Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844
Vol. VI (2011), No. 4 (December), pp. 592-602

# A Non-cooperative Game Algorithm for Task Scheduling in Wireless Sensor Networks

L. Dai, Y.L. Chang, Z. Shen

**Liang Dai**
School of Electronic and Control Engineering
Chang'an University
Xi'an 710064, China
E-mail: ldai1981@gmail.com

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

**Abstract:** Scheduling tasks in wireless sensor networks is one of the most challenging problems. Sensing tasks should be allocated and processed among sensors in minimum times, so that users can draw prompt and effective conclusions through analyzing sensed data. Furthermore, finishing sensing task faster will benefit energy saving, which is critical in system design of wireless sensor networks. But sensors may refuse to take pains to carry out the tasks due to the limited energy. To solve the potentially selfish problem of the sensors, a non-cooperative game algorithm (NGTSA) for task scheduling in wireless sensor networks is proposed. In the proposed algorithm, according to the divisible load theory, the tasks are distributed reasonably to every node from SINK based on the processing capability and communication capability. By removing the performance degradation caused by communications interference and idle, the reduced task completion time and the improved network resource utilization are achieved. Strategyproof mechanism which provide incentives to the sensors to obey the prescribed algorithms, and to truthfully report their parameters, leading to an effient task scheduling and execution. A utility function related with the total task completion time and tasks allocating scheme is designed. The Nash equilibrium of the game algorithm is proved. The simulation results show that with the mechanism in the algorithm, selfish nodes can be forced to report their true processing capability and endeavor to participate in the measurement, thereby the total time for accomplishing the task is minimized and the energy-consuming of the nodes is balanced.

**Keywords:** wireless sensor networks; task scheduling; divisible load theory; game algorithm; mechanism design.

## 1 Introduction & Motivation

Wireless sensor networks are constituted by a large number of nodes, and without fixed infrastructure and centralized control mechanisms, so the task scheduling needs of mutual cooperation in all nodes. The energy of sensors is limited, so not all the sensors are willing to participate in collaborative efforts. Lack of coordination between the nodes in wireless sensor networks is a new problem[1].

Owing to the sensors with limited energy, the task should be completed within the shortest possible amount of time, 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. Divisible load theory [2] provides an effective solution to wireless sensor networks for task scheduling [3-6]. Different from other heuristic solutions of task scheduling problem in wireless sensor networks [7, 8], 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 scheduling algorithms were applied to wireless sensor networks in [3-6]. 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 [9]. Those algorithms mentioned above designed on the premise that sensors manipulate the algorithms and perform a task to the best of their abilities. But the sensors may be owned by autonomous rational organizations or people that have no a priori motivation for cooperation. Consequently, they will manipulate the algorithms if it benefits them to do so.

Therefore, we present a non-cooperative game algorithm for task scheduling in 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, and through the proposed mechanism to encourage collaboration.

## 2 Mechanism design

The energy of sensors is limited, if we do not integrate contributions with benefits, not all nodes willing to participate in collaborative efforts to perform tasks. This selfish, non-cooperative behavior will result in poor performance. If we follow the principle of more labor and more benefits, some selfish sensors, driven by the interests, will intentionally exaggerate its data-measuring capacity to increase their income. Such dishonest behavior will result in that it's difficult to determine which sensor reporting its actual capacity, and which said a lie, so we can not globally optimize tasks allocation according to each sensor's capacity. These dishonest or selfish behavior, has seriously affected the efficiency of wireless sensor network. Therefore, we should provide incentives to the nodes to obey the prescribed algorithms and to truthfully report their parameters, leading to an efficient allocation and execution. This problem can be solved by applying mechanism design of game theory.

Mechanisms are generally composed of allocation $o = o(a_1, a_2, \cdots, a_n)$ and payment $p = (p_1(a), p_2(a), \cdots, p_n(a))$ rules. Allocation rules also are called output function. In mechanism design, there are usually $n$ agents, and each agent has some private information, which is known as the type of agent or true value. The private information is only known by agent, and is confidential for the external, for example, the type of a agent $t_i$ can be the cost executing an assigned task. $v_i(t_i, o)$ is the value function of an agent, said the cost of performing a task. $p_i(\cdot)$ is the payment function for agent performing a task. $u_i(\cdot)$ is the utility function of an agent, that $u_i(\cdot) = p_i(\cdot) - v_i(t_i, o)$. To maximize the utility function value is the goal of agents. Mechanism should be designed to maintain unity with the goal the agent seeking for. Mechanism can induce the agent's choice or behavior by means of payment.

