An Optimal Task Scheduling Algorithm in Wireless Sensor Networks

L. Dai, Y. Chang, Z. Shen

Liang Dai, Yilin Chang, Zhong Shen
State Key Laboratory of Integrated Service Networks
Xidian University
Xi'an 710071, China
E-mail: ldai1981@gmail.com, ylchang@xidian.edu.cn, zhshen@mail.xidian.edu.cn

Abstract: Sensing tasks should be allocated and processed among sensor nodes in minimum times so that users can draw useful conclusions through analyzing sensed data. Furthermore, finishing sensing task faster will benefit energy saving, which is critical in system design of wireless sensor networks. To minimize the execution time (makespan) of a given task, an optimal task scheduling algorithm (OTSA-WSN) in a clustered wireless sensor network is proposed based on divisible load theory. The algorithm consists of two phases: intra-cluster task scheduling and inter-cluster task scheduling. Intra-cluster task scheduling deals with allocating different fractions of sensing tasks among sensor nodes in each cluster; inter-cluster task scheduling involves the assignment of sensing tasks among all clusters in multiple rounds to improve overlap of communication with computation. OTSA-WSN builds from eliminating transmission collisions and idle gaps between two successive data transmissions. By removing performance degradation caused by communication interference and idle, the reduced finish time and improved network resource utilization can be achieved. With the proposed algorithm, the optimal number of rounds and the most reasonable load allocation ratio on each node could be derived. Finally, simulation results are presented to demonstrate the impacts of different network parameters such as the number of clusters, computation/communication latency, and measurement/communication speed, on the number of rounds, makespan and energy consumption.

Keywords: wireless sensor networks; divisible load theory; multi-round task scheduling.

1 Introduction & Motivation

Owing to the wireless sensor network node with limited energy, the task should be completed within the shortest possible amount of time. Divisible load theory [1] provides an effective solution to wireless sensor networks for task scheduling [2-5]. Different from other heuristic solutions of task scheduling problem in wireless sensor networks [6, 7], this scheme can get not only the optimal solution, but also the analytic solution, thus ensuring the consistency of the results of scheduling.

Divisible load theory has been intensively studied in the past decades. It is mainly concerned with obtaining an optimal partitioning and scheduling strategy for a given task such that it can be processed in the shortest amount of time. Divisible load scheduling algorithm can be divided into single-round scheduling algorithms [2-5] and multi-round scheduling algorithms [8, 9]. Single-round scheduling algorithm is relatively simple, but its computation and communication overlap are rather poor, and the extra overhead is relatively large. Single-round

scheduling algorithms were applied to wireless sensor networks in [2-5]. Although the authors derived closed-form solutions to obtain the optimal finish time, the network topology discussed in those papers is single-level tree structure. While in wireless sensor networks, as compared with the single-level tree structure, clustered structure (multi-level tree structure) has a great of advantages [10]. Multi-round scheduling algorithm has the characteristics of better computation and communication overlap, thus properly reducing the scheduling overhead. However, it is more difficult to analyze, so fewer results of multi-round scheduling algorithm are available and the existing multi-round scheduling algorithms are designed based on grid computing environments.

Therefore, we present a multi-round task scheduling algorithm (OTSA-WSN) in clustered wireless sensor networks. The goal of this algorithm is to minimize the overall execution time (hereafter called makespan) and fully utilize network resources, by finding an optimal strategy of splitting the original tasks received by SINK into a number of sub-tasks as well as distributing these sub-tasks to the clusters in the right order.

2   Optimal Scheduling Algorithm

Wireless sensor networks construct clusters several times in its life cycle. Each cluster will have a set-up phase and a steady-state phase[10]. We discuss our multi-round task scheduling algorithm in a steady-phase phase.

The original tasks received by SINK are divided into two stages: inter-cluster task scheduling and intra-cluster task scheduling. First, inter-cluster task scheduling partitions the entire tasks into each cluster, and then the sub-tasks in a cluster is assigned to each intra-cluster sensor node by intra-cluster task scheduling. To improve overlap of communication with computation, inter-cluster task scheduling assigned sensing tasks among all clusters in multiple rounds.

