Paper Title\* **(**use style: ***paper title***)

Subtitle as needed ***(paper subtitle)***

Authors Name/s per 1st Affiliation (*Author*)

line 1 (of *Affiliation*): dept. name of organization

line 2-name of organization, acronyms acceptable

line 3-City, Country

line 4-e-mail address if desired

Authors Name/s per 2nd Affiliation (*Author*)

line 1 (of *Affiliation*): dept. name of organization

line 2-name of organization, acronyms acceptable

line 3-City, Country

line 4-e-mail address if desired

*Abstract*—Battery recovery effect is a phenomenon that available capacity of a battery could increase if the battery can sleep for a period of time since its discharge last time. Accordingly, the battery can work for a longer time when it takes some rest between consecutive discharging processes than when it works all the time. However, this effect has not been considered in the design of existing topology management algorithms for wireless sensor networks. In this paper, we propose a distributed battery recovery effect aware connected dominating set constructing algorithm. In this algorithm, each node in the network periodically decides to be in the dominating set or not. Nodes that have taken sleep in the preceding period are encouraged to involve in dominating set in the current period while nodes that have worked in the preceding period are encouraged to sleep in the current round for battery recovery. The complexity of the proposed algorithm is derived to be *O*(*D*3), where *D* represents node degree. Simulation results show that our algorithm outperforms existing work.

Keywords—Battery recovery effect; connected dominating set; wireless sensor networks; network lifetime

# Introduction

Battery recovery effect is a phenomenon that the available capacity of a battery could increase if the battery can sleep for a certain period of time since its discharge last time. This is because sensor nodes in a wireless sensor network (WSN) are usually powered by batteries. When the battery of a sensor node runs out of energy, the node cannot work any more. Battery recovery effect has received much attention and much work has been carried out to build accurate battery model [x, x] or figure out how to take advantage of such battery recovery effect in the design of network protocols for WSNs [x] or other battery-related applications [].

Existing work in the area of battery recovery effect can be divided into two aspects: Some focused on how to build accurate battery model to characterize the battery recovery effect and some others focus how to incorporate such battery recovery effect into the design of a network protocol for WSNs or other battery-related applications. Regarding building accurate battery models, based on the authors’ knowledge on battery models, they can be divided into four categories: electrochemical models, kinetic battery models, electrical circuit models and stochastic battery models. As for how to incorporate such battery recovery effect into the design of a network protocol for WSNs or other battery-related applications, there exists many related works. Fu proposed an efficient algorithms for maximum lifetime coverage problem incorporating battery recovery effect in [1] and results show the network lifetime of his algorithm is 10-40% longer than the network lifetime of an algorithm without battery recovery effect. In [], Rakhmatov presented several algorithms for task ordering and voltage assignment in embedded systems, meanwhile, they insert some idle periods to take advantage of battery recovery effect. Chau has done a series of experiments about the extraordinary significance of battery recovery effect, and experimental results show the gains of battery recovery effect can extend sensor nodes’ network life by up to 45% [].Chau presented a distributed duty cycle scheme incorporating battery recovery effect by pseudo-random sequence in []. However, to the best of our knowledge, no existing work has considered how to incorporate battery recovery effect into the design of a topology control algorithm in WSNs.

In this paper, we present a distributed battery-aware topology management algorithm incorporating battery recovery effect. For this purpose, we first abstract a battery recovery model based on the experimental results reported in [x]. Based on this model, we propose a distributed battery recovery effect aware connected dominating set constructing algorithm. In this algorithm, each node in the network periodically decides to be in the dominating set or not. Nodes that have taken sleep in the preceding period are encouraged to involve in dominating set in the current period while nodes that have worked in the preceding period are encouraged to sleep in the current round for battery recovery. Detailed design description is given. The complexity of the proposed algorithm is derived to be *O*(*D*3), where *D* represents node degree. Simulation results show that our algorithm can significantly improve network lifetime performance as compared with existing work.

The rest of the paper is arranged as follows. Section 2 gives a brief review of related work of battery recovery model and topology management algorithms. In Section 3, we propose our algorithm. We first present an equivalent battery model based on the experimental results in [], and then present detailed design description of our proposed algorithm. Performance evaluation is carried out in Section 4. In Section 5, we conclude this paper.

# related work

In this section, we will first introduce related work in battery model and then introduce typical work in topology management.

## Battery Model

If we want to take advantage of battery recovery effect in topology management, we have to choose a battery model carefully for sensor nodes. Researchers have done much work to design a battery model that can characterize the battery recovery effect. Based on the authors’ knowledge on battery models, they can be divided into four categories: electrochemical models, kinetic battery models, electrical circuit models and stochastic battery models.