If in a mechanism, the only optional behavior of all agents is directly stating their preferences, then we say that the mechanism is a direct revelation mechanism. In a direct revelation mechanism, if all agents will tell their true preferences in order to maximize their utility, then we say that the mechanism is incentive compatibility. The nature of incentive compatibility allows

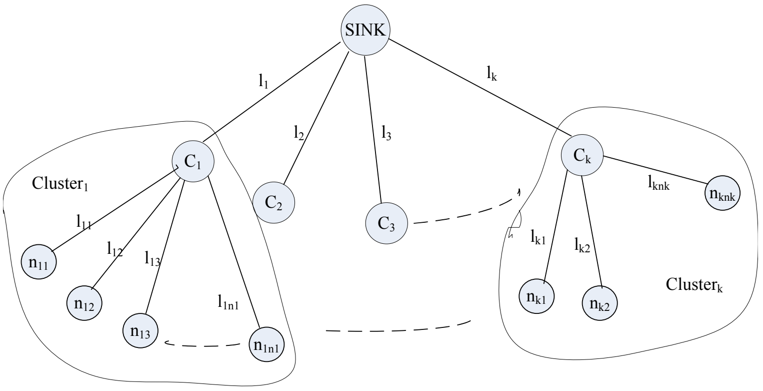

Figure 1: Network topology

us to design a mechanism to overcome the selfishness of agents, because in an incentive compatible mechanism, the agent will always choose to say their true private information in order to maximize their utility. In a direct revelation mechanism, if it's a dominant strategy that they gave their true preferences, then we claimed that the mechanism is *strategyproof*. If each agent is honest, and each agent's income is not less than zero, then the mechanism satisfies the voluntary participation condition.

# 3 Scheduling Algorithm

## 3.1 Network model

The network topology discussed in this paper is shown in Fig.1. Wireless sensor networks construct clusters several times in its life cycle. Each cluster will have a set-up phase and a steady-state phase[11]. Assuming that there are $k$ clusters ($Cluster_i$, $i = 1, ...., k$) in the network during a steady-state phase. Each cluster head is expressed as $C_i$. Within the $Cluster_i$, there are $n_i$ nodes expressed as $n_{i,j}$ ($i = 1, ....k; j = 1, ...n_i$), respectively. Communication links between cluster heads and SINK are expressed as $l_i$, $i = 1, ...., k$, respectively. Communication links between intra-cluster nodes and cluster head $C_i$ are expressed as $l_{i,j}$ ($i = 1, ....k; j = 1, ...n_i$), respectively.

The following notations will be used throughout this paper:

$\alpha_i$: The total fraction of load that is assigned by SINK to cluster head $C_i$;

$\alpha_{i,j}$: The fraction of load that is assigned to intra-cluster node $n_{i,j}$ in $Cluster_i$ by $C_i$;

By definition we can see:

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

$$\sum_{j=1}^{n_i} \alpha_{i,j} = \alpha_i \tag{2}$$

$\omega_i$: A constant that is inversely proportional to the processing (data fusion) speed of cluster head $C_i$;

$y_{i,j}$: A constant that is inversely proportional to the measuring speed of intra-cluster node $n_{i,j}$;

$z_i$: A constant that is inversely proportional to the speed of link between SINK and cluster head $C_i$;

$z_{i,j}$: A constant that is inversely proportional to the speed of link between the cluster head $C_i$ and the intra-cluster node $n_{i,j}$;

$T_{cp}$: Data fusion intensity constant. $T_{cp}$ is the time that takes to fuse all the load on a cluster head when $\omega_i = 1$. The entire load can be fused on cluster head $C_i$ in time $\omega_i T_{cp}$;

$T_{ms}$: Measurement intensity constant. $T_{ms}$ is the time that takes the intra-cluster node $n_{i,j}$ to measure the entire load when $y_{i,j} = 1$. The entire assigned measurement load can be measured on the intra-cluster node $n_{i,j}$ in time $y_{i,j}T_{ms}$;

$T_{cm}$: Communication intensity constant. This is the time it takes to transmit all the processing loads over a link when $z_i = 1$. The entire load can be transmitted over the $i$th link in time $z_i T_{cm}$;

$\varphi_i$: The information utility constant [8,10] of cluster head $C_i$. 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.

