Effective Cache based Policies in Wireless Sensor Networks: A Survey

Sudhanshu Pant
Department of Computer Science
National Institute of Technology
Hamirpur (H.P.), INDIA

Naveen Chauhan
Department of Computer
Science National Institute of
Technology,
Hamirpur (H.P.), INDIA

Prashant Kumar
Department of Computer Science
National Institute of Technology
Hamirpur (H.P.), INDIA

ABSTRACT

The sensor nodes in Wireless Sensor Network are battery powered devices which consumes energy during data transmission, processing, etc. The critical task in WSN is to deal with optimizing power consumption. One possible way to minimize power consumption is by the use of caching the data. Generally data transmission in WSN consumes more energy than processing, so it is good to utilize the benefits of caching so that data access can be made faster. Caching, if used efficiently, could reduce overall network traffic and hence bandwidth can be optimally utilized. In this paper we are reviewing the various data caching techniques in WSNs. This paper aims at future research work and provide researcher with helpful guidelines.

Keywords

Caching, Wireless Sensor Networks, Energy Efficiency.

1. INTRODUCTION

The recent developments in making energy efficient Wireless Sensor Network [1-2] is giving new direction to deploy these networks in applications like surveillance [3-5], industrial monitoring [6], traffic monitoring [7], habitat monitoring [8], cropping monitoring [9], crowd counting [10], etc . The growing use of these networks is making engineers to evolve innovative and efficient ideas in this field. A lot of research in data routing, data compression and in-network aggregation has been proposed in recent years. One of the major factors that could help in building an efficient Wireless Sensor Network is by the use of caching. To best of our knowledge, few research has been done in making an energy efficient Wireless Sensor Network utilizing the benefits of caching the data.

Caching is a technique which provides faster data access in any computing system. With the discovery of cache the accessibility of data has been increased as it stores data to be needed in future and can be retrieved rapidly. Caching [11] has made its impact in the Wireless sensor networks also.

A wireless sensor network consists of a large number of nodes [12] spread over a specific area where we want to look after at the changes going on there. A sensor node generally consists of sensors, actuators, memory, a processor and they do have communication ability. All the sensor nodes are allowed to communicate through a wireless medium. The wireless medium may either of radio frequencies, infrared or any other medium, of course, having no wired connection. These nodes are deployed in a random fashion and communicate within to make an ad-hoc network. If the node is not able to communicate with other through direct link, i.e. they are out of coverage range of each other, the data can be send to the other node by using the nodes in between them. This property refers to multi-hoping.

Generally WSNs are not centralized one as there is peer-to-peer communication between the nodes. So there is no requirement of prior established infrastructure to deploy the network. WSN gives flexibility of adding nodes and removing the nodes as required. But this gives rise to many drastic changes to deal with in the network topology such as updating the path, or the network tree, etc. In a WSN the node that gathers the data information refers to sink. The sink may be connected to the outside world through internet where the information can be utilized within time constraints.

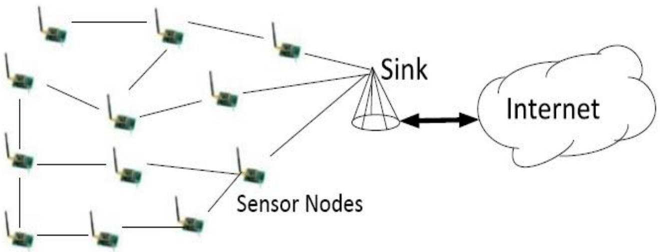

Figure 1. Wireless Sensor Network

The well known problem in using these networks is limited battery life. This is due to fact that the size of a sensor node is expected to be small and this forces constraints on its components i.e. battery size, processors, data storing memory, all are needed to be small. So any optimization in these networks should focus on optimizing energy consumption. In WSN a lot of sensed data and routing information has to be sent which often have some time constraints so that the information can be utilized before any mishap occurs. The energy power consumption is much higher in data communication than internal processing. So energy conservation in WSN is needs to be addressed.