According to divisible load theory, to remove performance degradation caused by communications interference, SINK sends each round's tasks to cluster heads sequentially. After each cluster finishing its tasks and fusing the data, the cluster heads also send this round's results to SINK sequentially. That in every moment only allows SINK node sends sub-tasks to a cluster head, or a cluster head return fusion data to the SINK.

2.1   Intra-cluster task scheduling

In order to ensure that the tasks are processed orderly, SINK allocates the tasks to each cluster according to the task-processing rate of each cluster, which guarantees that the task execution time of all clusters in each round remains the same.

Definition: The task-processing rate of a cluster is the average rate the cluster takes to complete the intra-cluster tasks, that is the number of tasks dealt (measurement and reporting data) per unit of time.

Assuming there are k nodes in a cluster, according to divisible load theory, the cluster's task-processing rate is as follows:

$$S = (1 + \sum_{i=2}^{k} \prod_{j=2}^{i} h_j)/(1/s_1 + 1/b_1) \qquad (1)$$

where $h_i = (1/s_{i-1})/(1/s_i + 1/b_i)$, $i = 2, \cdots, k$, where s_i is node i 's measuring rate, in this case, the number of tasks completed per unit of time, b_i states node i 's transmitting rate to cluster head, in this case, the number of tasks transmitted per unit of time.
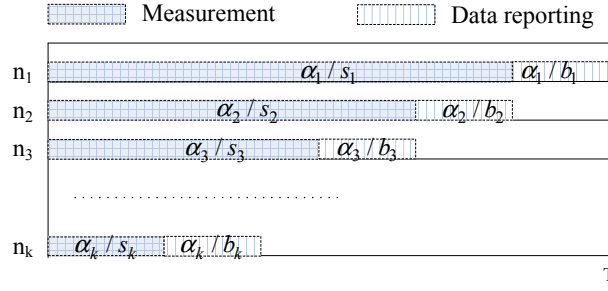
Figure 1: Timing diagram for in-cluster task-processing

Proof: α_i is defined as the fraction of sensing task assigned to node n_i by the cluster head. It is assumed that every node will be assigned non-zero task, i.e., $0 < \alpha_i < 1$, and the task for all nodes in this cluster sums to 1.

By definition we can see:

$$\sum_{i=1}^{k} \alpha_i = 1 \tag{2}$$

So, the time for node n_i measuring its tasks and reporting results to cluster head are α_i/s_i and α_i/b_i, respectively.

One cluster head and a set of sensor nodes constitute a cluster where each node is able to communicate with the cluster head directly. Sensor nodes measure data from surroundings related to the given task, and then report these data to the cluster head. To complete certain amount of sensor readings in minimum finishing time, the sensing task should be allocated to each sensor node and scheduled to avoid transmission conflicts and idle time on the cluster head.

Fig.1 illustrates the timing diagram for a set of sensor nodes, indexed from n_1 to n_k, in one cluster. From Fig.1, it can be observed that there is no time gap between every two successive nodes because the divisible workload can be transferred in the cluster. All sensor nodes start to measure data at the same time. Once the previous node finishes transmitting data, the other one completes its measuring task and starts to report its data. As a result, the proposed timing diagram minimizes the finish time by scheduling the measuring time and reporting time of each senor node. Moreover, since the intra-cluster scheduling tries to avoid the transmission conflicts at the cluster head, energy spent on retransmission are conserved.

The working time of a sensor node can be divided into two parts: measuring time and reporting time.

In Fig.1, one can set up the following corresponding recursive load distribution equations:

$$\alpha_{i-1}/s_{i-1} = \alpha_i/s_i + \alpha_i/b_i, i = 2, 3, ...k \tag{3}$$

Rewriting the above set of equations as:

$$\alpha_i = h_i \alpha_{i-1} \tag{4}$$

where $h_i = (1/s_{i-1})/(1/s_i + 1/b_i)$, $i = 2, 3, ...k$

Using Fig.1, Eq.(2) and Eq.(4), the largest workload for the sensor node can be solved as:

$$\alpha_1 = 1/(1 + \sum_{i=2}^{k} \prod_{j=2}^{i} h_j) \tag{5}$$

Similarly, the workloads of other sensor nodes given by Eq.(4) can be obtained:

$$\alpha_i = \prod_{j=2}^{i} (h_j)/(1 + \sum_{i=2}^{k} \prod_{j=2}^{i} h_j) \tag{6}$$

