Review of Beacon Collision Avoidance Methods in IEEE 802.15.4 Standard

Ankita Hatmode[1], Prof. P. H. Ghare[2]

Dept. of Electronics and Communication, Visvesvaraya National Institute of Technology, Nagpur, India

Abstract: *The IEEE 802.15.4 standard along with ZIGBEE specifications is used as communication standard for minimum power consumption, inexpensive networks such as LR-WPAN. The networks have different topologies depending on its applications such as Peer-to-Peer, Cluster tree, Mesh and STAR topology. In IEEE 802.15.4 standard for MAC layer , there are two modes of operations which are non-beacon mode and beacon mode. In beacon mode, PAN coordinators will send beacon signals to synchronize with their respective nodes, manage the network and the power of the transmitting systems and also perform the time synchronization of the data. Thus, if there are more than one PAN coordinators in the vicinity of each other, there is a probability of the collision of beacon signals from different PAN coordinators which may lead to loss of data. In order to handle this problem, various algorithms through different approaches are being developed but still the problem persists. In this paper, we provide the overview of various beacon collision avoidance algorithms which can reduce the beacon frame collisions and also supports Quality of Service issues.*

Keywords: *cluster tree network; beacon collision; scheduling; MAC Layer; LR-WPAN; IEEE 802.15.4.*

I. Introduction

The IEEE 802.15.4 standard [1] describes specifications for Physical Layer (PHY) & Medium Access Control (MAC) sublayer to support low data-rate wireless connectivity with fixed, portable and moving devices in a less complex network and operates for short-range radio frequency transmission space of about 10m. The main advantages of Low Rate-Wireless Personal Area Networks (LR-WPAN) are power-efficient network, Carrier Sense Multiple Access with collision avoidance (CSMA-CA) for selecting channel, fully recognized protocol for reliable transfer and flexibility of the network. Thus, the IEEE 802.15.4 standard is used as communication standard in LR-WPAN between its devices. The LR-WPAN comprises of following types of devices namely Full Function Devices (FFD) and Reduced Function Devices (RFD). The FFD functions as a coordinator or a device, provides synchronization services through periodic beacon transmissions. The personal area network (PAN) is controlled by PAN coordinator. The application decides the topology for interconnection of devices. The IEEE 802.15.4 standard supports STAR topology and Peer-to-Peer topology. In STAR topology, only one controller called PAN coordinator is being associated with the devices (FFDs or RFDs) within its radio-sphere through synchronization services. In Peer-to-Peer topology, the first device communicating on the channel is considered as PAN coordinator. In this topology, direct communication between devices can be established if they are within radio-sphere of each other and allow more complex network formation. A special case of Peer-to-Peer topology is the Cluster tree topology, which allows complex network formation with increased coverage area. The features of this topology are scalability, low power, low cost, and multi-hop communication. Due to these magnificent features, it is being used for wide range of applications such as health tracker, industrial surveillance and diverse industrial automation fields, emergency guiding [7] [8], object tracking [9], environmental monitoring [10] [11], etc. Fig.1 shows basic three level structure of Cluster tree topology and the corresponding level of each device is indicated by the number in the circle. In this topology, majority of devices are FFDs while RFDs connects at the end of any branch as a leaf node. The cluster head (CLH) starts broadcasting beacon frames to neighbouring devices when its cluster identity (CID) equal to zero. After receiving beacon frames, the devices may request to join the network at CLH. When CLH allows the device to link with the network, device will be added under CLH's neighbour list as a child node and newly connected device updates its neighbour list by adding CLH as a parent node. These newly added devices comprise of level 1 device as seen in Fig.1. The newly added devices further broadcast the beacon frames and add child nodes in its neighbour list. When more than one node at same level broadcasts beacon frames simultaneously, there arises a probability of beacon collision [2]. Another possibility of beacon collision is when nodes are in the radio-sphere of each other. Thus, the synchronization between parent node and child node is lost. If collision between beacon frames and data frames happens, there is a loss of data. Depending upon the different situations beacon collision can be classified into two categories namely direct beacon collision and indirect beacon collision. The details of beacon collision problem are discussed in section II. The section III discusses about the different algorithms developed to resolve beacon collision problem.
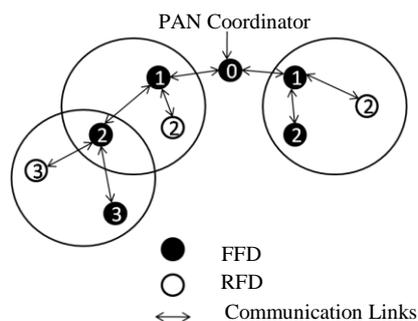
Fig.1 Cluster Tree topology