$T_f$: The finish time of a given task.

In this paper, we formulate the problem of task scheduling in a large scale sensor network as an optimization problem to minimize $T_f$.

### 3.2 Scheduling strategy

The proposed task scheduling algorithm is based on LEACH [11] protocol. LEACH was firstly proposed clustering protocol in wireless sensor network, and its certain governing ideas breathe through many other clustering protocols, such as TEEN and DAEA. LEACH protocol structures the single-hop networks. Each cluster's intra-cluster nodes share the same channel, while each cluster head has its own channel to SINK.

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 are assigned to each intra-cluster node by intra-cluster task scheduling.

**1) Intra-cluster task scheduling**

Fig.2 illustrates the timing diagram for a set of intra-cluster nodes, indexed from $n_1$ to $n_k$, in one cluster. From Fig.2, 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 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 makespan by scheduling the measuring time and reporting time of each node. Moreover, since the intra-cluster scheduling tries to avoid the transmission conflicts at the cluster head, energy spent on retransmission is conserved.

For cluster $i$, based on the timing diagram shown in Fig.2, one can write the following set of equations about tasks assigned to the intra-cluster nodes in $Cluster_i$:

$$\alpha_{i,j-1}y_{i,j-1}T_{ms} = \alpha_{i,j}y_{i,j}T_{ms} + \alpha_{i,j}z_{i,j}T_{cm}, i = 2, 3, ...k \tag{3}$$

A general expression for the above set of recursive equations can be written as:

$$\alpha_{i,j}{=}s_{i,j}\alpha_{i,j-1} \tag{4}$$

where $s_{i,j} = y_{i,j-1}T_{ms}/(y_{i,j}T_{ms} + z_{i,j}T_{cm})$.

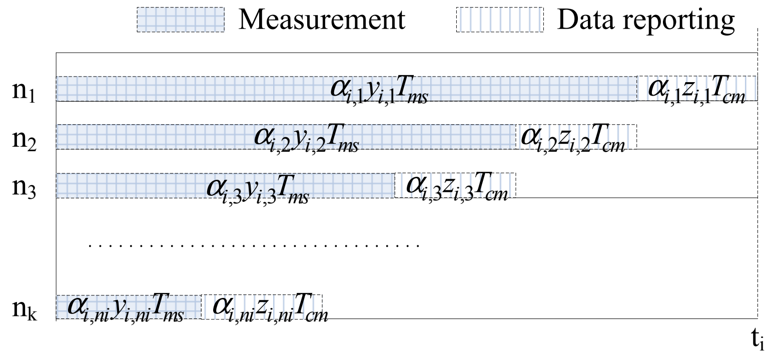
Measurement
Data reporting
n1
n2
n3
nk
$\alpha_{i,1}y_{i,1}T_{ms}$
$\alpha_{i,1}z_{i,1}T_{cm}$
$\alpha_{i,2}y_{i,2}T_{ms}$
$\alpha_{i,2}z_{i,2}T_{cm}$
$\alpha_{i,3}y_{i,3}T_{ms}$
$\alpha_{i,3}z_{i,3}T_{cm}$
$\alpha_{i,ni}y_{i,ni}T_{ms}$
$\alpha_{i,ni}z_{i,ni}T_{cm}$
ti

Figure 2: Timing diagram for intra-cluster task-processing

Using eq.2 and eq.4, The above recursive equation for $\alpha_{i,1}$ can be rewritten in terms of $\alpha_i$ only as

$$\alpha_{i,1} = \alpha_i/(1+\sum_{j=2}^{n_i}\prod_{k=2}^{j} s_{i,k}) \tag{5}$$

The cluster head $C_i$ will use the above value of $\alpha_{i,1}$ to obtain the amount of data that has to be measured by the rest of the $n_i - 1$ sensors by using

