are programmed to perform certain tasks to optimize system performance, which is very different from ours. In [18][19], the mobile sinks cannot be controlled, and protocols are designed to resolve the disadvantages of mobility but not make use of it. In [20], the mobile sink cannot be controlled either, but the data would never be out of date; so the sensors transmit data to mobile sink only when they are close. TwinRoute is proposed in [21], a hybrid algorithm that combines the proactive and reactive data collection techniques, which outperform the pure approaches. In our work, we forward the data to certain sensors waiting for the sink. Though the sink mobility is not manageable, we can adjust our schedule dynamically as described in Section 4.2.2. As such, it is not a pure proactive work. Furthermore, we focused on balancing energy consumption between sensors not the total energy consumption as in [21]. In our scenario, we face the uniqueness of the SHM application and we leverage the advantage of the elevators for data collection; which is very different from these previous studies.

There are many works on mobile ad hoc network routing, e.g., DSR [22], AODV [23]. The objectives and the treatment of these works are very different from our problem.

There is another set of study on Delay Tolerant Network [24][25][26]. In DTN, a wide range of performance metrics such as throughput, delay and reliability are studied. In this paper, we are more focused on a specific elevator-assisted data collection problem, which have never been studied before, and concentrate on the energy efficiency of the sensor networks with data routing.

8 Conclusion and Future Work

In this paper, we have proposed a novel elevator-assisted data collection scheme; targeting on structural health monitoring applications. Intrinsically, the free movement of the elevators substantially decreases the communication distance; and thus the energy consumption. We provide a comprehensive study on the problem; where we want to guarantee the data to be delivered on time and maximize system lifetime. We first study three different offline scenarios and show that all these scenarios can be solved optimally. We further study the online algorithms. We show that the online scenario is significantly more difficult and no online algorithm can achieve a constant competitive ratio as opposed to an optimal offline algorithm. We thus developed online algorithms based on the level of knowledge of the future movement information of the elevators. We show that the exact elevator movement sequence and arriving time at certain floors are not necessary. The simulation and real sensors experiments have shown the effectiveness of our scheme.

As the first work of the elevator-assisted data collection problem, this paper confines itself to the experimental validation, and understanding of its intrinsic complexity and algorithmic solutions from the angle of traffic flow scheduling. We believe there are many future works. We will study the joint consideration of MAC layer optimization and advanced algorithms with consideration of sensor storage.

Acknowledgment

The authors would like to thank the anonymous reviewers and the editors for their invaluable suggestions that improved the quality of this paper. Dan Wang's work is supported by grants Hong Kong PolyU/G-YG78, A-PJ19, 1-ZV5W, A-PK95, A-PL23, 4-BC01, 4-BC02, 4-BC03, and RGC/GRF PolyU 5305/08E. Tao Zhang, Li-Jun Chen and Daoxu Chen's work is supported by the 973 Program of China (No. 2006CB303000 and 2009CB320705), the National Natural Science Foundation of China (No. 60873026 and 61021062), Science and Technology Support Program of Jiangsu Province (No.BE2010178), Industrialization of Science Program for University of Jiangsu Province (No.JH10-3), Yi Qing Ni's work is supported by HKPolyU Project No. G-U845.

References

[1] J. M. Ko, Y. Q. Ni, H. F. Zhou, J. Y. Wang, and X. T. Zhou, "Investigation concerning structural health monitoring of an instrumented cable-stayed bridge," *Structure and Infrastructure Engineering*, vol. 5, no. 6, pp. 497–513, 2009.

[2] Y. Q. Ni, Y. Xia, W. Y. Liao, and J. M. Ko, "Technology innovation in developing the structural health monitoring system for Guangzhou New TV Tower," *Structural Control and Health Monitoring*, vol. 16, no. 1, pp. 73–98, 2009.

[3] F. Wang, D. Wang, and J. Liu, "Poster: Exploiting elevator for wireless sensor data collection in high-rise structural monitoring," in *Proc. MobiCom'10*, 2010.