II. Beacon Collision Problem

The IEEE 802.15.4 standard [1] supported the formation of cluster tree networks but the proper functioning of such networks gets disturbed in a beacon enabled mode. There are two operating modes for data transfer in IEEE 802.15.4 standard:

- Non-Beacon enabled mode.
- Beacon enabled mode.

In a non-beacon enabled mode, device uses unslotted CSMA/CA mechanism to transmit the data frames to the coordinator and coordinator gives an acknowledgement on successful reception. While in beacon enabled mode, the beacon frames are broadcasted by coordinator prior to data frames and establishes synchronization with its neighbour devices. The communication sequence can be seen in the Fig.2.
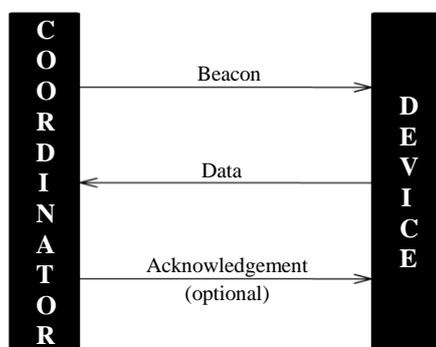

Fig.2 Communication to a coordinator in a beacon enabled network

The beacon frames are transmitted periodically at the start of each superframe structure where superframe structure consists of the superframe duration (active period) and beacon interval. We know that, there can be more than one FFDs or coordinators in the neighbourhood of each other in Cluster tree topology. As a result there will be collision between distinct beacon frames or between beacon frames and data frames, when two or more coordinators are broadcasting beacon frames simultaneously in the network. This results into critical problems in such applications where some of the intermediate devices are present between two or more coordinators. Thus, there can be two types of collision depending upon distance between two coordinators and nodes in the overlapping transmission ranges. They are as follows:

2.1 Direct beacon frame collisions

Direct beacon frame collisions happen when multiple coordinators are direct neighbours and are in the radio-sphere of each other. It is assumed that the node N is a child node of P1 and the coordinators P1 and P2 transmits respective beacon frames mostly at the same time, as shown in Fig.3. Thus, node N loses the synchronization between itself and parent P1 because of beacon frame collision [2].
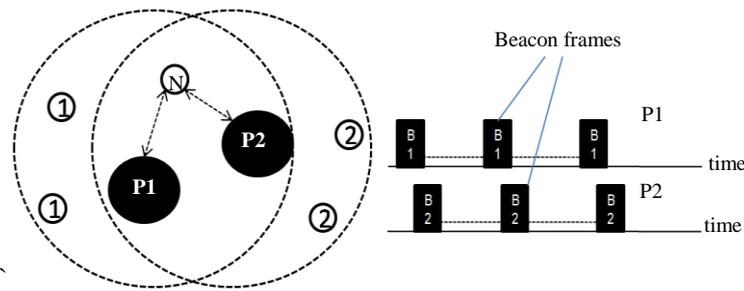