Fig.1 indicates that when the node with the largest measuring data finishes transmission, the local cluster completes its assigned sensing task. Then, the finish time of measuring and reporting data for the cluster is:

$$T = (1/s_1 + 1/b_1)/(1 + \sum_{i=2}^{k} \prod_{j=2}^{i} h_j) \tag{7}$$

We can get the task-processing rate of the cluster:

$$S = (1 + \sum_{i=2}^{k} \prod_{j=2}^{i} h_j)/(1/s_1 + 1/b_1) \tag{8}$$

It's not difficult to see that in the homogeneous network environment, every cluster have the same parameters, and their task-processing rate is:

$$S = (1 - h^k)/(1/s + 1/b)(1 - h) \tag{9}$$

where $h = (1/s)/(1/s + 1/b)$

Theoretical analysis and simulation in this paper are based on LEACH protocol[10] family (inter-cluster one-hop topology). For the multi-hop networks, namely, the multi-layer tree structure, we can do the calculations on parent node in the tree using Eq.(3)　Eq.(8). According to divisible load theory, the conflict-free scheduling strategy for each parent node is calculated in order to save energy and prolong network lifetime.

In wireless sensor networks, cluster head is responsible for data exchange for SINK and in-cluster nodes. In order to reduce energy consumption caused by transmitting redundant data, lower latency and prolong the survival period, cluster head needs fuse the data [11]. A new estimation method for data fusion information utilization constant is introduced [5] in this paper. Information utilization constant is based on a technique of information accuracy estimation. Through estimating accuracy of information, cluster head can know the approximate percentage of data fusion.

2.2   Inter-cluster task scheduling

The following notations will be used throughout this paper:

- W_{total}: total amount of workload that resides at SINK;

- W_{ji} :number of tasks assigned to cluster i in round j.

- S_i: rate of cluster i's task-processing, that is, the tasks processed in a unit time.

- B_i: downlink communication speed SINK to cluster head i; B_i': uplink transmission rate cluster head i to SINK, that is, the number of tasks transmitted per unit time.

- t_j: processing time of round j.

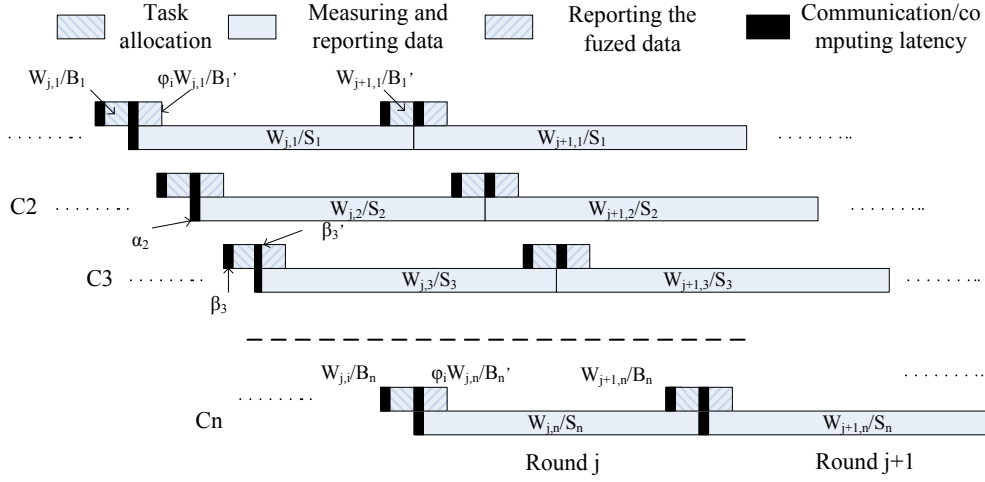- W_j: size of the total load dispatched during round j.

Figure 2: Process of multi-round scheduling algorithm

- φ_i: information utility constant of cluster head i.