Electrochemical models are the most accurate kind of battery models, and they [] use non-linear differential equations to describe the detail of electrochemical reactions within battery. This kind of models required a full knowledge of the battery and it takes days to predict battery lifetime.

~~Kinetic battery models [], [], [] also have high time complexity so that they can’t be incorporated into sensor node.~~

KiBaM [] is a classical Kinetic battery model. Compared with electrochemical models, KiBaM uses equations of reduced complexity to describe the kinetic process of the electrochemical reactions within the batteries. Both electrochemical models and kinetic battery models have a high time complexity and require large memory size that are not suitable for sensor node.

Electrical circuit models [], [] abstract the physical phenomena inside batteries using equivalent electrical circuit. It can’t be applied into simulation works by software, either.

Stochastic battery models [], [] characterize battery behavior by mathematical model. ~~This type of models is the most suitable one for WSNs among these four kinds of battery models.~~ Panigrahi proposed a stochastic battery model incorporating both battery recovery effect and rate capacity effect in []. Rate capacity effect is another key battery effect, but we won’t discuss it in this paper. In this model, the state of battery is represented by state of charge, which is united by the smallest amount of capacity that could be discharged. But this model needs accurate current load profile and can’t be implemented in a sensor node with the requirement of complex computations and large memory size. ~~The authors of [] provides a recursive version of this model~~

Chau proposed a simplified Markov chain model, and presented the definition of saturation threshold for the first time in []. Saturation threshold means available battery capacity will recover less when sleep time exceeds it. Current battery state in their model is represented by, where is the nominal battery capacity, is the theoretical battery capacity, t is the number of time slots since last discharging. But this model use deterministic battery recovery. It may works for fixed duty cycle, but not for stochastic duty cycle.

## Topology management

Topology management has been studied sufficiently. There are two main categories of topology management: power control and hierarchical topology organization []. Since the radio is always on in power control management, we won’t discuss this kind of topology management in this paper. Hierarchical topology organization, also known as clustering algorithm, can be divided into three categories: probability-based, location-based, and Connected Dominating Set-based (CDS).

LEACH is the famous probability-based algorithm, and executes periodically. Each period is composed of cluster head selection phase and data transition phase. Each node decides whether to be cluster head by certain probability on its own or not.

GAF selects its cluster head nodes according to their location. It divides the networks into grids at first phase, and selects cluster head in each grid.

CDS-based algorithm is based on the concept of dominating set in graph theory. Nodes in CDS take the role of cluster heads. TopDisc is a classical algorithm that uses color to mark cluster head nodes and non-cluster head nodes []. Wu proposed a power-aware connected dominating set-based topology management.

However, all the topology management discussed above did not consider battery recovery effect in their respective design and therefore can lead to degraded network lifetime performance.

1. Notation

| Symbols | Definitions |
| --- | --- |
|  | work current |
|  | sleep current |
| *RC(t)* | recovery current value when it has relaxed for t seconds |
|  | the i-th period |
| *NS(v,)* | *NS(v,)=T* means node v is in CDS in, while *F* means not in CDS. |
| *NBR(v)* | the one-hop neighbor set of node v |
| *NBR[v]* |  |
| *hasRelaxed(v,)* | *hasRelaxed(v,)=T* means node v has relaxed in, while *F* means not. |
| *E(v)* | energy level of node v |
| *DEG(v)* | node degree of node v |
| *ID(v)* | id of node v |
|  | the length of one time slot |

# Propsoed Algorithm

In this section, we present the design of our XXXX algorithm. We will first abstract a battery model based on the experimental results in [x]. This model can characterize time-variant recovery current. Then, we give the design of XXX, which incorporates the battery recovery effect based on the above battery model for energy efficient topology management. We derive the complexity of XXXXX to be O(X).

## Battery model

Inspired by the experimental results in [], we calculated the equivalent recovery current of sensor node during sleep phase. We found that the recovery current is not stable during sleep phase. In [], they use TelosB nodes with two AA NiMH 600mAh batteries in these experiments to get the gains of under different duty cycle, and the work time of each cycle is 10s while sleep time is different from 0s to 14s.

1. Fitted curve of sleep time per period and the gain of network lifetime compared with the always active case, the data comes from experiment result of [].