Fig.3 Direct beacon frame collision

2.2 Indirect beacon frame collisions

Indirect beacon frame collisions happen when multiple coordinators are not in radio-sphere of each other, but have coinciding transmission ranges. In Fig.4, node N is situated in the coinciding area of the transmission ranges of P1 and P2 and coordinators P1 and P2 transmits respective beacon frames mostly at the same time. Thus, node N will not receive the beacon frame correctly [2].
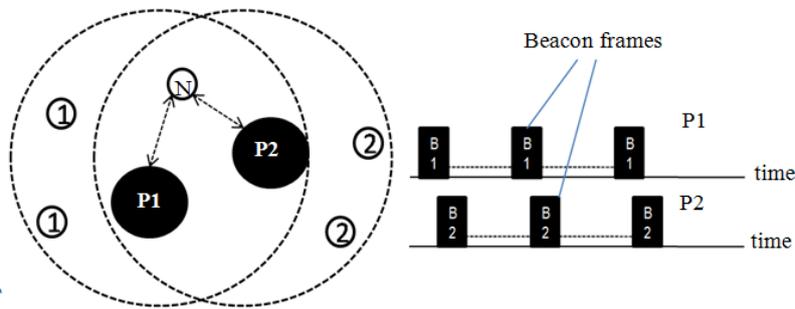

Fig.4 Indirect beacon frame collision

Thus it is required to broadcast the beacon frames in the organized manner to minimize beacon collision and efficient data transfer. Thus various algorithms are proposed by different authors to overcome beacon collision problem.

III. Algorithms

The IEEE 802.15.4 standard avoids data packets collision in beacon enabled mode using CSMA/CA technique, but still the problem of beacon frames collisions remains unsolved [1]. Thus, number of techniques is developed through different approaches like TDMA based scheduling [2] [12], Channel based scheduling [14], Graph theory methods [5], energy detection based techniques [4], etc. This paper gives an overview of some of these algorithms in brief. Fig.5 shows the categorization of the algorithms based on their approaches.
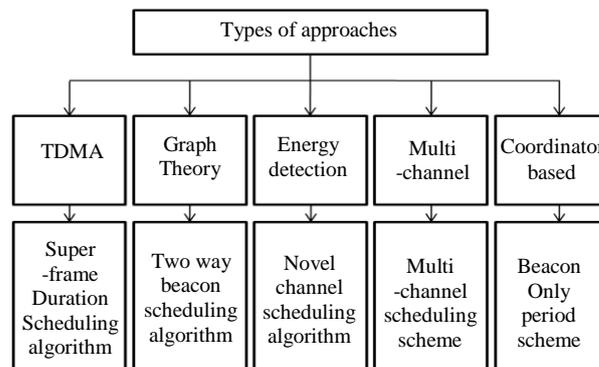

Fig.5 Types of approaches

3.1 Superframe Duration Scheduling (SDS) algorithm for the time division approach

A scheduling is the most basic approach to solve this problem. Let us consider a scenario having multiple coordinators with their respective superframe duration (SD) and beacon interval (BI). This algorithm is inspired by the pinwheel problem [3] and scheduling of coordinators is done using a simple greedy algorithm. The following are the steps to implement SDS algorithm.

Algorithm

Let there are m coordinators with A = { BI_i , SD_i | $1 \leq i \leq m$ } is a set of beacon intervals arranged in increasing order of BI and decreasing order of SD in case of equal BI.

$$DC = \sum_{i=1}^{m} DC_i = \sum_{i=1}^{m} \frac{SD_i}{BI_i} \leq 1$$

if (DC < 1)

        display("Coordinators are Schedulable");

else

        display("Coordinators are not Schedulable");

endif

$$BI_{max} = LCM \left\{ 2^{BO_1}, 2^{BO_2}, \ldots 2^{BO_m} \right\}$$

$$BI_{min} = \min \left\{ 2^{BO_1}, 2^{BO_2}, \ldots 2^{BO_m} \right\}$$

Slot = Empty slotted time line of duration equal to BI_{max}

One slot duration = minimum SD i.e. for SO = 0

SD_{min} =15.36ms * 2^0 = 15.36ms

for (each coordinator in set A)

    do{

        Look for the first available empty slots of length equal to SD_i ;

        Slot(from first available SD_i empty slots) = coordinator id i ;

        Repeat after every BI_i ;

