

Design and Deployment of 2 Fault Tolerant Hierarchical Sensor and Tracking System in Underground Mines

Anurag D
Indian Institute of Management
Calcutta
Kolkata, India 700104
anurag@email.iimcal.ac.in

Siuli Roy
Indian Institute of Management
Calcutta
Kolkata, India 700104
siuli@iimcal.ac.in

Somprakash Bandyopadhyay
Indian Institute of Management
Calcutta
Kolkata, India 700104
somprakash@iimcal.ac.in

ABSTRACT

In this paper, we describe the design and deployment of a wireless sensor system based on 802.15.4 to track underground miners and other assets. The system was developed for the Central Institute of Mining and Fuel Research (CIMFR) in Dhanbad, India. The critical requirements in such developing regions are: a low cost solution and guaranteed fault tolerance. The research in this paper is along two aspects – a) development of an addressing and routing scheme suited for tracking applications where the address wastage problem in C_{skip} (ZigBee) is solved and b) development of a unique fault tolerant strategy which is completely scalable and improves network lifetime. The evaluation of the system is made by way of both a simulation and a pilot implementation in an underground mine. We compare the fault tolerance scheme with the AODV protocol where we show the significant improvement of our scheme in terms of the control packet overhead. The practical results of deployment measure the packet loss, latency and lifetime of the system. By our analysis, we conclude about the good MAC layer robustness provided by 802.15.4, thus negating the need for higher layer control mechanisms for packet loss, like TCP, in wireless sensor systems.

Categories and Subject Descriptors

C 2.1 [Network Architecture and Design]: Network Topology

C 2.2 [Network Protocols]: Routing Protocols

General Terms

Algorithms, Performance, Design, Reliability.

Keywords

Fault Tolerance.

1. INTRODUCTION

The availability of consistent, accurate and timely information greatly improves the quality and speed of

planning and decision making in any organization. Total Asset Visibility (TAV) [11] is a term, used in US Department of Defense's logistic practices, that implies knowing where assets are at all times. It also implies having a unique identity for each item and knowing what is happening to it, as it happens. TAV can usefully be applied to any organization for improving enterprise visibility. Timely and accurate information on the location, movement, status, and identity of units, personnel, equipment, material and supplies can improve the resource utilization by a great extent. It also improves the capability to act upon that information for better performance of the organization. The core of this kind of Asset Visibility System is the Automatic Object Identification technology, like RFID, integrated with wireless mesh networks to communicate this identification data to a remote station.

An Enterprise visibility system is not confined to asset visibility only. The same framework can be used to monitor the enterprise environment, to protect it against possible damage by terrorist attacks, fire or emission of toxic gases (in case of chemical industries). Wide variety of sensors connected together using wireless sensor networks forms the core of this Environment Visibility System. Moving a step further towards a "better" visibility, all this information should be available in real-time from anywhere. This "anywhere" accessibility of information is made possible through the Internet.

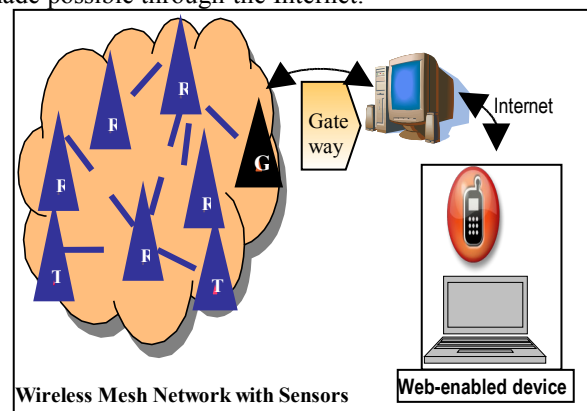


Figure 1. A conceptual Framework for Tracking, Monitoring and Actuation of Enterprise objects through the Internet

This paper presents a unified framework for both asset visibility and environment visibility of an enterprise through real-time tracking and monitoring of enterprise objects which can be also made available over the Internet (Figure 1).