A number of schemes related to caching have been proposed. First scheme includes cooperative caching [13] which utilizes dual-radio based antennas and caches data at the nodes nearer to the sink. In second scheme, [14] proposes making effective caching by data negotiation between Base Station and the sensors, developing expectancy of data change and data vanishing. Another strategy [16] gives a waiting cache which waits for the data of the same cluster until it becomes available within a threshold, aggregating it with the packet from the lower cluster and then sending it to the sink, thus reducing number of packets from the network. The fourth [15] scheme considers the whole network to be a Steiner Data Caching Tree which actually is a binary tree and buffers data at some intermediate node (data cache) such that it reduces the network traffic during multicast. In fifth [17] implementation of TCP support for the WSNs and schemes to avoid end to end transmission between the nodes by caching TCP segments and by local retransmission of TCP segments has been proposed. Some more discussion about cache utilization has also been discussed. Rest of the paper is organized as follows. Section II describes various techniques to exploit caching. Section III gives the conclusion out of using caching in WSNs presented in this paper.

2. IMPROVING PERFORMANCE BY CACHING

As described earlier, cache can be utilized to improve the energy efficiency in Wireless Sensor Networks. Retrieving data directly from sensor node is not a good idea when sensed data does not changes. So we can make use of caching the data in the sensor nodes' cache nearer to the sink or at some point in between sink and the source. This reduces unnecessary load from the network and helps in minimizing the power consumption.

In this section we are presenting some of the proposed work in this field and will be giving various ways in which cache can be utilized to make performance better.

T.P. Sharma et al. proposes the entire sensor field is a square sized grid as shown in figure 2.The idea is to provide data nearer to the sink by caching the data in the nodes nearby the sink [13]. The diagonal of each cell should not exceed the range of high power antenna of the sensor nodes. The path setup for the query and data flow is shown in the figure 2. For storing data, a node uses its low power antenna to communicate with each other.

As sensor nodes have limited storage capacity, this scheme uses a cooperative caching mechanism by utilizing the caches of nearby nodes which is also known as Cumulative Caching (CMC). Basically, Cumulative caching is done by making the Cache Zone utilizing the low power antenna (R_L) of the sensor nodes. Cache zone is taken as region of radius $R_L/2$ around each dissemination node which acts as a cache coordinator. Choosing radius $R_L/2$ makes all nodes in a cache zone to communicate in a single hop within the cache zone. Each cell is considered as a cluster and the nodes nearer to the corner of cluster are Dissemination Nodes (DN). The cluster head is selected out of these four DNs. For communication between sensor nodes and cluster head and within DNs the nodes utilizes its high power antenna which helps in single hop communication.

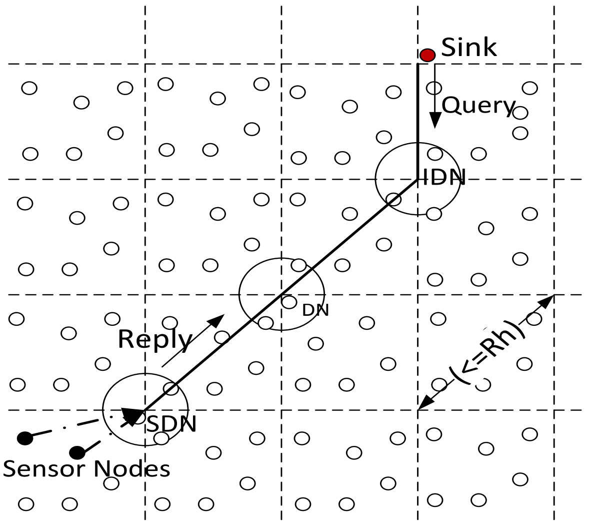

Figure2. Sensor field grid alignment and path setup

Cooperative caching is done intended to increase the proximity nearer to the sink which reduces inter-node transmission and delay in fetching the data items. Initially sink caches the data items in its local cache. When its local cache is full, it passes token to its Immediate Dissemination Node (IDN). When IDNs local cache is full it utilizes one of its caches zone nodes to start caching the data items. When local cache of this node is full, next node from the cache zone of IDN is selected and the node starts caching data. When all the nodes local cache is full in cache zone of the IDN, the token passes to the next Dissemination Node which is nearest to the sink. The new DN becomes the Active Cache Manager and caches the data items in the Cumulative Cache and the process goes on till source dissemination node becomes full. At this point a circle is said to be completed. The circle is restarted again after this.

There is an associated TTL (Time To Live) value with each data item. After first round the sink cache may contain some valid data items. So Item Replacement Policy (IRP) deals with it in the following way.