        If (No empty slots of length SD_i found after BI_i)

           return("Coordinators are not Schedulable")

        else

           Slot(from first available SD_i empty slots) = coordinator id i ;

        endif

    }

Endfor

3.2 Multichannel Scheduling Scheme (MSS)

The proposed MSS algorithm [14] allows the scheduling of superframe of multiple clusters on different radio channels. This algorithm overcomes the drawbacks of time division approaches which are SDS algorithm [2] and Superframe Duration Scheduling with coordinator grouping [2]. The detailed algorithm steps are discussed as follows:

Algorithm

Let there exists a cluster tree network consisting of n PAN coordinators with

A = { BI_i , SD_i | $1 \leq i \leq n$ } is a set of their respective BI and SD.

$$BI_{max} = LCM \left\{ 2^{BO_1}, 2^{BO_2}, \ldots 2^{BO_n} \right\} \qquad \text{… major cycle period}$$

$$BI_{min} = \min \left\{ 2^{BO_1}, 2^{BO_2}, \ldots 2^{BO_n} \right\} \qquad \text{… minor cycle period}$$

The cluster is divided into two groups.

group1 = {PAN coordinators and all clusters at even depth of tree are scheduled at

        time zero for SD_i time length and repeated after every BI_i }.

group2 = {all clusters at odd depth of tree ordered in the increasing order of BI and

        decreasing order of SD in case of equal BI }.

for (each coordinator in group1)

    do{

        Search for empty slots of length equal to its SD_i starting from time zero ;

```
            if (found empty slots of length SD_i)
                slot = coordinator id ;
                Free channel selected using channel selection strategy() ;
            else
                 return(not schedulable) ;
            endif
            repeat after every BI_i till remaining empty slots in minor cycle is lesser than
            SD_i;
        }
endfor
for ( each coordinator in group2)
    do{
            In each minor cycle: interval between first and second time slice T_i =
            largest SD .
```

The starting offset of group2 clusters = $t_i^{start} = \max_{j \in MC_i}(T_j)$

```
            Search for empty slots of length equal to its SD_i starting from starting offset
            of first minor cycle;
            if (found empty slots of length SD_i)
                slot = coordinator id ;
                Free channel selected using channel selection strategy() ;
            else
                 return(algorithm not feasible)
            endif
            repeat after every BI_i till remaining empty slots in minor cycle is lesser than
            SD_i;
        }
Endfor


channel selection strategy()
{
  Number of available distinct radio channels = 16 ;
  for ( each superframe )
  do{
  if (already allocated superframes do not overlap wholly or partly with current
      superframe)
          return (the first free channel is selected);
  else
              check if there are other superframes on the same channel say C_k far
              enough to schedule the current superframe;
              if(an appropriate superframe is found)
                     return(the channel C_k for this superframe) ;
              else
                  return(choose a new channel);
                endif
  endif
  }
 endfor
}
```

Note: The algorithm first selects the even channels and if no even channels are found then it selects the odd channels to limit cross-channel interference and no adjacent channels are used unless required.

3.3 Novel Channel Scheduling Algorithm

A scheduling with energy detection (ED) scan which is proposed in [4] can be considered as more effective than just simple scheduling as it considers channel parameters also. The parameters threshold N_{loss} and packet error rate (PER) are used to detect beacon losses.

Algorithm

Let us consider a scenario with multiple WPANs.
Tasks performed by PAN coordinator()
{
  active scan → choose PAN identifier for new PAN ;
  ED scan   → selection of clear channel ;
}
if (clear channels found)
        start(periodic beacon frames transmission);
else
        return(detected interference from other WPAN);
endif
scans for other candidate channel during inactive period using Clean channel searching scan(CCS) ;
if(number of successively lost beacon frames > Nloss)
  return(current channel interference is affected)
else
  DeviceCalculates(PER of beacon reception);
  if(PER > PER_threshold)
    return(interference has occurred);
    the device changes its channel to the candidate channel.
  endif
endif

