Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 3 (2011) 96–103

Procedia Computer Science

www.elsevier.com/locate/procedia

WCIT 2010

Delay efficient STEM by pipelining

Taner Cevik [a] *, Derya Yiltas [b], A. Halim Zaim [b]

[a]*Department of Computer Engineering, Fatih University, 34500, Istanbul*
[b]*Department of Computer Engineering, Istanbul University, 34320, Istanbul, Turkey*

Abstract

Wireless sensor networks are comprised of energy constraint, battery powered small devices that sense the environment and transmit the data to the sink in order to take action according to data. Since the sensors are small energy constraint devices energy consumption is the main problem for wireless sensor networks. Energy spent during data communication is much more than spent during in-sensor computing. Most of the effort is spent on designing protocols in order to conserve energy. This paper proposes an improved version of energy efficient MAC protocol STEM by including pipelining mechanism. Results show that the proposed method overperform the original version of STEM by sustaining less delay.

Selection and/or peer-review under responsibility of the Guest Editor.

Keywords: Wireless sensor networks; energy consumption; MAC

---

1. Introduction

Wireless Sensor Networks are energy constraint ad hoc networks that consist of small sensor nodes. Those nodes sense the environment and collect physical data according to the purpose of the network [1]. Sensor nodes are small, energy constraint devices that are capable of processing and transmitting data. Power of sensor nodes is usually provided by battery. It is not feasible to change battery outsourced nodes deployed randomly in a wide geographical area. Besides, power which means lifetime of the network must be long enough to satisfy application requirements [2]. Seldomly, energy can be provided from external sources such as solar cells [3]. However this is an unfavourable method because of the non-continuous behaviour of those sources. Since energy is scarce and lifetime must be held long enough, energy should be consumed carefully and sparsingly [2].

Experiments show that energy consumption in transmitting data is much more than in processing data [4]. Transmitting a single bit or processing thousands of data bits will consume the same amount of energy [5]. Also, the energy consumption in the sensing process is negligible when compared to the energy spent in the communication process.

For most of sensor networks such as used for area surveillance, events occur rarely and communication radios of sensor nodes stay idle most of the time. Nodes spend energy redundantly during those idle states, since there is no event and no data to transmit. This redundant energy consumption is mostly caused by radios making idle listening.

---

* Taner Cevik
E-mail address: tcevik@fatih.edu.tr

doi:10.1016/j.procs.2010.12.017

Thus, in order to prevent redundant energy consumption, nodes should be passed to a state in which they turn their radios off. This state is called sleep state. Number of times the nodes will go into the sleeping mode and the amount of time they will stay in this mode is very important. This periodical sleep-wakeup scheme which is often called "duty-cycle" is performed at MAC layer.

There are also other methods for conserving energy at other levels of network topology. One of these methods is establishing routes in order to lengthen the lifetime of the network. Several works exist in the literature related to this approach such as choosing the nearest node, farthest one or closest angled node [6], etc. as the next hop. Another approach is geographic al adaptive fidelity (GAF) [7], in which the sensing areas being deployed by the sensors are divided into grids. Every node in two adjacent grids must be able to communicate with each other and also with other nodes in the adjacent grid. Each time there must be a single active node in a grid so other ones sleep in order to preserve energy. Nodes in a single grid must be coordinated with each other in order to choose the active node and to provide the load balance by preventing the active node depleting the energy. In another method called geographic random forwarding (GeRaF) [8], the periodic sleep-awake scheduling approach is combined with a geographic based routing method which chooses the node with highest priority. Geographical area between the sender node and the sink is divided into regions. Regions closer to the sink have higher priority than those which are at a further distance. Thus, one of the nodes available in the highest priority region is chosen as the next hop.

As mentioned above, various methods have been applied at the MAC layer in order to define sleep awake schedules for the nodes. In S-MAC [9], a node firstly listens for a certain amount of time and if does not hear any synchronization message during that time, it defines a schedule. Then, it broadcasts this schedule to neighbors. If that node is the first node, others have to obey that schedule. Neighbors must sleep and wake up at the same time with that node. Collision avoidance is provided by RTS/CTS mechanism to avoid hidden terminal problem. In addition, large amount of data is being fragmented and a single RTS/CTS is being sent for all other fragments. All fragments are sent in bursts. RTS/CTS packets define the time period in which the communication channel will be reserved. After each fragment, its ACK packet is waited. By this ACK packet, neighbors realize the communication and does not attempt for a new communication. The reason for an ACK packet after each fragment is that to make new nodes realizing the ongoing communication when they enter the network.

