# Cluster-based Power Aware Scheduling (CPAS) Algorithm for Network Longevity in WSN

K. Arthi and P. Vanaja Ranjan

Department of Electrical Engineering, Guindy, Anna University, Chennai
arthimanivasakam@yahoo.co.in

**Abstract.** Efficient power utilization gains more importance for WSNs (Wireless sensor Network), since battery replacement is not possible in many sensor applications. Sensors consume energy when it changes from one radio state (transmission, reception, listen, sleep) to another. In this paper, a Cluster-based Power-Aware Scheduling (CPAS) algorithm is proposed to specifically design a low-data-rate WSNs to reduce the number of state transitions of a node, thereby efficiently maintaining the power level of the network. In CPAS, the nodes within the cluster are first synchronized to avoid collision during transmission. CPAS is based on IEEE 802.15.4 standard with dynamic routing ability based on the power level of the nodes. Performance evaluation is done by using simulation, and it has been showed that this cluster based algorithm considerably improves network lifetime when compared to non-cluster based network in WSN.

**Keywords:** EEE 802.15.4, Minimum Spanning Tree Algorithm, Power-Aware, Radio States, TDMA.

## 1 Introduction

Wireless sensor networks (WSNs), is a distributed collection of sensor nodes which forms network interconnected by wireless communication links. Sensor nodes are often powered by batteries and have limited computing and memory resources. Because of the limitations due to battery life, most sensor networks are built with power conservation in mind. WSNs can operate in an event-driven model or regular continuous monitoring model. Here, the protocol works for an event-driven model, where each sensor will monitor its vicinity and sends its information to the sink via the relay of other sensors nodes whenever an event occurs. There are various standards existing for wireless communication in industry. IEEE 802.15.4 standard is considered because it is specifically designed for low data rate applications. The 802.15.4 standard has many power management mechanisms, based on duty cycle, to minimize the activity of sensor nodes.

### 1.1 IEEE 802.15.4 Standard

IEEE 802.15.4 [1] is a standard for low-rate, low-power, and low-cost Wireless Personal Area Networks (WPANs). A PAN is formed by one PAN coordinator which

N. Meghanathan et al. (Eds.): CCSIT 2012, Part I, LNICST 84, pp. 429–436, 2012.

is in charge of managing the whole network, and, optionally, by one or more coordinators which are responsible for a subset of nodes in the network. Ordinary nodes must associate with a (PAN) coordinator in order to communicate. The supported network topologies are *star* (single-hop), *cluster-tree*, and *mesh* (multihop). The standard defines two different channel access methods: a *beacon-enabled* mode and a *nonbeacon-enabled* mode.

The beacon-enabled mode provides a power management mechanism based on a duty cycle. It uses a superframe structure (Figure 1) which is bounded by *beacons*, i.e., special synchronization frames generated periodically by the coordinator node(s). The time between two consecutive beacons is called *Beacon Interval (BI)*, and is defined through the *Beacon Order (BO)* parameter ($BI=15.36*2^{BO}$ms,with $0 \leq BO \leq 14$). Each superframe consists of an active period and an inactive period. In the active period nodes communicate with the coordinator to which they are associated with, while during the inactive period they enter a low-power state to save energy. The active period is denoted as *Superframe Duration (SD)* and its size is defined by the *Superframe Order (SO)* parameter. It can be further divided into a *Contention Access Period (CAP)* and a *Collision Free Period (CFP)*. During the CAP, a slotted CSMA / CA algorithm is used for channel access, while in the CFP communication occurs in a Time-Division Multiple Access (TDMA) style by using a number of *Guaranteed Time Slots* (GTSs), reassigned to individual nodes.

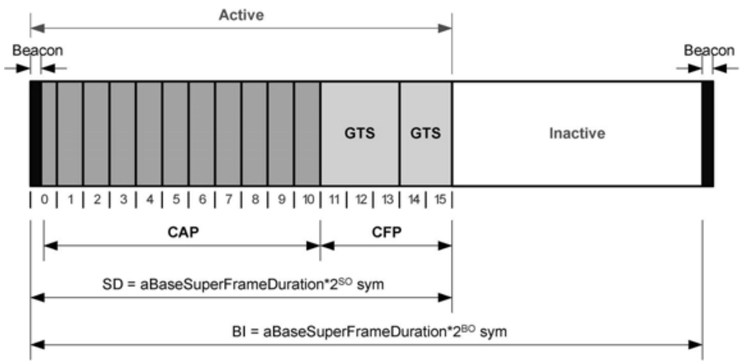


**Fig. 1.** IEEE 802.15.4 Superframe Structure

In the non beacon-enabled mode there is no superframe, nodes are always active (energy conservation is delegated to the layers above the MAC protocol) and use the unslotted CSMA/CA algorithm for channel access. The algorithm is based on beacon enabled IEEE 802.15.4 standard.