3.4 Two-way beacon scheduling (TBS)
        Till now the algorithms were proposed only for communication from coordinators to nodes. But this proposed centralized algorithm [5] considers the two-way communication and modifies the original superframe structure to achieve minimum latency for broadcast and convergecast communication and then schedules the nodes by traversing through them in bottom-up fashion in accordance to their depths in the tree.
        Let us consider a network graph $H = (N, E)$, where H's interference graph $H_I = (N, E_I)$, a ZigBee tree $Z = (N_T, E_T)$ has been constructed where $N_T \in N$ and $E_T \subseteq E$, and $k = 2(BO\text{-}SO)$ slots are available,

        TBS problem is to find for each node $i \in N$ a non-interfering slot assignment $s^u(i)$ and $s^d(i)$ for TbUp and TbDn respectively and $s^u(par(i))$ and $s^d(par(i))$ for RbUp and RbDn respectively such that network latency $L(T)$ is minimized where $par(i)$ is parent of i.
        Inspired by the Brooke's theorem [6] which proves that m colors are sufficient to color any graph with a maximum degree of m, we assume that

$$k \geq 2 \times D_I \qquad\qquad (1)$$

where D_I is the maximum degree of H_I .

Algorithm

//To find Upslot(n)
for(each node *n* at depth d being visited while traversing)
  do{
      sort nodes in a descending order as per their degrees in H_I and then
    serially traverse each vertex for finding slots for them;
    if (*n* is a leaf node)
        $t_u(n) = 0$ //$t_u(n)$ is a temporary variable used to calculate final $s^u(n)$ //
    else
        $t_u(n) = \max\{s^u(n') \mid n' \text{ is a child of } n\} + 1$
    endif
  }
endfor

X(n) = { n′ | n′ having interference with node n & received slots $s^u(n') \neq$ NULL };

---

jump : temp $= t_u(n) \bmod k$ // used to find feasible slots for node n //

if(direct interference && $s^u(n')$ or $s^d(n')$ = = temp)

       return(slot temp is not feasible);

elseif(indirect interference && $s^u(n')$ or $s^d(n')$ = = temp)

       call(RemoveIndInt(n′,n));

       if(interference is removed)

           return(slot temp is feasible for n);

           $s^u(n)$ = temp;

       else

           $t_u(n) = t_u(n) + 1$;

       go to *jump* ;

       endif

endif

//To find DnSlot(n)

for(each node *n* at depth d being visited while traversing)

   do{

       sort nodes in a descending order in accordance to their degrees in H_I and then

      serially traverse each vertex for finding slots for them;

      if (*n* is a leaf node)

         $t_d(n) = k - 1$;   //$t_d(n)$ is a temporary variable used to calculate final $s^d(n)$ //

      else

         $t_d(n) = \min\{ s^d(n') \,|\, n' \text{ is a child of } n \} - 1$;

      endif

     }

endfor

X(n) = { n′ | n′ having interference with node n & received slots $s^d(n') \neq$ NULL };

jump : temp $= t_d(n) \bmod k$ // used to find feasible slots for node n //

if(direct interference && $s^u(n')$ or $s^d(n')$ = = temp)

       return(slot temp is not feasible);

elseif(indirect interference && $s^u(n')$ or $s^d(n')$ = = temp)

       call(RemoveIndInt(n′,n));

       if(interference is removed)

           return(slot temp is feasible for n);

           $s^d(n)$ = temp;

       else

           $t_d(n) = t_d(n) - 1$;

       go to *jump* ;

       endif

endif

RemoveIndInt(n', n)

{

 //assume y is common neighbour of n′ and n//

 if(y is an end device)

       return(associate y with new router w);

      if(slots $s^u(w)$ and $s^d(w)$ exists for w)

          check these slots do not cause new interference;

          if(new interference is resolvable)

              return(y can be successfully associated with w);

              return(positive response);

         else

             return(search for another router);

             return(negative response);

```
                endif
            endif
    elseif(y is a router)
        return(no change required);
    endif
}
```

3.5 Beacon Only Period (BOP) scheme

This scheme [2] is proposed based on new superframe structure which will start with a period reserved for beacon frame transmissions. During this period, each coordinator selects a Contention Free Time Slot (CFTS) to send its beacon frames avoiding collisions with beacon frames from other coordinators.