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

Then from Fig.2, the minimum time for measuring, reporting SINK's sub-task $\alpha_i$ can be given as

$$t_i{=}\alpha_i(y_{i,1}T_{ms} + z_{i,1}T_{cm})/(1+\sum_{j=2}^{n_i}\prod_{k=2}^{j} s_{i,k}) \tag{7}$$

**2) Inter-cluster task scheduling**

After cluster heads fusing the intra-cluster nodes' measured data, cluster heads can send the fused data to SINK concurrently because each cluster head has a separate channel to SINK.

In order to remove the performance degradation caused by idle, and to improve efficiency, as shown in Fig.3, we can get

$$t_i + \alpha_i w_i T_{cp} + \varphi_i \alpha_i z_i T_{cm} = t_j + \alpha_j w_j T_{cp} + \varphi_j \alpha_j z_j T_{cm}, i \neq j \tag{8}$$

In eq.7, we make $(y_{i,1}T_{ms} + z_{i,1}T_{cm})/(1+\sum\limits_{j=2}^{n_i}\prod\limits_{k=2}^{j} s_{i,k}) = g_i$, then substitute $g_i$ into eq.8, we can get

$$\alpha_i r_i = \alpha_j r_j \tag{9}$$

where $r_i = g_i + w_i T_{cp} + \varphi_i z_i T_{cm}$.

Now using the eq.1 and eq.9, one can solve for $\alpha_i$ as

$$\alpha_i{=}(1/r_i)/\sum_{i=1}^{k}(1/r_i) \tag{10}$$

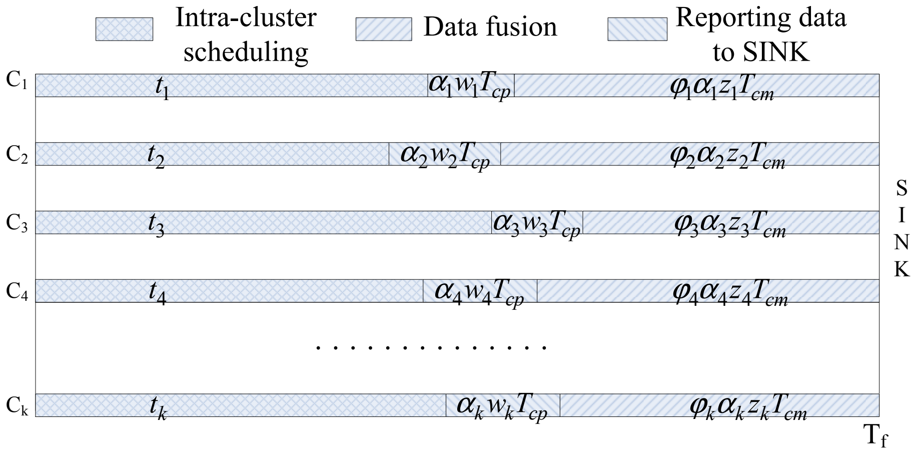

Figure 3: Timing diagram for inter-cluster task scheduling

Hereto, we can get the fraction of load cluster head $C_i$ and the intra-cluster nodes within it received from SINK ($\alpha_i$ and $\alpha_{i,j}$).

And the total task execution time is as follow:

$$T_f{=}t_i + (w_i T_{cp} + \varphi_i z_i T_{cm})(1/r_i)/\sum_{i=1}^{k}(1/r_i) \tag{11}$$

For the homogeneous networks, the parameters(measurement speed, communication speed, data-fusing speed, information utility constant) of the network are the same and there are $n$ sensors in each cluster. SINK can evenly distributed loads to each cluster. The sub-loads each cluster receiving is as follows:

$$\alpha_i{=}1/k \tag{12}$$

In the homogeneous networks, the minimum time-consuming for the total task is:

$$T_f{=}(D + wT_{cp} + \varphi z T_{cm})/k \tag{13}$$

where $D{=}\ (yT_{ms} + zT_{cm})/n$.