## 2 Power-Aware Scheduling Algorithm

A homogenous centralized scheduling of sensor activities is considered to minimize the energy cost and maintain the overall power level of the network. The tree T constructed based on Minimum Spanning Tree is considered as a cluster, with root node as the cluster head and the other nodes as the child nodes. The transmitting time slots for each individual child node and the cluster head is scheduled. The children will send their data in one period and the parent will receive the data when it is awake. Thus, the energy consumption due to the state transition will be definitely saved since each node needs only to wake up twice: once for receiving all data from its children and once for transmitting its data to its own parent.

TDMA is used for scheduling node activities in the CFP within the superframe structure.The time is logically divided into slots with slot size $t_s$, and time slots are synchronized among nodes. A schedule period T is composed of T consecutive time slots. The activities of every node are then repeated with period T. Assume that a node $v_i$ will produce $r_{vi}$ data packets per scheduling period T. When a node $v_i$ is transmitting packets to a neighboring node $v_j$, some other neighboring nodes that are in the listening state will also consume energy. Therefore, the total energy consumption upon the scenario that node $v_i$ transmits in L slots, while k neighboring nodes listening is $(P_{rcv}+P_{tx}+P_{lst}.K) \times L \times t_s$. To minimize the energy consumption, the activities of sensor nodes are scheduled to reduce state transitions.

**Table 1.** Energy Cost Symbols

| Symbol | Meaning |
|---|---|
| $P_{tx}$ | Energy consumption in transmitting |
| $P_{rcv}$ | Energy consumption in receiving |
| $P_{lst}$ | Energy consumption in listening |
| $P_{slp}$ | Energy consumption in sleeping |
| $t_p$ | Time needed to poll channel once |
| $r_{vi}$ | Data packets per period by $v_i$ |

The scheduling of activities for all nodes is to assign each time slot $1 \leq t \leq T$ (scheduling period) to one of the four possible states: transmitting, receiving, listening, and sleeping. Since TDMA access method is used, no nodes need to be in listening state if all nodes are perfectly synchronized. If the synchronization is needed, nodes will also have additional state listening so that adjacent nodes can synchronize their activities. See Figure. 2.for an illustration. Here, the sender node will use a short preamble to synchronize the receiving node. In other words, when a sender wakes up, it will periodically send a message SYN (contains its address and the receiver's address, and the time slots needed for sending data) and listen for the ACK message from the receiver. When the receiver wakes up, it will listen for the message SYN and reply a message ACK if it gets one completed SYN message. After getting the correct ACK message, the sender starts sending data.

In a simple event-driven data collection, a sensor, which is triggered by an event, will wake up and monitor its vicinity, and then, produce some sample data. It will then wake up its parent node (called dominator node sometimes) and send data to it. However, when the dominator node dominates k sensors, it may need to wakeup k times to receive all the data from its children nodes in the worst case, which is energy

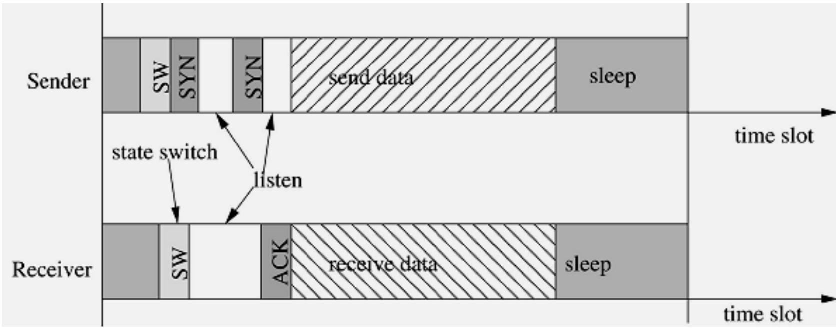


**Fig. 2.** Flow of Synchronization between two nodes

consuming because of multiple state transitions. Our objective is to schedule the activities of sensor nodes to minimize the states transitions (especially from sleeping state to active states), in the meanwhile, the data rate by all sensors is supported. In this paper, it is always been considered for low-data rate WSNs where in the majority of time slots, sensor nodes can sleep to save the energy. Notice that in low-data-rate WSNs, each sensor needs to switch from sleeping state to active state at least once. A schedule has been designed in which any sensor node only needs to wake up at most twice: once for receiving data from its children nodes and once for sending its data to its parent node as shown in Figure 3. This also dramatically reduces the cost of the clock synchronization.