[4] M. Buettner, G. Yee, E. Anderson, and H. Han, "X-MAC: A short preamble mac protocol for duty-cycled wireless sensor networks," in *Proc. ACM SenSys'06*, Nov. 2006, pp. 307–320.

[5] R. Ahuja, T. Magnanti, and J. Orlin, *Network Flows: Theory, Aplgorithms, and Applications*. Englewood Cliffs, NJ: Prentice Hall, 1993.

[6] *MICAZ motes*. [Online]. Available: http://www.crossbow.com

[7] A. Rao and G. Anandakumar, "Optimal sensor placement techniques for system identification and health monitoring of civil structures," *Smart Structures and Systems*, vol. 4, no. 4, pp. 465–492, 2008.

[8] Z. Shi, S. Law, and L. Zhang, "Optimum sensor placement for structural damage detection," *Journal of Engineering Mechanics*, vol. 126, pp. 1173–1179, 2000.

[9] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," in *Proc. ACM IPSN'07*, Apr. 2007.

[10] N. Xu, S. Rangwala, K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, "A wireless sensor network for structural monitoring," in *Proc. ACM SenSys'04*, Nov. 2004.

[11] K. Chebrolu, B. Raman, N. Mishra, P. K. Valiveti, and R. Kumar, "BriMon: A sensor network system for railway bridge monitoring," in *Proc. ACM MobiSys'08*, Breckenridge, CO, Jun. 2008.

[12] B. Li, D. Wang, F. Wang, and Y. Q. Ni, "High quality sensor placement for structural health monitoring: Refocusing on application demands," in *Proc. IEEE INFOCOM'10*, San Diego, CA, Mar. 2010.

[13] J. Luo and J.-P. Hubaux, "Joint mobility and routing for lifetime elongation in wireless sensor networks," in *Proc. IEEE INFOCOM'05*, Miami, FL, Mar. 2005.

[14] J. Luo, J. Panchard, M. Pirkowski, M. Grossglauser, and J. pierre Hubaux, "Mobiroute: Routing towards a mobile sink for improving lifetime in sensor networks," in *Proc. IEEE/ACM DCOSS*. Springer Verlag, 2006, pp. 480–497.

[15] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using predictable observer mobility for power efficient design of sensor networks," in *Proc. IPSN'03*, 2003, pp. 129–145.

[16] G. Xing, T. Wang, W. Jia, and M. Li, "Rendezvous design algorithms for wireless sensor networks with a mobile base station," in *Proc. MobiHoc'07*, 2008, pp. 231–240.

[17] Y. Shi and T. Hou, "Theoretical results on base station movement problem for sensor networks," in *Proc. IEEE INFOCOM'08*, 2008.

[18] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon, "Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks," in *Proc. ACM SenSys'03*, 2003, pp. 193–204.

[19] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A two-tier data dissemination model for large-scale wireless sensor networks," in *Proc. ACM MobiCom'02*, Jun. 2009.

[20] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: modeling a three-tier architecture for sparse sensor networks," in *Proc. IEEE SNPA'03*, Seattle, WA, May 2003.

[21] R. Wohlers, N. Trigoni, R. Zhang, and S. Ellwood, "TwinRoute: Energy-efficient data collection in fixed sensor networks with mobile sinks," in *Proc. MDM'09*, 2009, pp. 192–201.

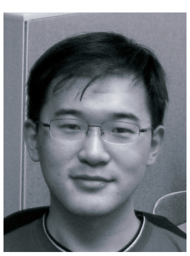
[22] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Kluwer Academic, 1996.

[23] C. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," in *WMCSA'09*, Feb. 1999.

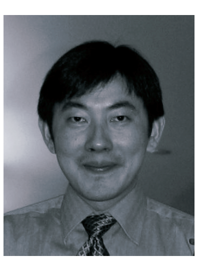
[24] K. Fall, "A delay-tolerant network architecture for chanllenged internets," in *Proc. SIGCOMM'03*, Karlsruhe, Germany, Aug. 2003.