The system was developed particularly for the CIMFR, Dhanbad, India and a 10 node wireless system was deployed. The topology of the system was a hierarchical tree and comprised of three components: a) A PAN (Personal Area Network) Coordinator which is the super parent and initiated the network formation; b) a Router or an FFD (Full Functional Device) which joins the network through a parent and acts as a parent for more FFDs to join; and c) an End-Device or RFD (Reduced Functionality Device) which acts as a mobile tag and is carried by the asset to be tracked. We measured extensively the packet loss, the latency and the network lifetime of the system. Tracking of assets in underground mines has unique challenges. The characteristics of such an application are:

- A pre-determined area of deployment where the path to be tracked is known apriori. Thus the routers (FFDs) are placed along this path.
- The data flow is always between the coordinator and an end device (RFD) and vice-versa. There is never a need for two end devices to talk to each other.
- Tracking applications are typified by a large area, but a well defined path. Further, the paths are bounded on either side (for example walls of a mine). Therefore the network is sparse and the probability of forming a mesh is low.
- In developing regions like India, price sensitivity is high. A deployment solution of placing wireless devices in random places and in as large a number as possible, is very rarely acceptable.
- Pre-determined bounds on fault tolerance, network latency and lifetime are mandatory.

This paper is structured as follows. Section 2 develops the motivation for the research through the literature survey. The proposed network architecture is presented in Section 3 and the simulation results in section 4. We present the practical results observed in section 5 and conclude with scope for further work in section 6.

2. RELATED WORK

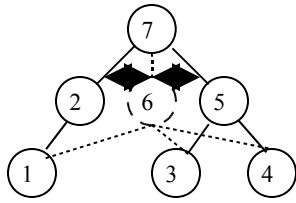
The ZigBee standard based on the IEEE 802.15.4 PHY and MAC Layers provides specifications for two kinds of network topologies- mesh and tree. Mesh networks utilize the slightly modified ad-hoc on-demand distance vector (AODV) [12]. The mesh network has inherent support for fault tolerance, but without studying the topology, questions on the exact amount of fault

tolerance, network lifetime and latency cannot be answered. Further, a deployment solution of dispersing wireless nodes in a random fashion is unacceptable (see above). The hierarchical tree topology suits the tracking application perfectly. However, the C_{skip} algorithm used in ZigBee [16] was found lacking when a sparse, but large depth in the network is needed. For example, the ZigBee tree structure is unable to support a chain of more than 7 devices when each device can have 6 children. (i.e a multi-hop reach of greater than 7 is not possible). Further, the ZigBee protocol does not support fault tolerance. We thus develop a new network layer based on 802.15.4 with support for fault tolerance.

The address wastage problem has been identified as a serious concern earlier [7]. The ZigBee alliance has addressed this problem by supporting a stochastic addressing scheme in the latest specification released towards the end of 2007. The specification has been recently made available to the general public. The stochastic addressing scheme assigns node addresses in a probabilistic manner to reduce the chance of address duplication. However, it cannot ensure unique addressing and measures to handle address duplication have to be devised. Further, the routing for such a mechanism is not specified and would typically be supported by AODV.

Fault tolerance has been extensively studied in computer architecture design. In Wireless sensor networks, the fault tolerance has looked at five categories: Node Placement, Topology Control, Event Detection, Data Gathering/Aggregation and Sensor Surveillance [5]. A k -fault tolerant network is one where every node is k connected and the network remains connected even after k -node failures.

Work on deploying additional relay nodes to ensure a pre determined k -connectivity between all nodes has been made in [10]. Existing sensors are assumed to be modeled as a unit-disk graph and their geographical co-ordinates is known. The problem of placing the minimum number of additional sensors geographically among the existing sensors is NP-hard. The authors of [10] provide an approximate solution to the optimal deployment of additional nodes. The problem of optimizing the deployment of additional nodes that have different transmission radii has been solved in [14]. However, in either of these works, the addressing and routing of sensors has been assumed. Further, we contend, the tracking applications in underground mines have very little probability of a single device acting as a support for multiple other devices (see Figure 2).