An improved version of S-MAC [10] has been proposed in the literature [10] in which data transmitted from source to the destination undergoes less latency. In S-MAC, packets are being started to be sent after CTS period expires. However, in this improved version, packets are being started to be sent immediately after the CTS packet arrives without waiting the CTS period expiration.

Another approach is presented in T-MAC [11] where unlike S-MAC, active period of the duty-cycle is not constant rather it is adaptive. By dynamically ending the active part, it handles load variations in location and time. By this way, it is prevented to waste energy during idle listening.

Sparse topology and Energy Management (STEM) [12] uses two separate radios and channels for signaling and data transmission. In this method, nodes periodically wake up and listen to the signaling channel. They stay awake long enough to realize a sender node's beacons. Receiver node sends an acknowledgement back via the signaling channel and turns its data radio on. After the communication terminates, nodes turn their data radios off again. However, signaling radios continue to wake up periodically during transmission.

In this paper, we present a delay improved version of STEM by pipelining. On this way, the receiver node does not have to wait for the following next hop to wake up and turn its data radio on. This method helps to save substantial amount of time.

2. STEM with pipelining

We assume that our scenarios are implemented for situations that sensor nodes mostly stay in the idle state because of monitoring the environment as in STEM [12]. As soon as a sensor detects an event, it gets out of the idle state and starts to transmit related data towards the sink. There are two radio channels as shown in Fig. 1 and all nodes use these channels for data communication and signaling.

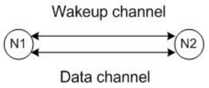


Fig. 1. Signaling and data channels

Fig. 2 represents the situation where three nodes communicate in a flat manner. node a wants to send data to node b, but it must wait for the signaling channel wakeup period. When the wakeup time comes, it starts to send beacons to node b. At this time, node b has a signaling radio in state ‘on’ and listens to the signaling channel if there is a beacon addressed for it. By the time it realizes the beacon, it sends back an acknowledgement and turns its data radio on. Node a starts to transmit the data. After node b completely gets the data, it can not immediately transmit it over the data channel because data radio of the next hop denoted by node c is turned off. Thus it has to wait until the next wake up time to start the process again.

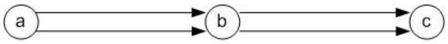


Fig. 2. Simple scenario

State transitions of all three nodes are shown in Figure 3.

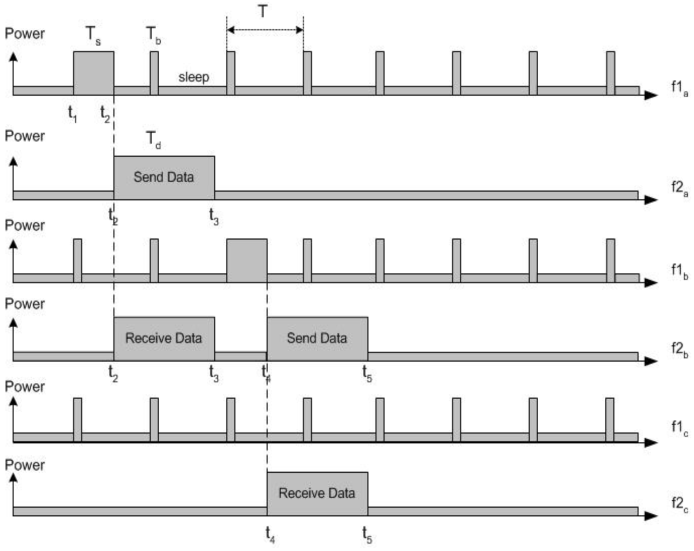


Fig. 3. Send-receive operations for STEM

By adding pipelining approach to the STEM method, node *b* sends acknowledgement back to node *a* after it receives RTS packet from node *a*. Next hop address chosen by node *b*, e.g. node *c*, is defined inside this broadcasted acknowledgement. When node *b* starts to send data to node *c*, data radio of node *c* will be turned on and ready to receive. Thus, node *b* will not have to wait for node *c* to wake up as it is shown in Fig. 4. In Fig. 4, all delays such as propagation and processing delays are ignored.