With the increasing density of wireless sensor networks, the number of the clusters is higher with the current clustering algorithms. From eq.13, it can be known that $T_f$ approaches 0 with $k$ approaching infinite.

## 4 Mechanism Design for NGTSA

In this section, we propose strategyproof mechanism for task scheduling of wireless sensor networks. The $i$th cluster is composed of $n_i$ strategic sensors. Each task-performing sensor $n_{i,j}$ is characterized its true value $y_{i,j}$, which is equal to the time to measure the unit task. We denote by $\mathbf{y}_i = (y_{i,1}, y_{i,2}, \cdots, y_{i,n_i})$ the vector of true measuring speed. Measuring speed $y_{i,j}$ is private to $n_{i,j}$. As the sensor $n_{i,j}$ with rationality may give a bid $b_{i,j}$ which is not equal the true measuring speed $y_{i,j}$, if it benefits it to do so.

Cluster heads calculated the fraction of load that is assigned to intra-cluster nodes $\alpha_i(\mathbf{b}_i) = (\alpha_{i,1}(\mathbf{b}_i), \alpha_{i,2}(\mathbf{b}_i), \cdots, \alpha_{i,n_i}(\mathbf{b}_i))$, according to the bids of intra-cluster nodes, where $\sum\limits_{j=1}^{n_i} \alpha_{i,j}(\mathbf{b}_i) = \alpha_i$, and $\mathbf{b}_i = (b_{i,1}, b_{i,2}, \cdots, b_{i,n_i})$ is the bid vector of intra-cluster nodes.

As the selfishness of sensors, the sensor may complete the measurement task at a relatively low speed $\tilde{y}_{i,j}$, where $\tilde{y}_{i,j} \leq y_{i,j}$. After intra-cluster nodes completed their task, cluster head node will get the actual measuring speed.

The value function of $n_{i,j}$ under task allocation $\alpha_i(\mathbf{b}_i)$ is defined as follow:

$$V_{i,j}(\alpha_i(\mathbf{b}_i), \tilde{y}_{i,j}) = -\alpha_{i,j}\tilde{y}_{i,j} \tag{14}$$

The value of $V_{i,j}(\alpha_i(\mathbf{b}_i), \tilde{y}_{i,j})$ is equivalent to the negation of the actual time required for $n_{i,j}$ to complete $\alpha_{i,j}$ of total load. It can be regarded as the cost of $n_{i,j}$ to complete its task.

The mechanism will provide payment for involving in measurement, so each node will select the appropriate strategy to maximize its benefits. The utility function of $n_{i,j}$ is defined as follow:

$$U_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) = Q_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) + V_{i,j}(\alpha_i(\mathbf{b}_i), \tilde{y}_{i,j}) \tag{15}$$

where $Q_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i)$ is the payment cluster head $C_i$ paid to $n_{i,j}$, $\tilde{\mathbf{y}}_i = (\tilde{y}_{i,1}, \tilde{y}_{i,2}, \cdots, \tilde{y}_{i,n_i})$ is the vector of each node's actual measurement speed. Utility function equals the payment the node got from mechanism minus the cost for completing its tasks.

Designing such mechanisms involves finding an allocation and a payment scheme that minimize the makespan according to the sensors' bid $\mathbf{b}_i$ and motivate all the intra-cluster sensors to bid their true value $y_{i,j}$ and measure the task at their full measuring capacity $\tilde{y}_{i,j} = y_{i,j}$.

Define the payment function for mechanism is as follow:

$$Q_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) = C_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) + B_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) \tag{16}$$

where $C_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) = -V_{i,j}(\alpha_i(\mathbf{b}_i), \tilde{y}_{i,j})$ is the compensation to the node $n_{i,j}$.

The bonus of $n_{i,j}$ is defined as follow:

$$B_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) = T_{-i,j}(\alpha_i(\mathbf{b}_{-i,j}), \mathbf{b}_{-i,j}) - T(\alpha_i(\mathbf{b}_i), (\mathbf{b}_{-i,j}, \tilde{y}_{i,j})) \tag{17}$$

where $T_{-i,j}(\alpha(\mathbf{b}_{-i,j}), \mathbf{b}_{-i,j})$ is the optimal task completion time when $n_{i,j}$ is not involved in measurement, that is, the bonus of intra-cluster node is equal to its contribution in reducing the task completion time because of its participation for measurement.