Node 6 acts as a backup for nodes 2 and 5. In underground mines, there is very little probability of such a scheme

Figure 2. A sample topology resulting from current optimization schemes for fault tolerance.

3. NETWORK ARCHITECTURE

The design of the system was made based on the following requirements.

- Tracking of assets in a fixed area and in a pre-determined path. Location of assets must be shown on a map at all times.
- Achieve 2 fault tolerance. Inform the PAN coordinator when a fault occurs.
- Determine bounds on network lifetime, latency and packet drop.

The network was developed in three phases. In phase I, we determined the placements of routers along the path and resolved the addressing and routing aspects. This topology was made in the form of a hierarchical tree. In phase II, we developed the 2 fault tolerance by means of a novel Active-Standby system. This setup is completely scalable and in addition to fault tolerance, increases the network lifetime. In phase III, we developed a front end to display the incoming packets from the wireless system on a map displayed on a Personal Computer.

3.1 PHASE I: Hierarchical Topology Development

We modify the concept of the C_{skip} algorithm to prevent the wastage of address space but at the same time maintain efficient routing through a tree based structure. A pictorial topology is first prepared with the routers placed at appropriate distances along the path. The maximum number of end devices each router would have to handle is estimated (E_n). The address of each router is then determined by the simple algorithm as shown below. The algorithm works on the depth first concept where the deepest router is assigned an address first and the algorithm “works up” the topology. The needs of tracking allow us to set the address in advance to preserve the tree structure.

Algorithm 1 Hierarchical Address allocation

- 1: Determine location of FFDs and their parent/children
 - 2: **input** : E_n , set R of Routers and their children
 - 4: **Initialise** node \leftarrow PAN coordinator, $addr \leftarrow 0$
 - 6: **For** all children of node from left to right
-

- 7: **If** node has no children
 - 9: **assign** node address \leftarrow $addr$
 - 10: **Increment** $addr$ by $E_n + 1$
 - 11: **End if**
 - 13: **Else**
 - 14: **assign** node \leftarrow node’s children
 - 15: **End if**
 - 16: **End For**
-

Every Router maintains an address list, which has the addresses of its child routers and child end devices. The routing is done based on this. The routing algorithm is shown below.

Algorithm 2 Routing in Hierarchical Topology

- 1: **input** : set C, children address of Router R, Destination address, D, of received packet at R
 - 2: **For** all children of node R from left to right
 - 3: **If** children address \leq D
 - 4: **assign** next hop \leftarrow children address
 - 5: **break** from loop
 - 6: **End if**
 - 7: **End For**
 - 8: **Return** next hop
-

The addressing algorithm is akin to the C_{skip} in the sense of addressing in the depth first approach. However, C_{skip} assumes the worst case and would earmark addresses for non-existent nodes. This leads to a huge amount of address wastage and this precisely is avoided in the static algorithm. The tracking applications have a well defined path and this is known in advance. Thus, the need to maintain addresses for future nodes does not arise. Further, the address allocation can be suitably tweaked to include an address provision for future nodes as the need may be. This can be achieved, by including a larger value for E_n .

The address of a node denotes the maximum address of all devices under it. A router assigns address to end devices as $node_address - n$. Where $1 \leq n \leq E_n$. At any given point, the property of the node addresses is maintained and thus routing is achieved. For example, consider an end device joins the network at node 35. It is assigned an address of 34. The routing decisions would be at each node and checks under which child, can the destination address exist and forward accordingly. The pictorial representation is shown below.

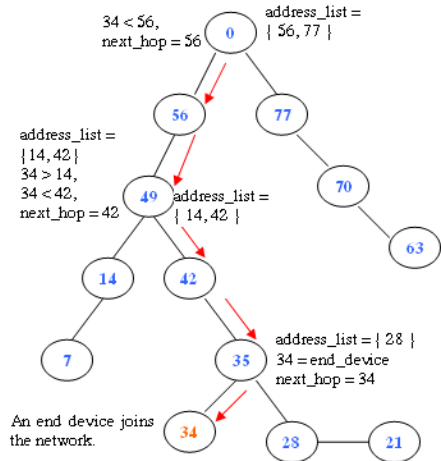


Figure 3. Addressing and Routing in the modified Hierarchical Tree.