Thus, the entire load for cluster i in round j can be sensed (measured, transmitted) is W_{ji}/S_i. In round j, the time for SINK sending tasks to cluster i and for cluster head i sending the fused data are W_{ji}/B_i and $\varphi_i W_{ji}/B_i'$, respectively.

In practical wireless sensor network environment, communication and computing latency caused by pre-initialization are inevitable.

Suppose affine cost parameters are as follows:

- α_i: computing latency of cluster i, that is, the time taken for initialization.

- β_i(resp. β_i'): communication latency incurred by SINK to initiate a data transferring to cluster head i. (resp. start-up time for communication from the cluster head i to SINK).

Fig.2 describes the procedure of SINK dispatching the tasks to each cluster, each cluster measuring and reporting data, as well as cluster heads reporting the fused data to SINK. In this paper, we assume that there are total n clusters in a stable stage, where C_i, $i = 1, \cdots, n$ represents each cluster.

As the computational cost of each cluster remains the same, so there are:

$$\alpha_i + W_{ji}/S_i = t_j, j = 1, \cdots, M - 1 \tag{10}$$

where t_j is only related to the number of rounds j, and M is the optimal scheduling round.

The sum of tasks allocated to every cluster in round j is equal to the tasks in round j:

$$W_j = \sum_{i=1}^{n} W_{ji} \tag{11}$$

From the Eq.(10) and Eq.(11) we can compute:

$$W_{ji} = a_i W_j + b_i \tag{12}$$

where $a_i = S_i / \sum_{k=1}^{n} S_k$, $b_i = (S_i / \sum_{k=1}^{n} S_k) \sum_{k=1}^{n} (S_k \alpha_k) - S_i \alpha_i$.

As shown in Fig.2, in order to fully utilize the bandwidth and avoid cluster waiting among different rounds, SINK must send the tasks allocated in round $j + 1$ to all the cluster heads and receive the fused data from all the cluster heads in the round j, before cluster n finished the tasks in round j. When the time for intra-cluster nodes processing the tasks in round j is exactly equal to the sum of the time for SINK sending sub-tasks to all cluster heads in round $j+1$ and receiving the fused data from all the cluster heads in round j, then, the best bandwidth utilization is achieved, that is:

$$\sum_{i=1}^{n} [(W_{j+1,i}/B_i) + \beta_i + (\varphi_i W_{j,i}/B'_i) + \beta_i'] = t_j \tag{13}$$

Utilizing Eq.(10), Eq.(12) and Eq.(13), we have:

$$W_{j+1} = W_j * \frac{1 - \varphi_i \sum\limits_{i=1}^{n} (S_i/B'_i)}{\sum\limits_{i=1}^{n} (S_i/B_i)} + \frac{\sum\limits_{i=1}^{n} (S_i \alpha_i)}{\sum\limits_{i=1}^{n} S_i \sum\limits_{i=1}^{n} (a_i/B_i)} - \frac{\sum\limits_{i=1}^{n} (\beta_i + b_i/B_i + \beta_i' + \varphi_i b_i/B'_i)}{\sum\limits_{i=1}^{n} (a_i/B_i)} \tag{14}$$

Simplify the Eq.(14) as follows:

$$W_j = \theta^j (W_0 - \eta) + \eta \tag{15}$$

where $\theta = [1 - \varphi_i \sum\limits_{i=1}^{n} (S_i/B'_i)] / \sum\limits_{i=1}^{n} (S_i/B_i)$,

$$\eta = \frac{\sum\limits_{i=1}^{n} (S_i \alpha_i)}{[\sum\limits_{i=1}^{n} (S_i/B_i) + \varphi_i \sum\limits_{i=1}^{n} (S_i/B'_i)] - 1} - \frac{\sum\limits_{i=1}^{n} S_i \sum\limits_{i=1}^{n} (\beta_i + b_i/B_i + \beta_i' + \varphi_i b_i/B'_i)}{[\sum\limits_{i=1}^{n} (S_i/B_i) + \varphi_i \sum\limits_{i=1}^{n} (S_i/B'_i)] - 1}$$