**Theorem (strategyproofness)**. The mechanism proposed in this paper is *strategyproof.*

**Proof.** According to the bids of intra-cluster nodes in cluster $i$, $\mathbf{b}_i = (b_{i,1}, b_{i,2}, \cdots, b_{i,n_i})$, we can get the utility of $n_{i,j}$ is as follow:

$$\begin{aligned} U_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) &= Q_{i,j}(\mathbf{b}_i, \tilde{\mathbf{y}}_i) + V_{i,j}(\alpha_i(\mathbf{b}_i), \tilde{y}_{i,j}) \\ &= T_{-i,j}(\alpha(\mathbf{b}_{-i,j}), \mathbf{b}_{-i,j}) - T(\alpha_i(\mathbf{b}_i), (\mathbf{b}_{-i,j}, \tilde{y}_{i,j})) + \alpha_{i,j}(\mathbf{b})\tilde{y}_{i,j} - \alpha_{i,j}(\mathbf{b})\tilde{y}_{i,j} \\ &= T_{-i,j}(\alpha(\mathbf{b}_{-i,j}), \mathbf{b}_{-i,j}) - T(\alpha_i(\mathbf{b}_i), (\mathbf{b}_{-i,j}, \tilde{y}_{i,j})) \end{aligned} \tag{18}$$

When $n_{i,j}$ perform a task to the best of their abilities ($\tilde{y}_{i,j} = y_{i,j}$), and bid the measurement as $b_{i,j} = y_{i,j}$, the utility of $n_{i,j}$ is as follow:

$$\begin{aligned} U^t_{i,j} &= T_{-i,j}(\alpha(\mathbf{b}_{-i,j}), \mathbf{b}_{-i,j}) - T(\alpha_i(\mathbf{b}_i), (\mathbf{b}_{-i,j}, \tilde{y}_{i,j})) \\ &= T_{-i,j}(\alpha(\mathbf{b}_{-i,j}), \mathbf{b}_{-i,j}) - T^t_{i,j} \end{aligned} \tag{19}$$

where $T^t_{i,j}$ is the shortest possible amount of time to finish total tasks according to divisible load theory. If $n_{i,j}$ bids a lower value, that is, $b_{i,j} < y_{i,j}$ ,the makespan is increased by a small amount as the load allocated to the other intra-cluster nodes is increased. Therefore, only when $n_{i,j}$ voluntarily gives the true measuring speed and be assiduous in its duty, they can get the maximum benefit.

Because $T_{-i,j}$ is the optimal task completion time when $n_{i,j}$ is not involved in measurement, its value must not be less than when all the nodes involved in the tasks. So, we can get $U^t_{i,j} \geq 0$, that is, all nodes within one cluster will try to do their best to carry out a task.

# 5 Wireless Energy Use

In this section, the energy model of the NGTSA 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 [11].

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:

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

# 6 Performance Evaluation

In the above sections, we have obtained the close-form solution of the task completion time and the formula for the energy consumption of each node. In this section, we investigate the effects of different cases under homogeneous network environment on the makespan, energy consumption of every intra-cluster nodes, and the payment/utility of the cheating nodes.

We consider the wireless sensor network comprising 10 clusters. Within each cluster, there are 20 intra-cluster nodes. Assume that only the first intra-cluster node cheats by reporting values different than its true measuring speed, and by measurement at a different speed than the true speed in each cluster. The speed of communication link between each node were 0.05; the measuring speed of each intra-cluster node is 0.2; the data fusion speed of each cluster head is 0.1, and the information utility constant of each cluster head is 0.5.

In the simulation, the following energy parameters are adopted: transmitting a unit of sensor reading over a unit distance takes $e_{tx}$=200$nJ$, receiving one unit of sensor reading consumes $e_{rx}$=150$nJ$, measuring one unit of sensor reading needs $e_s$=100$nJ$, fusing one unit of observation consumes $e_p$=20$nJ$ and the distance between the sender and the receiver is $d$=100$m$. In the simulation, we supposed that: $T_{ms}$=$T_{cm}$=$T_{cp}$= 1.