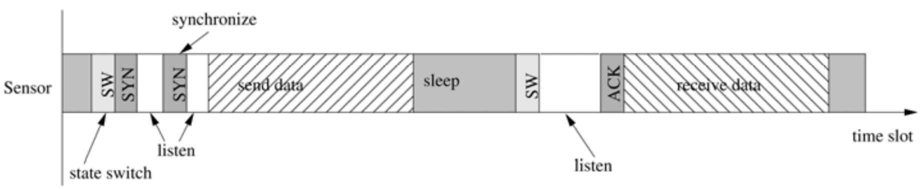


**Fig. 3.** Schedule for each node in a cluster

The clusters is scheduled in the decreasing order of their weight. Within the cluster, the child node or the leaf node need not use the same parent to send its data to the sink. Based upon the battery level they can share the neighboring parent to dynamically route the data to the sink. Our algorithm assumes that the nodes are first synchronized using a short preamble and it ensures that it will not cause any interference for the transmissions of any sensors in $C_i$.

## 2.1 Algorithm

Activity Schedule after constructing a Minimum Spanning Tree (MST)

Step 1: For each sensor $v_i$ do
Step 2: Calculate its receiving-weight $W_i$
(Based on amount of data)

Step 3: Sort the sensor in non-increasing order of weight $W_i$
Step 4: For i=1 to n do
Step 5: For i=1 to n do
Step 6: Sort the path for routing for each node based on the power level
Step 7: If power level is less than the threshold value
Step 8: Goto Step 9 else Goto Step 6
Step 9: Assign Cluster $C_i$ (equivalently sensor node $v_i$ for receiving) $W_i$ earliest available time slots for transmitting that will not overlap with time slots assigned to conflicting clusters.
Step 10: Each sensor node $v_j$ in $C_i$ will be sequentially assigned $w_j$ consecutive time slots for transmitting.

# 3 Proposed System Analytical Model

In this paper, the described cluster based power-aware scheduling (CPAS) algorithm which randomly selects a node as the cluster head (CH) from the group of sensor nodes (as it is done in the LEACH algorithm, in which the residual energy of sensor nodes to is used to select the cluster head), thereby avoiding unbalanced energy consumption of the sensor nodes. The main energy consumption of wireless sensor nodes is typically for the following operations: transmitting a packet, receiving a packet, listening radio signals, sampling the vicinity, reading sample data from the ADC, reading data from the flash, and writing / erasing data in the flash. In this project, it has been focused to synchronize the nodes within the cluster and efficiently schedule the timeslots for data transfer.

The radio is in any of the four states: transmitting, receiving, listening, and sleeping, each of which has different energy consumption (energy consumption per unit time) of $P_{tx}$, Prcv, $P_{lst}$ and $P_{slp}$, respectively (as shown in Table 1). The model also consider the energy $E_{A,B}$ consumed by transiting from one state A to another state B for a sensor and other control units. Typically, the time to restart a sensor node from the sleep mode to active mode is about 4ms [11].For a node $v_i$, if it is scheduled to transmit at time slot t, denote it as $X_{i,S,t}=1$; otherwise denote it as $X_{i,S,t}=0$. Variables $X_{i,R,t}€\{0,1\}$,$X_{i,P,t}€\{0,1\}$ and $X_{i,L,t}€\{0,1\}$ to denote whether the node $v_i$ is scheduled to receive, sleep, or listen at time slot *t* or not, respectively. Energy consumed by state transition is denoted as $E_{P,S}$, $E_{P,R}$ or $E_{P,l}$. See Table 2 for notations used.

In practice, the energy consumed for transition from an active state (such as transmitting, receiving, and listening) to an idle state (sleeping or deep sleeping) is often ignored. Notice that the energy cost by a node $v_i$ in all states is

$$T = \sum_{t=1}^{T} (X_{i,S,t} \,.\, P_{tx} + X_{i,R,t} \,.P_{rcv} + X_{i,l,t} \,.P_{lst} + X_{i,P,t} \,.P_{slp}).t_s;$$

The energy cost for state transitions is

$$\sum_{t=1}^{T} (X_{i,P,t} \,.\, X_{i,S,t+1}.\, E_{P,S} + X_{i,P,t} \,.\, X_{i,R,t+1} \,.E_{P,R} + X_{i,P,t} \,.\, X_{i,L,t+1} \,.E_{P,L});$$

The objective of a schedule S is to minimize the summation of these two energy costs.