Also the total load is equal to the sum of the tasks allocated in all rounds:

$$\sum_{j=0}^{M-1} W_j = W_{total} \tag{16}$$

The following constraint relations can be obtained:

$$G(M, W_0) = (W_0 - \eta)(1 - \theta^M)/(1 - \theta) + M\eta - W_{total} = 0 \tag{17}$$

The problem of minimizing the total task finish time in scheduling algorithm[8] is described below:

$$EX(M, W_0) = \sum_{j=0}^{M-1} t_j + \frac{1}{2} \sum_{i=1}^{n} [(W_{0,i}/B_i) + \beta_i + (\varphi_i W_{M-1,i}/B'_i) + \beta_i'] \tag{18}$$

The above minimization problem can be solved through the Lagrange multiplication.

$$W_0 = (1 - \theta)/(1 - \theta^M)(W_{total} - M\eta) + \eta \tag{19}$$

After solving W_0 and M, the sizes of all the chunks $W_{j,i}$ can be obtained using Eq.(12) and Eq.(15).

3   Wireless Energy Use

In this section, the energy model of the OTSA-WSN algorithm is presented in detail and the equations of energy consumption of individual sensor nodes are derived. The model is based on first-order radio model [10]. There are three kinds of energy consumption in the wireless sensor network: measurement, data fusion, and communication. Because nodes in the sensor network cooperate with each other via data transmission. Energy consumption of communications exist in sensor nodes, cluster heads and SINK. It is not necessary for cluster heads and SINK to perform any sensing task. Thus, there is no energy cost for cluster heads due to the measurement of these nodes, while the additional energy cost of cluster heads attributes to data fusion. The energy to sense, fuse, and transmit a unit sensory data are denoted by e_s, e_p, and e_{tx}, respectively. Sensor nodes also consume the energy of e_{rx} to receive one unit of data. The distance between the sender and the receiver is d.

The energy use for each kind of nodes is outlined as follows:

Energy use for individual sensor nodes j in cluster i:

$$E_{i,j} = \alpha_{i,j}(e_s + e_{tx}d^2), i = 1, \cdots, k, j = 1, \cdots, n_i \tag{20}$$

Energy use for individual cluster head:

$$E_i = \alpha_i(e_{rx} + e_p + \varphi_i e_{tx}d^2), i = 1, \cdots, k \tag{21}$$

Energy use for SINK:

$$E_{SINK} = \sum_{i=1}^{k} \alpha_i \varphi_i e_{tx} \tag{22}$$

4   Performance Evaluation

In the above sections, we have obtained the optimal number of rounds M for a given sensing task, and energy use for individual sensor nodes. In this section, we investigate the effects of three network parameters, such as the number of clusters, computation/communication latency, and measurement/communication speed, on the number of rounds, makespan and energy consumption in the homogeneous network environment.

In the simulation, the following energy parameters are adopted: transmitting a unit of sensor reading over a unit distance takes e_{tx}=200nJ, receiving one unit of sensor reading consumes e_{rx}=150nJ, measuring one unit of sensor reading needs e_s=100nJ, fusing one unit of observation consumes e_p=20nJ and the distance between the sender and the receiver is d=100m. There are 20 sensor nodes in each cluster.

The simulation results are shown in Fig.3 to Fig.6.

Firstly, Fig.3 plots the M values computed by OTSA-WSN versus computation/communication latency when they vary among 0 and 1.0. Assume the communication speeds of the uplink and downlink between SINK and cluster head are identical, namely, $\beta=\beta'$. As can be seen from Fig.3, M decreases with either the communication or computation latency increasing, owing to the reason that fewer rounds may result in less overhead. Because when the communication latency and computation latency increase, the tasks allocated for each round will increase to meet the requirement, which makes full use of bandwidth. Therefore, the number of rounds will be reduced.