We simulated the following eight cases: ① $y_{i,1} = b_{i,1} = \tilde{y}_{i,1}$; ② $y_{i,1} = b_{i,1} < \tilde{y}_{i,1}$; ③ $y_{i,1} < b_{i,1} = \tilde{y}_{i,1}$; ④ $y_{i,1} = \tilde{y}_{i,1} < b_{i,1}$; ⑤ $b_{i,1} < y_{i,1} = \tilde{y}_{i,1}$; ⑥ $y_{i,1} < \tilde{y}_{i,1} < b_{i,1}$; ⑦ $y_{i,1} < b_{i,1} < \tilde{y}_{i,1}$; ⑧ $b_{i,1} < y_{i,1} < \tilde{y}_{i,1}$. where $b_{i,1}$ states the bid of the first intra-cluster node; $y_{i,1}$ represents the node's actual value; $\tilde{y}_{i,1}$ is the true value of the measurement speed.

The bid $b_{i,1}$ and the measurement value $\tilde{y}_{i,1}$ for each case are presented in Tables 1:

Table 1: Bids and Execution Values

| Case | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $b_{i,1}$ | 0.2 | 0.2 | 0.3 | 0.3 | 0.1 | 0.4 | 0.3 | 0.1 |
| $\tilde{y}_{i,1}$ | 0.2 | 0.3 | 0.3 | 0.2 | 0.2 | 0.3 | 0.4 | 0.3 |

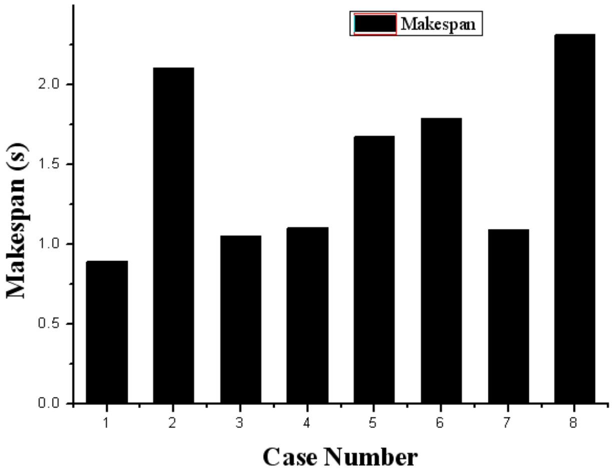

Figure 4: Makespan when the first intra-cluster node cheats

The simulation results are shown in Fig.3 and Fig.5.

Firstly, Fig.3 shows the makespan for the eight cases. Notice that case 1 ($y_{i,1} = b_{i,1} = \tilde{y}_{i,1}$) results in the minimum makespan, while all other cases result in a larger makespan. In that case, each node honestly bids, and makes effort to measure. When $\tilde{y}_{i,1} \leq b_{i,1}$, the case 3,4, and 7, the makespan is increased by a small amount as the load allocated to the other intra-cluster nodes is increased. The effect of cheating is dispersed throughout the remaining nodes. In the remaining case ($\tilde{y}_{i,1} > b_{i,1}$),the case 2,5,6, and 8, the network performance dramatically degrades as the first intra-cluster nodes is overloaded, and the other nodes are underutilized. In these cases, the first intra-cluster nodes is slowing down the entire network. The increase in makespan is large, and it is due to the impact of $\tilde{y}_{i,1}$.

Next, the second simulation is about the energy consumption of intra-cluster sensor nodes for the 8 cases. 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. In each case, the energy consumption of sensor nodes monotonically decreases due to the reduced workload. In the case ($y_{i,1} = b_{i,1} = \tilde{y}_{i,1}$), each node honestly bids, and make effort to measure, so the energy-consuming of each intra-cluster node is most evenly. In the case 2,5,6, and 8, because the cluster head allocates more tasks on the first intra-cluster node as it bids a rate slower than its true rate, the energy-consuming of the first intra-cluster nodes increases noticeably, which will run out power and affected the entire network performance.