The sink starts caching data items in some of its available free locations. But when none of its location is free, items has to be replaced by some Item Replacement Policy. But there should be some constraints on how many times a data item should be replaced from the same node. This is limited to the number of data items that can be stored in the local storage of the node. After applying this replacement policy turn goes to the next IDN and process follows the same as in the first round utilizing the cache zone too now. In IRP which data should be removed or replaced may be decided from checking following two parameters:

- Access probability: With this, the item should be replaced is having low access probability. The access probability can be calculated as:

$$Ai = ai / \sum_{k=1}^{N} ak$$

  where, a_i is the mean access rate to item d_i and N is the total number of data items in the local cache of the node.
- TTL value: based on the TTL value and keeping in mind the access probability of the data item, the importance of the data item can be calculated as :
  $Imp_i = A_i * TTL_i$

With this the item with least Imp_i value should be removed by the IRP.

The discovery of data item occurs in the same way as during the establishment of path, i.e., starting from the local cache of the sink then to the cumulative cache of the IDN then next DN until SDN is reached. When there is a global miss a new data is sensed from the source node and hence gets available to the source by using the mechanism described above.

The advantage of above scheme is that it utilizes cache of the neighboring nodes of Cache zone to increase data availability nearer to the sink. The scheme works well when used in a 2-D geometry sensor field. When used in an uneven or 3-D geometry the grid formation becomes impossible. Moreover the scheme proposes the dissemination nodes in every edge of the cell. Suppose a cell is more prone to the changes occurs in it. In this case same DNs will have to involve in the path formation and query replies, which results in power consumption of same DNs and eventually they dies. No further routing can takes place.

Md. A. Rahman et al. gives way to provide effective caching in [14]. The overall energy efficiency of the WSN is a measure of optimizing the value of the factors that actually effects the lifetime of the WSN. Some of the factors that influence the energy efficiency are Data Routing Protocol, Node Distribution, etc. WSN lifetime can be achieved by some other means than data routing and node distribution. These means are:

- **Data Negotiation between sink and the sensors**- This proposed scheme allow sensors to send data only when it changes and enforces sink to store the latest data from all sensors. Finally invalidate cached data of inactive sensor in sink after it is expected that sensor becomes inactive.
- **Developing Expectancy of Data change**- Unnecessary sensing can be avoided by estimating when its sensed data is going to be changed in the sensor before sensing. A sensing is unnecessary if the sensed data is not send to the sink.
- **Data Vanishing**- Duplicate sensed data can be vanished by a routing node in between the source node and sink, and only single packet is send to the sink, as there can be more than one sensor which would have sensed the same data.

The advantage of this scheme is in exploiting the given new constraints that certainly gives better way to design protocols for WSNs.

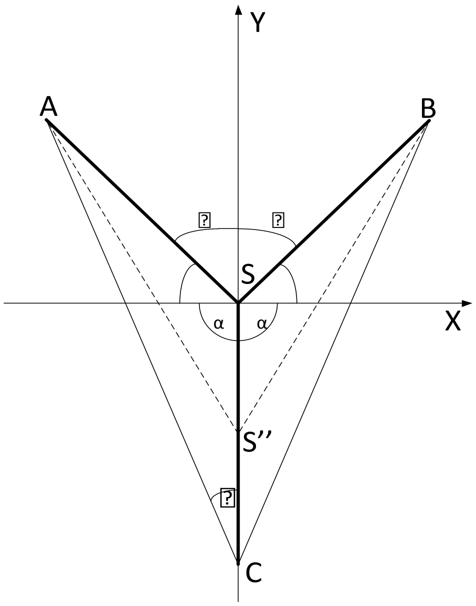

Figure3. Location of Steiner Point

Data can be cached in an intermediate node to provide data availability and reduces transmission delay, when there is a need to broadcast or multicast the data from publisher to subscribers [K.S. Prabh et al. 2005]. The network in this case forms a binary tree dynamically also known as Minimum Steiner Data Caching Tree (MSDCT) [15]. Placing the data in an intermediate node provides asynchronous multicast. So data are provided to the subscriber by the dedicated cache and not directly by the publisher. The author shows the degree of MSDCT to be 3 and at least two of the three internal angles at any internal nodes are equal. The node where data is to be cached is known as Steiner points. Each internal node in the tree has three neighbors. Given the data generation rate of the subscriber and the data refresh rate of the publisher, a Steiner tree can be made with optimal structure.

In figure 3, S is a Steiner point and the value of angles α and γ together finds the location of the Steiner point [15] by the following equation:

$$\mathrm{Sin}\,(\alpha - \gamma - \angle BAC) = (2b / c)\sin\gamma\cos\Psi$$

where, b = AC and c = AB.