**Table 2.** Symbol Notations

| Symbol | Meaning |
| --- | --- |
| $X_{i,S,t}$ | Node $v_i$ transmitting at time t |
| $X_{i,R,t}$ | Node $v_i$ receiving at time t |
| $X_{i,P,t}$ | Node $v_i$ sleeping at time t |
| $X_{i,L,t}$ | Node $v_i$ listening at time t |
| $E_{P,S}$ | Energy consumed from sleeping to transmitting |
| $E_{P,R}$ | Energy consumed from sleeping to receiving |
| $E_{P,L}$ | Energy consumed from sleeping to listening |

For a given data gathering tree T, the time slot needed to transmit and receive by an individual node is fixed because it only depends on the tree structure. So, the difference of the energy consumption from different schedules is the cost for the wake up and clock synchronization. In our scheduling, there are at most two state switches for each node. So, the total number of state switches is at most 2n times of the energy consumption for a node to switch the state. That is, $E_S \leq 2.\ n.\ E_S$, where $E_S$ is the energy consumption for state switch of one sensor in our scheduling, and n is the number of sensors. Because there is at least one state switch for each node in any scheduling, $E^{opt}_S \geq n.\ E_S$, where $E^{opt}_S$ is the optimal energy consumption for state switch. $E_T$ is used to denote the total energy consumption in the active states by all nodes by our method; E as the total energy consumption in our scheduling and $E^{opt}$ as the optimal energy consumption.

$$E = E_T + E_S \leq E^{OPT}_T + 2E^{OPT}_S < 2\ E^{OPT}_T + 2E^{OPT}_S = 2E_{opt}.$$

Thus the energy consumption for the scheduling derived by algorithm is at most twice of the optimum.

# 4 Performance Evaluation

To perform our simulation analysis, ns2 simulation tool [16] is used.The algorithm is experimented with a single cluster, it is assumed the same performance with multiple clusters in a densely deployed network. In our experiments, construct two trees for the data collection using minimum spanning tree (MST) algorithm. The clustering concept is implemented to one tree and make the root node acts as the PAN coordinator and all other nodes operate with a duty cycle for power management. Assumed that in the considered radio model, all nodes are in the carrier sensing range of each other. This minimizes the probability of collisions due to the hidden node problem. The network uses the beacon-enabled mode. The duty cycle is set to about 1.5%, according to the typical values recommended by the ZigBee standard [3] which are in the range 0.1%–2%. Specifically, the Beacon Interval is 125.8 s, while the active period is 1.97s. Note that the active period is large enough to let every node send its data packets in all the analyzed scenarios, so that the enforced duty cycle does not harm the packet transmission process.

In this simulation, different data rates is used to study how the data rate can affect energy consumption. Based on the topology, a sink node (with a fixed position) will collect data from the other sensors in the network. Figure.4 show that energy consumption will decrease when the data rate increases, but the reduction is slower than the data rate increasing. Among different structures, it has been found that MST tree has the smallest time span, and therefore, the less energy consumption, when compared to non-tree node distribution. This is because MST tree structure has the shortest hops from all other nodes to the sink.

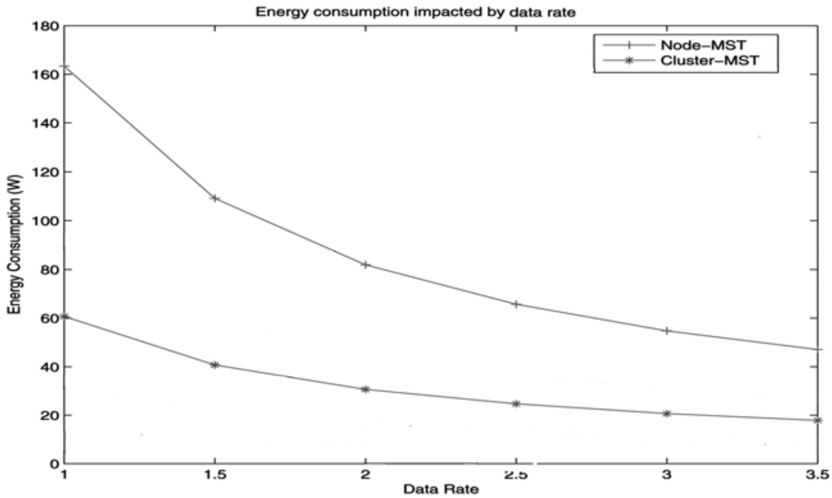


**Fig. 4.** Impact on the energy cost with various data rates in homogeneous networks using cluster based MST tree, and non cluster based MST tree

It is also found that the schedule by our cluster-based method performs much better than the node-based method. In cluster-based method, to avoid potentially delaying the transmission to the next scheduling period, time slots of a parent node after its children nodes' sending time slots is intentionally scheduled. That is, the transmissions are always within one scheduling period. The energy consumption is dramatically reduced when scheduling the node to wake up at most twice in a time period. The energy consumption in cluster-based method is also much better than node-based method, as shown in Figure.4.