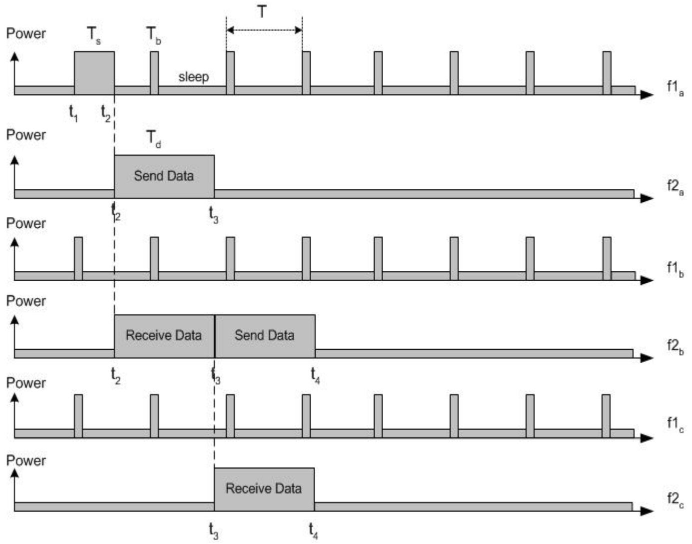


Fig. 4. Send-receive operations for STEM with pipelining

3. Performance evaluation

The performance of the STEM and pipelined STEM methods have been compared via a software code written in JAVA on NetBeans environment. CSMA/CA method is used for collision avoidance. Choosing the nearest, farthest, and closest angled hop routing methods described in [6] have been tested individually. The terms used in the program are described below:

T_d : Time period for data transmission (In our program all data packets are assumed to be same length).
T_s: Setup latency; the difference between the time that a sender starts to send beacons for a specific receiver and the time it gets an acknowledgement from the receiver.
T_b: Time for every node stay awake on f1 in order to determine whether any call for it is presented.
T: Time period for a node to wake up.
B_1: Transmit time of a beacon
B_2: Inter-beacon spacing

Average T_s is calculated in the following as in STEM:

$$T_s = (T+B_1+B_2)/2 \tag{1}$$

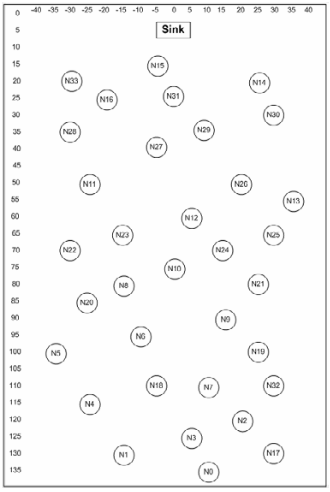
Other values defined in STEM [12] are also used in our program as:

$$B_1+B_2 = 150\ ms,\ T_d = 400\ ms,\ T_b = 225\ ms,\ T = 600\ ms \tag{2}$$

Characteristics of power consumption of the radio simulated are taken from [13]:

$$P_{Transmit} = 14.88\ mW,\quad P_{Receive} = 12.50\ mW,\quad P_{Idle} = 12.36\ mW,\quad P_{Sleep} = 0.016\ mW \tag{3}$$

Processing and other delays are ignored. The simulation has been accomplished using the scenario shown in Fig. 5:



Vicinities: N0 → (1, 2, 3, 17)
N1 → (0, 3, 4, 18)
N2 → (0, 3, 7, 17, 18, 32)
N3 → (0, 1, 2, 17, 4, 7, 18)
N4 → (1, 3, 18, 5, 6)
N5 → (4, 6, 18, 20)
N6 → (4, 5, 18, 7, 9, 20, 8, 10)
N7 → (2, 3, 18, 32, 19, 6, 9)
N8 → (6, 20, 22, 23, 10, 9)
N9 → (7, 6, 18, 19, 8, 10, 21, 24)
N10 → (8, 6, 22, 23, 12, 24, 9, 21)
N11 → (22, 23, 12, 27, 28, 16)
N12 → (10, 23, 24, 25, 13, 11, 26, 27)
N13 → (25, 24, 12, 26, 30)
N14 → (30, 29, 31, 15)
N15→ (16, 33, 31, 14)
N16 → (28, 27, 11, 31, 33, 15, 29)
N17 → (0, 23, 32)
N18 → (1, 4, 3, 2, 7, 5, 6, 9)
N19 → (7, 32, 9, 21)
N20 → (5, 6, 18, 22, 23)
N21 → (9, 19, 10, 24, 25)
N22 → (8, 20, 23, 11)
N23 → (8, 10, 20, 22, 11, 12, 27)
N24 → (9, 10, 21, 25, 12, 13, 26)
N25 → (21, 24, 12, 13, 26)
N26 → (27, 29, 30, 13, 12, 25, 24)
N27 → (11, 23, 12, 26, 28, 29, 16, 31)
N28 → (11, 16, 27, 33)
N29 → (27, 16, 31, 14, 30, 26)
N30 → (13, 26, 29, 31, 14)
N31 → (27, 16, 29, 30, 14, 15)
N32 → (17, 2, 7, 19)
N33 → (28, 16, 15)