3.2 PHASE II: Developing Fault Tolerance for Hierarchical Topology

To achieve fault tolerance, we introduce duplicate routers for every router present. Each of these routers has the same address as its counterpart (the active router). The Active-Standby pair decide among themselves to toggle the active and stand by roles. By having the same addresses, the parent and children need not be aware of which of the Active-Standby nodes is active. The physical layout of the standby node is made close to the active. We placed the standby routers on the other wall of the mine facing the active. The control messages exchanged by the active standby pair are shown below.

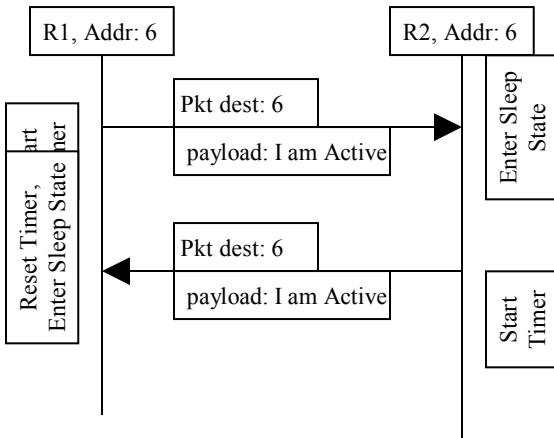


Figure 4. Control messages exchanged among the Active-Standby Pair

Upon boot, the node sends an “I am Active” packet with its destination address set to its own address. The nodes running 802.15.4 (CC 2430) cannot receive and send packets at the same time and hence the transmitting node does not receive the packet it is sending. If the corresponding node is up at this time, upon receiving of

the “Active” packet goes to sleep. The former device starts a timer and expects to receive the “Active” packet from its counterpart. Once this is received, it goes into sleep and the standby now takes the role of the active node. In case, the standby does not wake up from its sleep, the active node sends a “Standby Failure” packet to the PAN Coordinator.

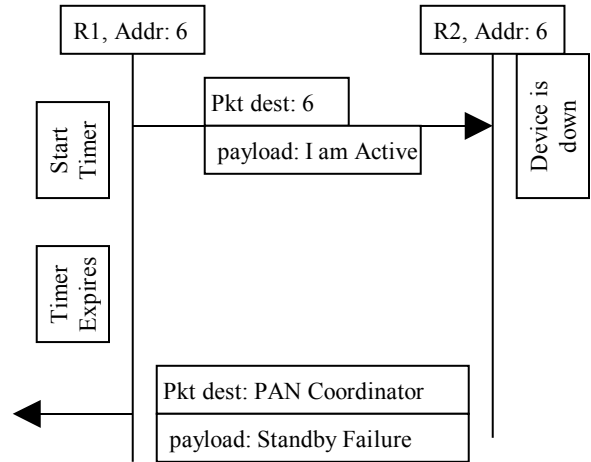


Figure 5. Control messages in case of a node failure.

The Active-Standby scheme has two important properties that make it ideal for deployment in resource constrained networks. The distributed sleep mechanism is localized to only the Active-Standby pair and no communication needs to be made with their parent or their children. This significantly reduces the control packet overhead and correspondingly increases the network lifetime (See simulation results). Second, the scheme is completely scalable. If we wish to achieve 3 fault tolerance, we include a third device with the same address. This also triples the network lifetime.