Then, the third simulation is about the payment and utility of intra-cluster nodes. Fig.6 shows the payment and utility for the first intra-cluster nodes in the 8 cases. As we expected, case 1 ($y_{i,1} = b_{i,1} = \tilde{y}_{i,1}$), maximizes the first intra-cluster node's utility. In all the other cases, utility is lessened. When $\tilde{y}_{i,1} > b_{i,1}$, the utility is negative due to the impact of $\tilde{y}_{i,1}$ on the

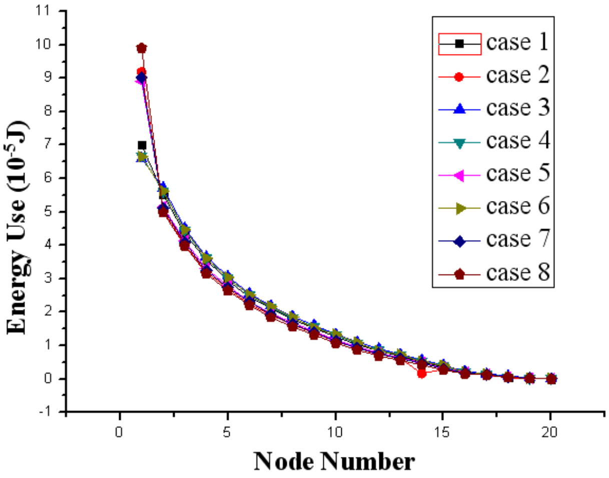

Figure 5: Energy consumption when the first intra-cluster node cheats

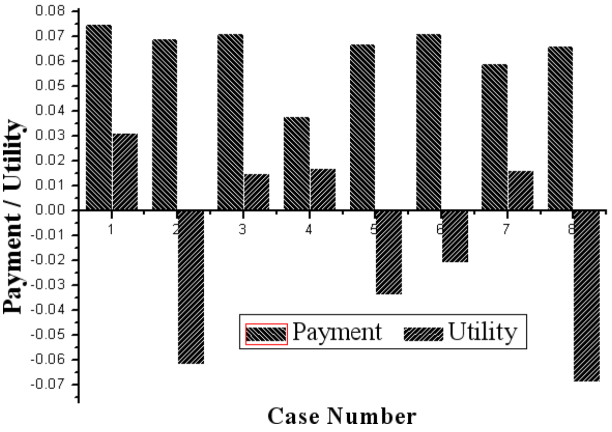

Figure 6: Payment and utility of the first intra-cluster node when it cheats

makespan. In the remaining cases, payment and utility are reduced as $\alpha_{i,1}$.

## 7 Conclusions

As the node in wireless sensor network has limited energy, the tasks should be completed as quickly as possible, and the network resources should be fully utilized. In this paper, we present a task scheduling algorithm (NGTSA) 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.

As the gradual development of the wireless sensor networks and ubiquitous networks, this scheduling model using divisible load theory needs further study and validation. In addition, the validation of divisible load theory in wireless sensor networks still remains in the simulation phase. There are short of scanty of large-scale, practical applications to demonstrate its true potential, so applying the divisible load theory in actual wireless sensor network should be an important research direction for the next step.

## Acknowledgement

The authors thank the editors and the anonymous reviewers for their valuable comments that helped to improve the paper. The work was supported by the National Natural Science Foundation of China (No.60972047), and the 111 project (No.B08038).

## Bibliography

[1] C. Pandana, H. Zhu, L. Ray, Cooperation Enforcement and Learning for Optimizing Packet Forwarding in Autonomous Wireless Networks. *IEEE Transactions on Wireless Communications*, vol.7, No.8, pp.3150-3163, 2008.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] P. Guo, T. Jiang, K. Zhang, H. Chen, Clustering algorithm in initialization of multi-hop wireless sensor networks. *IEEE Transactions on Wireless Communications*, vol.8, No.12, pp. 5713–5717, 2009.

[10] N. Nisan, A. Ronen. Algorithmic mechanism design, *Games and Economic Behavior*, vol.35, nos.1-2, pp.166 -196, 2001.

[11] 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.