1. the time-variant recovery current during relax phase, labeled as RC(t)



Fig. 1 is the fitted curve of sleep time per period and the gain of network lifetime compared with the always active case, according to the experimental data in []. In Fig. 1 shows the lifetime of network with duty cycle rate equaling 83.33% is 5% longer than that of the network without duty cycle, but network lifetime doesn’t get longer as the duty cycle rate decreasing. Experimental result in [] shows battery lifetime is about 1155 min when it works all the time. The extra battery lifetime is provided by the recovery capacity during sleep phase, so we can calculate the equivalent recovery current during each second according to the curve in Fig. 1. For example, the battery runtime is about 1200 min when the duty cycle rate is 91%. So we regard the extra 45 min as the result of the recovery capacity of 1s each cycle. Fig. 2 shows the recovery current of during sleep phase and the recovery current is not stable .

We abstract an equivalent battery model from the data in Fig. 2 and datasheet of TelosB []. The model is like this.

During discharging process:

During relax process:

Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors).*

## Algorithm design

We present the detailed design of the distributed battery-aware topology management algorithm incorporating battery recovery effect. In this algorithm, each node in the network decides to be in the dominating set or not at the beginning of each period. The while connected dominating set construction is divided by three phase, both the first and the second phase aim to construct a CDS with redundant nodes, and the third phase aims to make some cuts in the CDS. Nodes that have taken sleep (slept)in the preceding period are encouraged to involve in (the)dominating set in the current period while nodes that have worked in the preceding period are encouraged to sleep in the current round for battery recovery. The detailed algorithm is showed as follows:

Step1: If node have slept in the last period, then it will make its own decision to join in the CDS temporally.

|  |  |
| --- | --- |
| 1: | ; |
| 2: |  |
| 3: | ; |
| 4: |  |
| 5: |  |

Step2: If node has two neighbors, say, node and node , and either node or node is not the neighbor of each other, then node  will also join in the CDS temporally. Specially, if a node has only one node, then it will join in the CDS, too.

|  |  |
| --- | --- |
| 1: |  |
|  |  |
| 2: | ; |
| 3: |  |

Step3：Notice that node , node and node in this step have value for. If node is redundant in the CDS, the node will exit from CDS. There are two rules for us to decide whether node is redundant or not.

: This rule is for the case that node can be replaced by its neighbor node . We say node can be replaced by node if each neighbor of node is also a neighbor of node , that is . The decision is made as follow.

|  |  |
| --- | --- |
| 1: |  |
|  |  |
| 2: | ; |
| 3: |  |
|  |  |
|  |  |
| 4: | ; |
| 5: |  |
| 6: |  |
| 7: |  |

: This rule is for the case that node can be replaced by its two neighbors node and node , that is .

|  |  |
| --- | --- |
| 1: |  |
|  |  |
| 2: |  |
|  |  |
| 3: | ; |
| 4: |  |
| 5: | ; |
| 6: |  |
|  |  |
| 7: |  |
|  |  |
| 8: | ; |
| 9: |  |
|  |  |
|  |  |
| 10: | ; |
| 11: |  |
|  |  |
| 12: |  |
|  |  |
| 13: | ; |
| 14: |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 15: |  |

Compared with the algorithm presented in [], our algorithm make the nodes who have relaxed in the last period have the privilege to be in CDS. So the node can avoid working or relaxing continuously, since work for too long time could lead to a sharp decrease in single node’s available battery capacity, while relaxing for too long time wouldn’t get more energy recovered.

## An example

Fig.3 shows an example using our （the）algorithm we proposed in this paper. Nodes that construct the CDS are colored as black, and the node that is outside of the CDS are colored as grey. The nodes that have slept in the last period is marked by diagonal stripes. The number inside the circle represents the id and energy of the node, for example, 3(5) means the node id is 3 and the energy left in the node is 5.

Fig.3 (a) is the initial state of network after the network is deployed. No node has relaxed before, so we go step 2 directly. Node 2 decides to be in CDS since its neighbors, node 3 and node 1 are not neighbors. Node 3 decides to be not in CDS, because its only neighbors node 2 and node 4 are each other’s neighbor. There we get Fig.3 (b). Fig.3 (c) show the results after applying rule 1 of step 3, node 1 exits from CDS, since it can be replaced by node 2 and got lower energy than node 2. The same rule can be applied to node 8. In Fig.3 (d), after applying rule 2 of step 3, node 5 existed from CDS, because it can be replaced by node 4 and node 7, and it has the lowest energy among the three nodes. Notice that node 4 can be replaced by node 2 and node 5, but it doesn’t exit from the CDS since its energy is higher than the energy of node 5. So far, the CDS is constructed, and all nodes in the CDS keep active, while the others go to sleep with probability .After a period of time, some nodes’ state (residual energy, has slept or not) are changed, just like what Fig.3 (e) shows. At the beginning of the next round, a new CDS needs to be constructed. In Fig.3 (f), by re-applying step 1, all nodes that has slept in last period are in CDS. In Fig.3 (g), node 2, node 4 and node 7 are all in CDS after step 2, so we can make sure the dominating set is connected. After applying rule 1 of step 3, node 3 exits in Fig. 3 (h). In Fig. 3 (i), node 4 instead of node 5 exits from the CDS as it has the lowest energy this time. Fig. 3(j) shows the network state after one period.