Fig. 5. Deployment schema of nodes used in simulation

Euclid theorem is used for calculating the distances between nodes. If the node which is the nearest node to the sending node is chosen as the next hop, the path followed by a packet emerged from N0 is shown in Fig. 6.

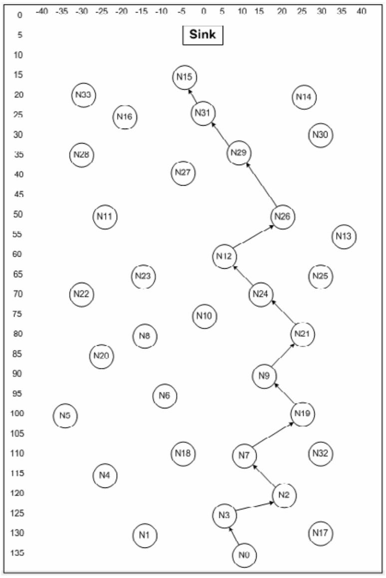


Fig. 6. Path followed when Nearest Node Method (NN) is used

If the node which is the farthest node to the sending node is chosen as the next hop, the path followed by a packet emerged from N0 is shown in Fig. 7.

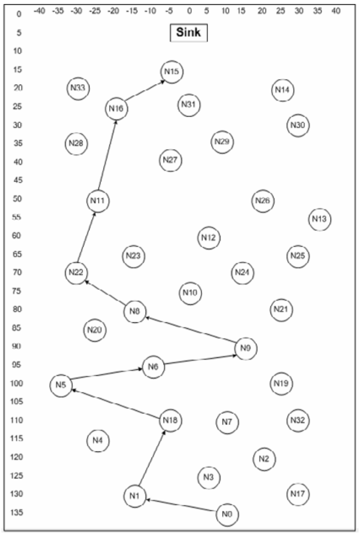


Fig. 7. Path followed when Farthest Node Method (FN) is used

On the other hand, if the node that has the closest angle with the sink is chosen as the next hop, the path followed by a packet emerged from N0 is shown in Fig. 8.

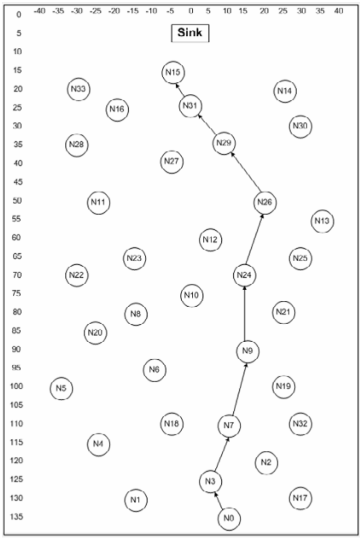


Fig. 8. Path followed when Closest Angled Node (CAN) Method is used

Furthermore, if the node that is closest to the sink is chosen as the next hop, the path followed by a packet emerged from N0 is shown in Fig. 9.

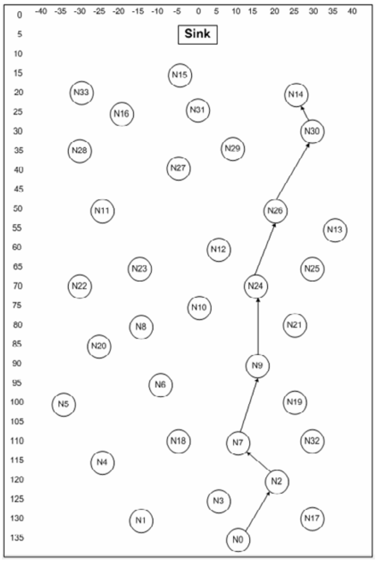


Fig. 9. Path followed when Closest To The Sink Method (CTTS) is used

Fig. 10 shows the comparison of the end-to-end-delays for all four routing methods used incorporation with STEM and pipelined STEM approaches.