## 5 Conclusion

In this paper, a algorithm has been proposed for efficient centralized scheduling that not only remove the unnecessary listening cost, but also reduce the energy cost for state switching and clock synchronization. In our protocol, every node needs only to wake up at most twice in one scheduling period: one for receiving data from its children and one for sending data to its parent. Dynamic routing ability based on the power level of the nodes is also been done to balance the overall network power level. The simulation result shows that the energy consumption is less for cluster-based nodes than non cluster based nodes. This work could also be enhanced for clusters with mobile sink.

**Acknowledgements.** I would like to acknowledge Dr.P.Vanaja Ranjan, Associate Professor, Department of Electrical and Electronic Engineering, Anna University, Chennai, for her valuable discussions and suggestions to development of this algorithm.

## References

[1] IEEE Standard for Information Technology, Part 15.4; Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). IEEE Computer Society (2006)

[2] Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., Anderson J.: Wireless Sensor Networks for Habitat Monitoring. In: Proc. First ACM Int'l Workshop Wireless Sensor Networks and Applications (WSNA 2002), pp. 88–97 (2002)

[3] ZigBee Alliance Official Site, `http://www.zigbee.org`

[4] IEEE: IEEE Std 802.15.4$^{TM}$ (2003), `http://standards.ieee.org`

[5] Clendenin, M.: ZigBee's improved spec incompatible with v1.0, EE Times Europe, `http://eetimes.eu/showArticle.jhtml?articleID=193006080`

[6] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: A survey. Computer Networks 38(4), 393–422 (2002)

[7] Anastasi, G., Conti, M., Di Francesco, M.: A Comprehensive Analysis of the MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks. IEEE Transactions on Industrial Informatics 7(1) (February 2011)

[8] Lu, G., Krishnamachari, B., Raghavendra, C.: Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks. In: Proc. Energy-Efficient Wireless Communications and Networks Conference (EWCN 2004), pp. 701–706 (2004)

[9] Zheng, J., Lee, M.J.: A comprehensive performance study of IEEE 802.15.4. IEEE Press (2004); Leibnitz, K., Wakamiya, N., Murata, M.: Modeling of IEEE 802.15.4 in a cluster of synchronized sensor nodes. In: Proc. 19th Int. Teletraffic Congr. (ITC-19), Beijing, China, pp. 1345–1354 (August 2005)

[10] Ergen, S.C., Varaiya, P.: TDMA Scheduling Algorithms for Sensor Networks, Technical report, Univ. of California, Berkley (2005)

[11] Koubaa, A., Alves, M., Tovar, E.: A comprehensive simulation study of slotted CSMA/CA for IEEE 802.15.4 wireless sensor networks. In: Proc. IEEE Int. Workshop on Factory Commun. Syst. (WFCS 2006), Torino, Italy, pp. 183–192 (June 2006)

[12] Rhee, I., Warrier, A., Aia, M., Min, J.: ZMAC: A Hybrid MAC for Wireless Sensor Networks. In: Proc. ACM Conf. Embedded Networked Sensor Systems (SenSys) (2005)

[13] Yedavalli, K., Krishnamachari, B.: Enhancement of the IEEE 802.15.4 MAC protocol for scalable data collection in dense sensor networks. In: Proc. Int. Symp. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt 2008), Berlin, Germany, March 31-April 4, pp. 152–161 (2008)

[14] Pollin, S., Ergen, M., Ergen, S., Bougard, B., Van der Perre, L., Moerman, I., Bahai, A., Catthoor, F.: Performance analysis of slotted carrier sense IEEE 802.15.4 medium access. IEEE Trans. Wireless Commun. 7(9), 3359–3371 (2008)

[15] Park, P., Di Marco, P., Soldati, P., Fischione, C., Johansson, K.H.: A generalized Markov chain model for effective analysis of slotted IEEE 802.15.4. In: Proc. IEEE Int. Conf. Mobile Ad-Hoc and Sensor Systems 2009 (IEEE MASS 2009), Macau, China, pp. 130–139 (October 2009)

[16] Network Simulator Ns2, `http://www.isu.edu/nsnam/ns`