1. Examples for our algorithms



### (a)The initial network state (b)Network state after step 2



### (c)Network state after the rule1 (d)Network state after the rule 2



### (e)Network state after a period (f)Network state after step 1



### (g)Network state after step 2 (h)Network state after the rule 1



### (i)Network state after the rule2 (j)Network state after a period

# Performance evaluation

In this section, we will present the experiment results of our algorithm and highlight the extraordinary performance of battery recovery effect. For fair comparison, we also implement the algorithm in [] with battery recovery effect. The simulator is developed by C++.

In the simulations, we assume:

* All modules of sensor node is active when sensor node is active, while all is sleep in sleep mode.
* MAC protocol is ideal and thus no transmission conflicts/corruption can happen.
* The network selects the CDS at the beginning of each election period.
* Each node outside of the CDS decides to sleep or not also at the beginning of a time slot.
* There is no data transmission in the network.

### The current (current draw of RF Transceiver plus current draw of processor) is in active mode or in sleep mode, according to the datasheet of TelosB. We assume the current draw is stable during each mode. We choose 100 nodes that are randomly deployed in variant size field to get various node density in Fig. 4. The related parameters of each simulation are set at Table Ⅱ.

1. simulation parameterS Settings

| Parameters | Settings for Fig. 4 | Settings for Fig. 5 |
| --- | --- | --- |
| Simulation field (circle) radius | variable | 300 |
| Number of nodes | 100 | Variable |
| Communication range | 90 | 90 |
| Update duration | 7 | 7 |
| Node density | Variable | Variable |
| Probobility of relaxing | 0.5 | 0.5 |

### Fig. 4 and Fig. 5 show us that our algorithm gets a better performance than Wu’s algorithm as the node density or network size increasing. This highlights the significance of incorporating battery recovery effect into the battery related work.

1. Comparation of the network lifetime of these two algorithms



All the simulations above assume the update interval is the same all the time, and we choose 7s as the update interval. In Fig. 5, we measured the network lifetime of this network with different update interval, and experiment result shows the ideal update interval is different under different node density. The ideal update interval is 14s when the node density is 40, while 7s for 20. We choose 7s as the node density is more reasonable in practical situation.

1. Comparation of the network lifetime of different update duration



In Fig. 6, we measured the CDS size of these two algorithms. Obviously, our algorithm gets a larger CDS than Wu’s algorithm in []. This is because we can let more nodes work to keep connectivity instead of keeping few key nodes in work all the time. For example, there are only three nodes in the network, marked as node , node , node ,

1. Comparation of the CDS size of these two algorithms



# conclusion

Battery recovery effect is an important phenomenon of batteries. Incorporating battery recovery effect into battery-related work can prolong network lifetime effectively. In this paper, we first abstract a battery recovery model based on the experimental results reported in [x]. This battery model characterize the recovery current is time-variant during sleep phase. Then, we incorporate battery recovery effect into the design of topology control protocol. Experiment result show our algorithm can significantly improve network lifetime performance as compared with existing work. This highlights that battery recovery effect shouldn’t be ignored either in battery model design or related applications. But the battery model we abstract in this paper is restricted to the battery material (AA NiMH 600mA), and can’t be applied to other nodes powered by other batteries. So an analyzable simple common fitted battery model for sensor node is in very much need.

##### Acknowledgment

We would like to thank Yu Gong, Jia-Li Shuai for feeding me knowledge of battery. We are so grateful to Wei Gong, Zhi-Wei He, Guang-Lun Huang, Zhen-Zhen Jiao, Dong Li, Si-Dong Liu, De-Zhong Shang, Jin-Zhao Suo and Tao Wang for hot and fruitful discussion on battery behavior, topology management and reducing time complexity of algorithm.

##### References

1. G. Eason, B. Noble, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (*references*)
2. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
3. I.S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.