Next, the makesapn against the number of clusters are plotted in Fig.4. Assume that transmission rate of all links between nodes is the same, that is, $B=B' = b$. In Fig.4(a), the value of s
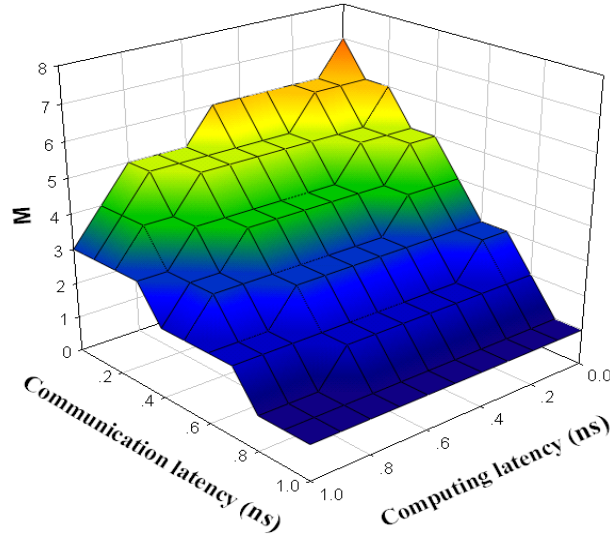
Figure 3: Impact of communications and computing latency on the number of rounds

is chosen from 4 to 10, while b is fixed to 1.0. This figure shows that measurement speed almost does not affect the makespan because sensing takes a small fraction of the entire execution time. Fig.4(b) shows that when the communication speed of sensor nodes increases, the makespan of a given task is reduced. It can be found that the four lines in Fig.4(b) converge when the number of clusters becomes large.

Then, the third simulation is about the energy consumption of intra-cluster sensor nodes. SINK and cluster heads are not taken into account because generally, SINK has no energy constraint and the chosen cluster heads have the possibly enough energy. The network is configured with 20 clusters. Without loss of generality, the intra-cluster sensor nodes in the first cluster are chosen to study the energy consumption, as shown in Fig.5. Fig.5 presents the energy consumption of all the nodes in the first cluster as given by Eq.(20), where the intra-cluster nodes are indexed from 1 to 20. In each case, the energy consumption of sensor nodes monotonically decreases due to the reduced workload. Fig.5(a) shows the higher the in-cluster node's measuring speed, the more evenly the tasks allocated to each nodes, hence the smaller the energy consumption of the nodes. Fig.5(b) presents the larger communication speed between nodes, the smaller the energy consumption of the in-cluster nodes.