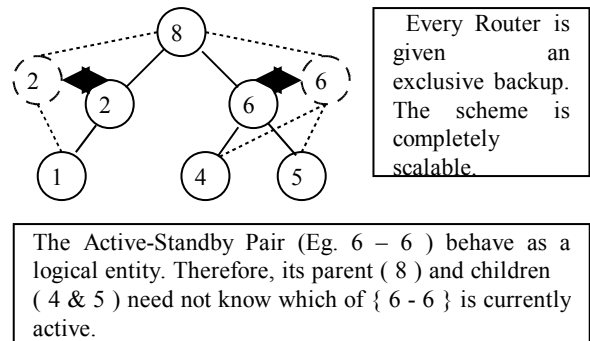


Figure 6. Localisation of control messages imply minimal control messages and excellent scalability.

3.3 PHASE III: Developing the Front End

The PAN coordinator collects periodic messages from the tags (end devices) and the Routers. The payload of the packet includes the name of the node and its address. Thus by having the addresses of the Routers and the addresses of

the end devices, the location of the end devices can be determined. The front end was developed in Visual Basic and includes a map of the area of deployment. The user is able to move the location of the router by drag and drop and place it accordingly on the map. Upon receiving the packets from the end devices, a corresponding icon is shown near the Router to which the end device is associated. A screen shot of the front-end developed is shown.

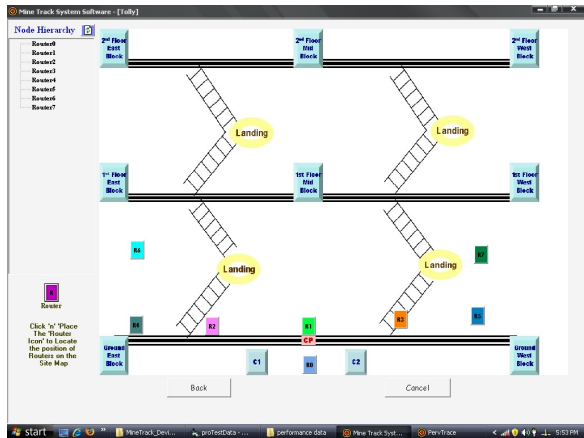


Figure 7. Front end design with schematic map of the tracking area.

There are however two issues with this scheme- a) there will be a small but finite amount of time when both the devices in an Active-Standy pair are active. This will lead to duplicate messages being propagated. This is a relatively harmless issue; b) a more important issue is of a node failure as soon as it sends the “I am Active” message. In such a scenario, the standby device (which is now in sleep state) will continue to sleep until its timer expires. During this time, network connectivity is lost. This problem can be abated by having a small Sleep-Awaken cycle. However, by having a smaller cycle, power consumption is increased in two ways- increase in control packets and the power needed to save the data registers while entering sleep and retrieve the data while coming out of sleep.

4. SIMULATION RESULTS

The simulation study was made to test the Active-Standy scheme with respect to the control packet overhead and the increase in network lifetime. For control packets, the setup was compared to AODV, where we assume the AODV setup comprises of devices that have twice the usual life. When a device goes down, the AODV protocol broadcasts the complete network to determine the new route. This broadcast is the overhead in AODV.

The hierarchical network topology for the simulation was generated using two discrete probability distributions – uniform and geometric. In uniform distribution, a new

device joining the network has an equal chance to choose any of the already joined devices as its parent. In geometric distribution, a new device can choose as its parent, a device that has recently joined the network, with more probability than a device that has joined the network much earlier. This is a more realistic simulation of the practical scenario since a hierarchical network grows from the parent down through the children. A child joining the network will induce another child to join it, rather than its parent.

The simulation of the node failure was done using the geometric distribution where the probability of a node failure increases with time.

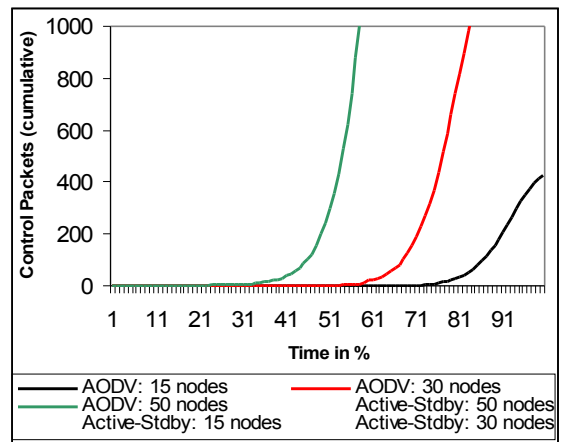


Figure 8. Simulation study of cumulative control packets needed for Active-Standy scheme Vs AODV.