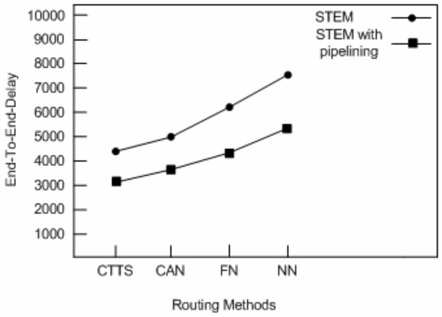


Figure 10. End-to-end-delay with STEM and pipelined STEM

Table 1 shows the total energy spent in the network and the number of hops visited by the packet for four routing methods mentioned above.

Table 1. Energy consumption and number of hops visited

Routing Method	Total Energy Spent (Joule)	Number of Hops Visited
NN	4.9587	12
FN	4.9220	10
CAN	4.8852	8
CTTS	4.8667	7

4. Conclusion

In this paper, a delay efficient version of STEM method using pipelining have been presented. Simulation results showed that, with the proposed method, the delay in the waking up of nodes has been substantially improved as compared to the STEM method. Besides, if CTTS is chosen as the routing method, the energy consumption and the end-to-end-delay will be less than the other methods.

References

1. I. F. Akyildiz,, W. Su, Y. Sankarasubramaniam, E. Cayirci,, *Wireless sensor networks: a survey,* Computer Networks, vol. 38, no. 4, pp. 393-422, Elsevier, 2002.
2. G. Anastasi, M. Conti, M. D. Francesco, A. Passarella, *Energy conservation in wireless sensor networks: A survey,* Ad Hoc Networks, vol.7, no. 3, pp. 537-568, Elsevier, 2009.
3. *Energy Harvesting and Conservation*, Pervasive Computing, vol. 4, no. 1, IEEE, 2005.
4. V. Raghunathan, C. Schurghers, S. Park, M. Srivastava, *Energy-aware wireless microsensor Networks*, Signal Processing Magazine, pp. 40-50, IEEE, 2002.
5. G. Pottie, W. Kaiser, *Wireless integrated network sensors*, Communications, vol. 43, no. 5, pp. 51-58, ACM, 2000.
6. I. Stojmenovic, X. Lin, *GEDIR: loop-free location based routing in wireless Networks*, Proceedings of the International Conference on Parallel and Distributed Computing and Systems (IASTED ‘99), 1999.
7. Y. Xu,, J. Heidemann, D. Estrin, *Geography-informed energy conservation for ad hoc routing,* Proceedings of the International Conference on Mobile Computing and Networking (MOBICOM '01), pp. 70-84, 2001.
8. M. Zorzi, R. R. Rao, Geographic *random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance*, Transactions on Mobile Computing, vol.2, no. 4, pp. 349-365, IEEE, 2003.
9. W. Ye, J. Heidemann, D. Estrin,, *An energy-efficient mac protocol for wireless sensor Networks*, Proceedings of the IEEE INFOCOM, pp. 1567-1576, 2002.
10. T. Chiras, M. Paterakis, P. Koutsakis, *Improved Medium Access Control for Wireless Sensor Networks- A Study on the S-MAC Protocol,* Proceedings of the 14th IEEE Workshop on Local and Metropolitan Area Networks(LANMAN 2005), 2005.
11. T. V. Dam, K. Langendoen, K, *An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks*, Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003), pp. 171-180, 2003.
12. C. Schurgers, V. Tsiatsis, M. B. Srivastava, *STEM: Topology Management for Energy Efficient Sensor* Networks, Proceedings of the Aerospace Conference, vol. 3, pp. 1099-1108, March 2002.
13. X. Yang, N.H. Vaidya,, A *Wakeup Scheme for Sensor .networks: Achieving Balance between Energy Saving and End-to-end Delay*, Proceedings of Real-Time and Embedded Technology and Applications Symposium, pp. 19-26, 2004.