The sensor node nearest to the Steiner point is selected as the data cache.

The formation of Steiner Data Caching Tree is done by considering the refresh rates in each edge of the tree. *Data Refresh Rate* on a link is the rate at which data pertaining to the event being monitored by the WSN should be delivered to the receiver. Using these refresh rates the internal angles and Steiner point can be calculated which can be utilized to optimize the performance of the tree.

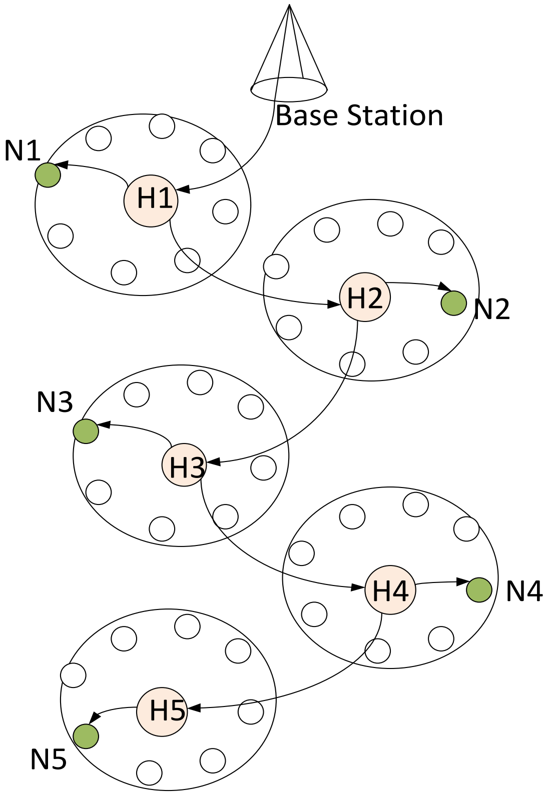

Figure 4. Cluster formation and query path

Since this scheme provides an efficient way to multicasting, it can be utilized in WSNs. Although multicasting is supported but the data cache placement scheme utilizes Steiner points which becomes bottleneck as multicast to various subscribers consumes data from the same node and this makes the Steiner points to die soon.

The network could be considered in the form of cluster as shown in figure 4 [Xu et al. 2008]. Clusters are made with very few nodes and each cluster has a head [16] which is not supposed to sense any data. The function of cluster head is just to process and manage the data between clusters. At a time only two nodes can be active, one is the cluster head and the other one is sensing the data. The energy consumption of head node becomes very high as compared to the other node in the cluster so the role of head node is played in turn to utilize energy consumption in an efficient manner.

First Base Station sends query to the sink and then sink to cluster head H1. Then H1 sends this query to the H2 and N1 and so on.

When the leaf node N5 sent the data to the H5 , H5 sends it directly to the H4, while H4 is waiting there for the N4 to sense data and send it to the H4.when H4 receives data from N4 it aggregates it with the data from H5 and then sends it to the H3. H3 will again apply the same process as H4. without giving way to form it. The routing of data is assumed to be in single hop between two cluster heads, which is not possible for every case unless they are in coverage area of each other which is quite often.

In wireless environment TCP is not a good performer. But connecting the WSN to the outside world through internet makes it usable in various monitoring applications. An optimization in TCP [17] has been proposed by the Dunkels et al. and the scheme could perform well in WSNs. Distributed TCP Caching (DTC) tries to avoid end-to-end retransmission and uses segment caching to provide this. Instead of end-to-end transmission this scheme uses local retransmissions for the lost acknowledgements. A major problem with this scheme is that the last node or the nodes closest to the Base Station may runs out of energy. DTC takes care about this issue by distributing the load in the network.

Before applying TCP into the WSN some changes has to be done so that WSN becomes usable. Some of the proposed works are as follows:

- A ***Spatial IP addressing*** should be used that uses spatial location information to provide semi-unique IP addresses to sensor nodes.
- The TCP header overhead can be reduced by using compression techniques for WSNs.
- For routing, application overlays network can be used to perform data-centric routing and data aggregation for TCP/IP sensor networks.
- As the size of sensor nodes are too small, TCP/IP stack may proves heavy for the nodes to deal with. But [18] has shown that µIP implementation of the TCP/IP stack can run only 8-bit microcontroller and works under a few bytes of RAM.