As can be seen in the simulation, the AODV scheme is severely heavy on the control packets as the number of devices failing increase. The active-standby scheme is costlier during the initial time, but a slower slope more than makes up for it. The phenomenal increase in the cumulative control packets for AODV can be explained by observing the number of control packets sent during the lifetime of the network.

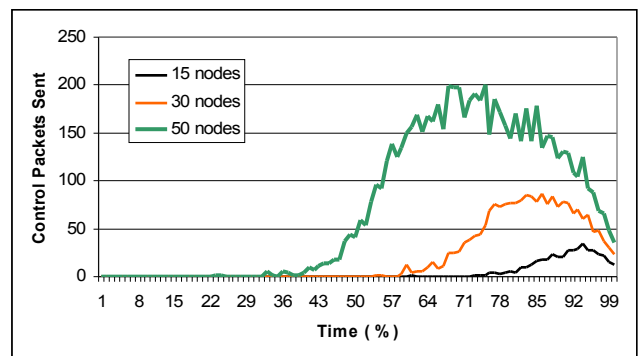


Figure 9. Control packets sent by AODV protocol

During the initial part of the network lifetime, hardly any device fails, therefore there are no control packets. However, as time goes on, the probability of a device failure increases (due to battery exhaustion) and more

devices fail. Thus the control packets increase. As further time passes, the number of devices left in the network reduces and thus a device failure will lead to a lesser number of control packets.

The simulation study was made in an object oriented framework developed in C++. The values are the average of 100 runs for each network size of 15, 30 and 50. The Active-Standby sleep duty cycle was set to 5%. The results shown are for the geometrical network formation. The binomial probability distribution results were similar. The theoretical lifetime increase expected is shown along the practical results.

5. TEST BED IMPLEMENTATION & PRACTICAL RESULTS

The designed 2 Fault tolerant network was implemented on TI supplied SOCs, CC2430 [4], running TIMAC (802.15.4) [13]. The network layer was implemented as a simple queue where messages with higher priority from the MAC layer (MLME and MCPS) were put to the front of the queue and messages of lower priority (UART) were put at the back of the queue. These messages were processed by raising an interrupt and the writing and reading from the queue were interrupt-shielded. We avoided the use of an operating system like Tiny-OS as the performance of the queue based system was impressive and scaled well.

A total of 10 devices were used, including one PAN coordinator, 6 Routers (Active-Standby) and 3 end devices. The topology was made as shown in figure 10. The devices were programmed with different loads of messages per second and for each setting were left to run overnight where the interruption due to people moving in the area is non-existent. The devices for each run were placed in the exact location as before. Thus the randomness of the wireless medium was reduced as much as possible. We measured the packet drop and variance in the packet arrival rate at the PAN coordinator.

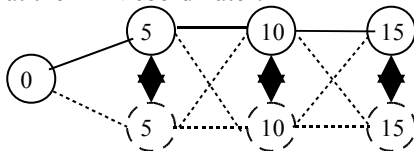


Figure 10. Topology of the Pilot Implementation. Nodes are placed along a path bounded by walls on either side.

The measure of variance as a function of the frequency of packets sent is shown in figure 2. The variance increases dramatically as the packet rate crosses one every three seconds. However, in the entire experiments carried out the packet loss rate was 0.003 % for 3 hops with the interval of 1.5s. In all other cases, there were no packet drops. We thus conclude, in such scenarios where wireless connectivity can be ensured and there is relatively low frequency of packet generation, the MAC support provided

by 802.15.4 is robust enough to prevent developing intricate control mechanisms in higher layers.