Finally, to simulate the OTSA-WSN algorithm in the heterogeneous network environments, the measuring speed of intra-cluster nodes is set as the random numbers vary from 4 to 10, and the communication speed of links from 0.4 to 1.0. Through 8 experiments, the performance of scheduling algorithms in the heterogeneous network environment is analyzed. Fig.6(a) shows the impact of random measuring speed and communication speed on the makespan with the increasing number of clusters in the heterogeneous network environment. Fig.6(b) presents the impact of random measuring speed and communication speed on energy-consuming in the order of tasks allocation. Compared with Fig.4 and Fig.5, it can be seen that, the impact of measuring speed on the makespan and energy consumption is less than communication speed. The makespan decreases with the increasing number of clusters, and the energy consumption are reduced in the order of tasks allocation.
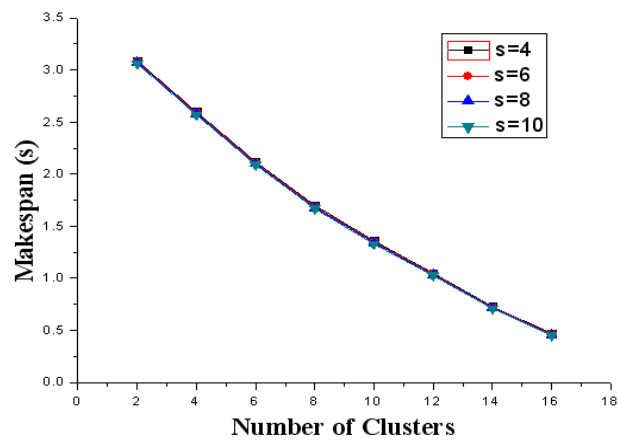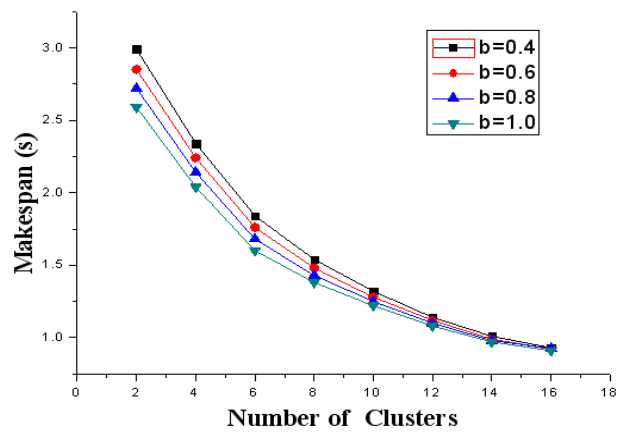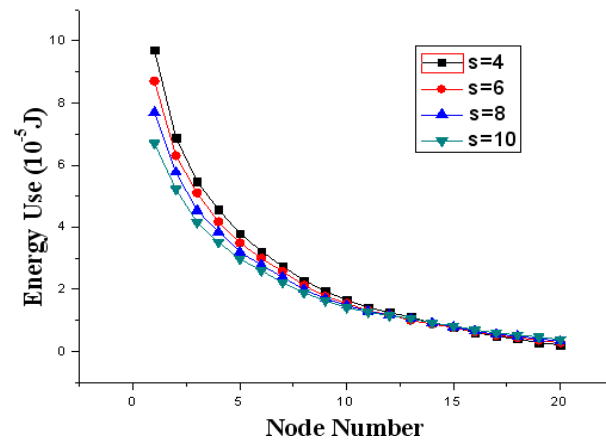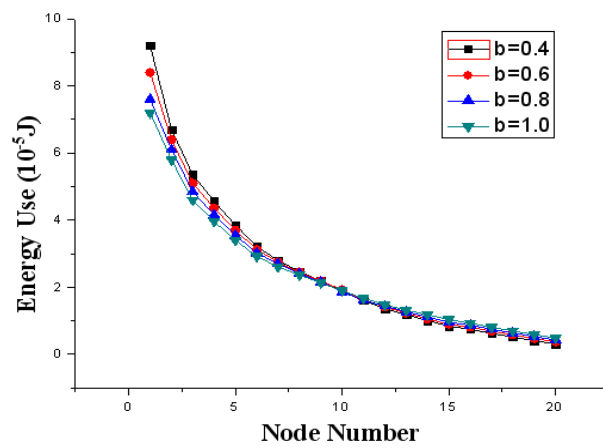
Figure 4: Impact of measuring speed and bandwidth on the makespan
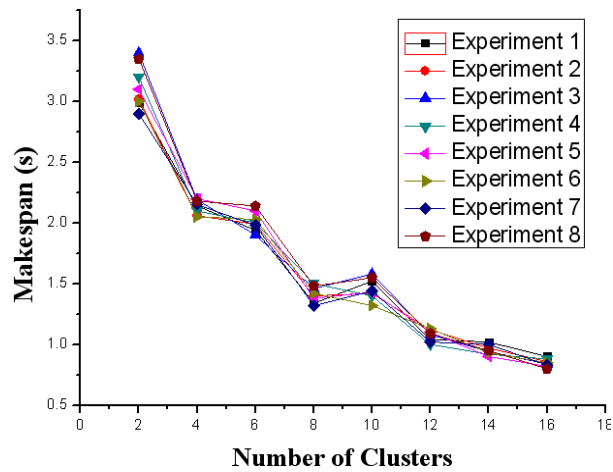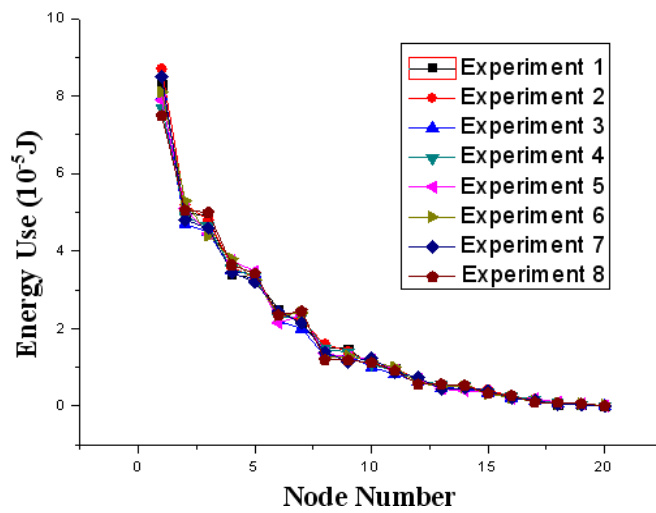
(a)