TCP was designed for wired networks where bit errors are usually not there. Any packets drop is concluded as the congestion in the network. But in wireless networks, bit errors are usual and TCP can misinterpret them as congestion in the network and slow down the sending window, even when network is not congested. As TCP uses end-to-end retransmission, all the nodes in the path of communication in wireless sensor network have to forward the same packet which consumes energy.

So a local retransmission is a good idea to avoid above problem. The node which will be selected for retransmission should be located nearer to the receiver. DTC implements this feature and requires no change in the sensor nodes and receiver. Only intermediate nodes have to deal with this. It is assumed that nodes have only single location to store TCP segment. More precisely a node can store only one TCP segment due to less storage space.

The figure 5 shows how DTC algorithm works. DTC calculates when the packet is lost in the transmission and cache the lost packet in the node when it gets lost. DTC does a Packet Loss Detection, by storing the largest segment of the packet during forwarding in the cache, and relies on link level acknowledgements to detect the packet drop. If there is no feedback from the next node, the node locked the data in the cache. This data is not allowed to overwrite with further segment number of the packets.

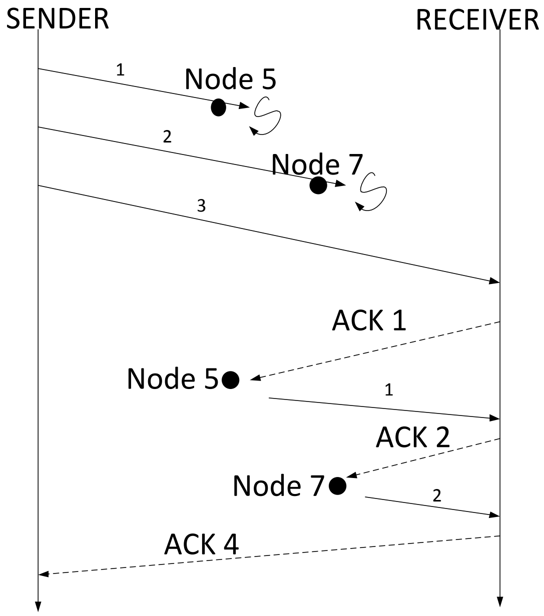

Figure 5. Distributed TCP Caching

In the above scenario TCP sender sends three segments and the lost nodes are cached by the intermediate nodes. Thus, node 5 caches the lost segment 1 and node 7 caches the lost segment 2. This is, as described, done by Packet Loss Detection. Now in receiving segment 3 by the receiver, it replies with the ACK1. On receiving ACK1 by node 7, it doesn’t send its cached segment 2 rather it waits. When ACK1 is received by the node 5, it sends cached copy of segment 1 to the receiver. Node 5 will not send ACK1 to the sender to avoid unnecessary retransmission. On receiving segment 1 by receiver it replies with ACK2. The ACK2 is received by node 7 and it retransmits the cached copy of segment 2. The receiver on receiving this segment ask sender for further segments with ACK4. This way the intermediate nodes, here node 5 and node 7, clears its cached segments and are ready to accept new segments.

Thus, DTC helps in reducing the number of packets in the network, avoids end-to-end retransmissions and does local retransmissions utilizing the intermediate nodes cache. The TCP support in Wireless Sensor Network [17] does not considers the local (data-link) transmission delay and overhead. The scheme is given only for single TCP data flow. Multiple data flows in TCP supported Wireless Sensor Network may not perform well and needed to be considered.

Some of the other areas where caching can be utilized is proposed in [19], [20]. In [19], scheme exploits caching for faulty nodes in WSNs and proposes a mechanism to handle the packets when node gets failed. The operational node handles the data during the node failure and recovers it when the faulty node again gets workable.

An improvement in DTC [20] has been proposed with implementation of Automatic Repeat ReQuest (ARQ). The scheme involves both Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) and the link-layer reliability.

3. CONCLUSION

In this paper we have discussed the various techniques that use caching to improve the performance of the Wireless Sensor Networks. The proposed works are shown to perform well if applied practically in real world scenario under particular situations by the means of proper simulations. We can only make an energy efficient WSN only when the load is equally distributed in the network so that all the nodes consumes power equally and network becomes operational as long as possible.

We hope that this summarized work will give better understanding to the researcher those who are new to applications of WSNs and helps them in forwarding new ideas to enhance the working of these networks.