[25] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," in *Proc. SIGCOMM'04*, Portland, OR, Aug. 2004.

[26] W. Zhao, M. Ammar, and E. Zegura, "Controlling the mobility of multiple data transport ferries in a delay-tolerant network," in *Proc. IEEE INFOCOM*, 2005.



Tao Zhang received the BSc degree in computer science from Nanjing University, China, in 2004. He is currently a Ph.D. student in the Department of Computer Science & Technology of Nanjing University under the supervision of Professor Daoxu Chen. He worked as a research assistant in the Department of Computing at Hong Kong Polytechnic University under the supervision of assistant professor Dan Wang. His current research interests include wireless sensor networks, RFID, and distributed computing.



Dan Wang (S05-M07) received the B.Sc degree from Peking University, Beijing, China, in 2000, the M.Sc degree from Case Western Reserve University, Cleveland, Ohio, USA, in 2004, and the Ph. D. degree from Simon Fraser University, Burnaby, B.C., Canada, in 2007; all in computer science. He is currently an assistant professor at the Department of Computing, The Hong Kong Polytechnic University. His research interests include wireless sensor networks, Internet routing and applications. He is a member of the IEEE.

Jiannong Cao received the BSc degree in computer science from Nanjing University, China, and the MSc and PhD degrees in computer science from Washington State University, Pullman. He is currently a professor and associate head in the Department of Computing at Hong Kong Polytechnic University. He is also the director of the Internet and Mobile Computing Lab in that department. His research interests include mobile and pervasive computing, computer networking, parallel and distributed computing, and fault tolerance. His recent research has focused on wireless networks and mobile and pervasive computing systems, developing test-beds, protocols, middleware, and applications. He has published more than 250 technical papers in the above areas. He is the coordinator in Asia of the Technical Committee on Distributed Computing (TPDC) of IEEE Computer Society. He has served as an associate editor and a member of editorial boards of several international journals, including Pervasive and Mobile Computing, Wireless Communications and Mobile Computing, Peer-to-Peer Networking and Applications, and the Journal of Computer Science and Technology. He has also served as a chair or member of organizing/program committees for many international conferences, including PERCOM, ICDCS, IPDPS, ICPP, RTSS, SRDS, MASS, PRDC, ICC, GLOBECOM, and WCNC. He is a senior member of China Computer Federation, the IEEE, the IEEE Computer Society, and the IEEE Communication Society, and a member of ACM.

Yi Qing Ni received the BEng and MSc degrees from Zhejiang University, China in 1983 and 1986, respectively, and the PhD degree from The Hong Kong Polytechnic University, Hong Kong in 1997. He is now a professor of civil and structural engineering, and the coordinator of university-wide interdisciplinary research on railway-related projects at The Hong Kong Polytechnic University, Hong Kong. His current research interests include structural health monitoring, structural dynamics and control, smart materials and structures, sensors and actuators, and monitoring and control in rail engineering.

Li-Jun Chen received his B.S. degree in Electrical Engineering from the Xi'an University of Science and Technology, P. R. China, in 1982 and his M.S. degree and Ph.D. degree, both in Automatic Control from China University of Mining and Technology, P. R. China, in 1993 and 1998 respectively. He was a Post Doctoral Fellow at the Nanjing University (1998 -2000), P. R. China and the Michigan State University (2001-2002), U.S.A, and a Visiting Scholar at The Hong Kong Polytechnic University in 2007. His current research interests include distributed computing and wireless sensor networks.

Daoxu Chen was a visiting researcher at Purdue University and the City University of Hong Kong. He is now a professor in the Department of Computer Science and Technology, Nanjing University. His research interests include distributed computing, parallel processing, and computer networks. He is a member of the IEEE, the ACM, and a senior member of the China Computer Federation.