(b)

Figure 5: The impact of measuring speed and bandwidth on the energy consumption in in-cluster nodes

(a)

(b)

Figure 6: Impact of random measuring speed and bandwidth on the makespan and energy consumption

5   Conclusions

As the wireless sensor network node with limited energy, so the tasks should be completed as quickly as possible, and the network resources should be fully utilized. In this paper, we present a multi-round task scheduling algorithm (OTSA-WSN) in clustered wireless sensor networks. The goal of this algorithm is to minimize the makespan and fully utilize network resources, by finding an optimal strategy of splitting the original load received by SINK into a number of chunks as well as distributing these chunks to the clusters in the right order.

Bibliography

[1] V. Bharadwaj, D. Ghose, T. G.ROBERTAZZI, Divisible Load Theory: A New Paradigm for Load Scheduling in Distributed Systems. *Cluster Computing*, vol.6, No.1, pp.7-18, 2003.

[2] M. Moges, T.G. Robertazzi, Wireless Sensor Networks: Scheduling for Measurement and Data Reporting. *IEEE Transactions on Aerospace and Electronic Systems*, Vol.42, No.1, pp.327-340, 2006.

[3] H. LIU, X. YUAN, M. Moges, An Efficient Task Scheduling Method for Improved Network Delay in Distributed Sensor Networks. *In Proceedings of TridentCom 2007*, Orlando, FL, USA, 1-8, 2007.

[4] H. LIU, J. SHEN, X. YUAN, M. Moges, Performance Analysis of Data Aggregation in Wireless Sensor Mesh Networks, *In Proceedings of Earth & Space 2008*, Akron, OH, USA, 1-8, 2008.

[5] C. Kijeung , T. G. Robertazzi, Divisible Load Scheduling in Wireless Sensor Networks with Information Utility Performance. *In Proceedings of IPCCC 2008*, Austin, Texas, USA, 9-17, 2008.

[6] Z. ZENG, A. LIU, D. LI, A Highly Efficient DAG Task Scheduling Algorithm for Wireless Sensor Networks, *In Proceedings of ICYCS 2008*,Zhang Jia Jie , Hunan , China, 570-575, 2008.

[7] J. LIN, W. XIAO, F. L. Lewis, Energy-Efficient Distributed Adaptive Multisensor Scheduling for Target Tracking in Wireless Sensor Networks. *IEEE Transactions on Instrumentation and Measurement*, Vol.58, No.6, pp.1886 - 1896, 2009.

[8] Y. Yang, R. Van, Der, K, H. Casanova, Multiround Algorithms for Scheduling Divisible Loads. *IEEE Trans on Parallel and Distributed Systems*, Vol.16, No.11, pp.1092-1102, 2005.

[9] C. Yeim-Kuan, W. Jia-Hwa, C. Chi-Yeh, C. Chih-Ping, Improved Methods for Divisible Load Distribution on k-Dimensional Meshes Using Multi-Installment. *IEEE Transactions on Parallel and Distributed Systems*, Vol.18, No.11, pp. 1618-1629,2007.

[10] W. Heinzelman, A. Chandrakasan, An Application-specifid Protocol Architecture for Wireless Microsensor Networks. *IEEE Transaction on Wireless Communications*, Vol.1, No.4, pp. 660-670, 2002.

[11] X. Tang, J. Xu, Optimizing Lifetime for Continuous Data Aggregation With Precision Guarantees in Wireless Sensor Networks. *IEEE/ACM Transactions on Networking*, Vol.16, No.4, pp. 904 - 917,2008.