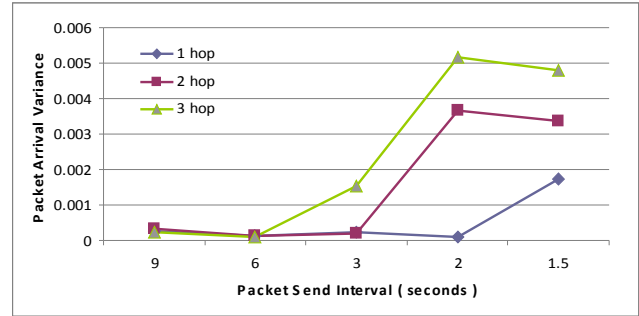


Figure 11. Topology of the Pilot Implementation. Nodes are placed along a path bounded by walls on either side.

The power consumption of the devices was measured and is tabulated below.

Table 1. Power Consumption Comparison

Parameter	Data Sheet Value []	Measured Value (CC 2430 Development Boards)
Tx current	27mA	53mA
Rx current	27mA	55mA
Sleep current	0.5 μ A	27mA

The devices were powered by two AA batteries developing 1400 mAh combined. The expected lifetime of the system then works out to 35 hours ($1400 / ((55 + 27)/2)$).

6. CONCLUSION

In this paper, we have developed a Hierarchical network topology suited for the tracking of miners and other assets in underground mines. The design depended on the apriori knowledge of the characteristics of the area of deployment. Further, this apriori knowledge was leveraged to develop a fault tolerant strategy. Our future work will look at developing a dynamic addressing scheme for real time network deployment. Further, the fault tolerance has to be incorporated in the initial network deployment scheme. On the practical side, we aim to setup a larger network of nodes and test the scalability of the proposed system.

7. REFERENCES

- [1] A Dunkels, J. Alonso, T. Voigt, Making TCP/IP viable for wireless sensor networks, European workshop on sensor networks, 2004
- [2] Akyildiz, I.F., Xudong Wang: A Survey on Wireless Mesh Networks, IEEE Communications Magazine (2005)
- [3] C.S. Raghavendra, A. Avizienis and M.D. Ercegovac, "Fault Tolerance in Binary Tree Architectures," IEEE Transactions on Computers, June 1984.
- [4] CC2430: focus.ti.com/analog/docs/gencontent.tsp?familyId=367&genContentId=24190

- [5] F. Koushanfar, M. Potkonjak and A. Sangiovanni-Vincentelli, "Fault Tolerance techniques for Wireless Ad Hoc Sensor Networks," 2002
- [6] G. Lewis et al, "A System for Simulation, Emulation, and Deployment of Heterogeneous Sensor Networks," SenSys '04.
- [7] Ho-In Jeon, Yeonsoo Kim: Efficient, Real-Time Short Address Allocations for USN Devices Using LAA (Last Address Assigned) Algorithm, 9th International Conference on Advanced Communication Theory (February 2007)
- [8] IEEE Std. 802.15.4-2003, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (2003)
- [9] ITU Report on Internet of Things – Executive Summary: www.itu.int/internetofthings/
- [10] J. L. Bredin et al, "Deploying Sensor Networks with Guaranteed Capacity and Fault Tolerance," MobiHoc '05.
- [11] Michael J. Hammons, Greg Chisholm: Enabling Total Asset Visibility, Defense Transportation Journal (August 2006)
- [12] Ran Peng, Sun Mao-heng, Zou You-min, ZigBee Routing Selection Strategy Based on Data Services and Energy-balanced ZigBee Routing, IEEE Asia-Pacific conference on Services Computing (December 2006)
- [13] TIMAC: <http://focus.ti.com/analog/docs/gencontent.tsp?familyId=367&genContentId=31261> (2007)
- [14] X. Han et al, "Fault Tolerant Relay Node Placement in Heterogenous Wireless Sensor Networks," InfoCom '07.
- [15] X. Luo, K. Zheng, Y. Pan, Z. Wu, A TCP/IP implementation for wireless sensor networks, IEEE Conference on Systems, Man and Cybernetics, 2004.
- [16] ZigBee Alliance: www.zigbee.org