4. REFERENCES

[1] J. C. Chen, L. Yip, J. Elson, H. Wang, D. Maniezzo, R. E. Hudson, K. Yao, and D. Estrin, "Coherent acoustic array processing and localization on wireless sensor networks," in Proc. of the IEEE, Vol. 91, No. 8, pp. 1154–1162, Aug. 2003.

[2] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling ultra-low power wireless research," in IPSN/SPOTS 2005, Apr. 25–27 2005.

[3] C.-K. Chang and J. Huang, "Video surveillance for hazardous conditions using sensor networks," in Proc. of the IEEE International Conf. on Networking, Sensing and Control, pp. 1008–1013, March 2004.

[4] L. Jiao, Y. Wu, G. Wu, E. Y. Chang, and Y.-F. Wang, "The anatomy of a multi-camera security surveillance system," in ACM Multimedia System Journal Special Issue, Vol. 10, No. 2, Oct. 2004.

[5] A. Jain, Y.-F. Wang, and E. Y. Chang, "A collaborative camera system for surveillance," in UCSB Technical Report, Nov. 2004.

[6] L. Nachman, R. Kling, R. Adler, J. Huang, and V. Hummel, "The intel mote platform: a bluetooth-based sensor network for industrial monitoring." in IPSN 2005, pp. 437–442, Apr. 25–27 2005.

[7] N. S. Love, I. Masaki, and B. K. P. Horn, "Efficient traffic monitoring," in MIT ITRC Report, 2003. [Online]. Available: http://www.mtlweb.mit.edu/researchgroups/itrc/ITRC_report/ MTL_2003_report/nsl_mtl_annual_2003.doc

[8] J. R. Polastre, "Design and implementation of wireless sensor networks for habitat monitoring," M.S. Thesis, U.C. Berkeley, 2003.

[9] M. A. Miskam, A. B. Nasirudin, I. A. Rahim, "Preliminary Design on the Development of Wireless Sensor Network for Paddy Rice Cropping Monitoring Application in Malaysia." ISSN, Vol. 37, No.4, pp. 649-657, 2009.

[10] D. Yang, H. Gonzalez-Banos, and L. Guibas, "Counting people in crowds with a real-time network of image sensors," in Proc. of IEEE ICCV, Oct. 2003.

[11] H. B. Jang, A. Kashif, M. Park, and S. W. Chung, "Reliable Cache Memory Design for Sensor Networks.", 3rd International Conference on Convergence and Hybrid Information Technology, Vol. 1, pp.651-656, 2008.

[12] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E.Cayirci, "A survey on sensor networks," IEEE Communications Magazine, Vol. 40 Issue. 8, pp. 102-114, August 2002.

[13] T.P. Sharma, R.C. Joshi, Manoj Misra, "Dual Radio based Cooperative Caching for Wireless Sensor Networks", 16th IEEE Conference, pp. 1 – 7, 2008.

[14] Md. A. Rahman, S. Hussain, "Effective Caching in Wireless Sensor Network", 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07), Vol. 1, pp. 43-47 2007.

[15] K.S. Prabh, T.F. Abdelzaher, "Energy-Conserving Data Cache Placement in Sensor Networks", ACM Transactions on Sensor Networks, Vol. 1, No. 2, pp. 178–203, November 2005.

[16] J. Xu, K. Li, Y. Shen, J. Liu, "An Energy-Efficient Waiting Caching Algorithm in Wireless Sensor Network", International Conference on Embedded and Ubiquitous Computing, Vol. 1, pp. 323 - 329, 2008.

[17] A. Dunkels, J. Alonso, and T. Voigt, "Distributed TCP Caching for Wireless Sensor Networks," in Proceedings of the 3 rd Annual Mediterranean Ad-Hoc Networks Workshop, 2004.

[18] A. Dunkels. "Full TCP/IP for 8-bit architectures". In MOBISYS`03, San Francisco, California, May 2003.

[19] M.N. Al-Ameen, MD. R. Hasan, "The Mechanisms to Decide on Caching a Packet on Its Way of Transmission to a Faulty Node in Wireless Sensor Networks Based on the Analytical Models and Mathematical Evaluations", 3rd International Conference on Sensing Technology, Tainan, Taiwan, Nov. 30 – Dec. 3, 2008.

[20] A. Ayadi, P. Maill´e, and D. Ros, "Improving Distributed TCP Caching for Wireless Sensor Networks", Ad Hoc Networking Workshop (Med-Hoc-Net),9th IFIP Annual Mediterranean, June 2010.