Let us consider a cluster tree ZigBee network having N coordinators with Superframe duration SD_i, Beacon interval BI_i and beacon order BO_i for ith coordinator. The 15.4b task group [13] suggested following two rules for CFTS allocation.

Rule 1: The CTFSs of the coordinator C_i and its parent must be distinguishable.

Rule 2: The CTFSs of the coordinator C_i and the parents of its neighbors must be different. Here, neighbor means only coordinators are considered, simple node neighbors are excluded.

The above two rules could not eliminate the beacon collision problems completely in case of simple node neighbors and direct beacon collision problem. Thus, the rules are further modified and stated as follows:

Rule 3: The CTFSs of the coordinator C_i and the parents of all its neighbors(coordinators + simple nodes) must be different.

Rule 4: The CTFSs of the coordinator C_i and its neighbor coordinators must be different.

The rules 3 and 4 are helpful to entirely remove direct beacon collision in non parent-to-child situation as well. These modified rules do not allow the neighbor coordinators to have CFTSs and all the parent coordinators are allocated CFTS before child nodes. This will protect the nodes at another depth from having same CFTS due to rule 3. The length of beacon-only period is computed dynamically when a node connects or leaves a network and allocates the CFTS according to rule 3 and rule 4.

The main contribution of this paper is to explain the algorithms based on different approaches to successfully avoid interferences in the network. They have been simulated by their respective researchers and based on their result certain merits and demerits are found out. The comparative study of their merits and demerits are shown in Table.1 below.

Table 1. Advantages and disadvantages of algorithms

ALGORITHM	ADVANTAGES	DISADVANTAGES
SDS	-Practically, this algorithm can be implemented with ease through minor changes in the IEEE 802.15.4. -If a new coordinator comes in the scenario and schedulability check test returns "schedulable" then all the coordinators set their new_beacon_offset and send its beacon frame according to this offset. -Avoids both direct and indirect beacon collisions. -Maximizes number of cluster.	-There can be unused empty slots even after whole scheduling is done which may leave some part of bandwidth unused and also limit the scalability of network. -The channel quality parameters are not considered. Thus, it may lead to retransmissions of beacon and data frames in case of failure and loss of energy.
MSS	-Provides higher scalability. -Supports Contention-free transmission in the GTS with minimum interference from different clusters.	-The farthest minor cycle if unused then it is wastage of bandwidth space. -Not all the set of superframes are schedulable using MSS algorithm; the exceptions are discussed in detail in [14].
Novel Channel Scheduling	-Minimizes recovery time after a collision occurs and identifies a channel to switch. -Avoids beacon collision between IEEE 802.15.4 devices as well as with network devices also like 802.11.	-If no idle channel is found, latency may increase.
TBS	-The overall network latency is minimized. -The number of interfering neighbors is effectively reduced.	-This algorithm is not applicable to distributed network topology.
Beacon Only Period scheme	-The direct communication between neighbor nodes is possible if active periods of different clusters begin simultaneously. -No limitation on the duty cycle value.	-The duration of BOP increases as the number of CFTS rises with increasing number of coordinators. -Complex to implement for real-time applications.

IV. Conclusion

This paper has considered different approaches to alleviate beacon collision problem proposed by various researchers. We studied different algorithms, their advantages and disadvantages. It can be observed that TBS algorithm is highly effective to minimize the overall latency of network, SDS algorithm and MSS algorithm supports high scalability in comparison with other algorithms. The algorithms discussed in this paper are analyzed based on simulation done by respective researchers and can be considered for hardware implementation for future work.

References

[1] IEEE Std 802.15.4™-2003, IEEE Standard for Information technology,Telecommunications and information exchange between systems,Local and metropolitan area networks Specific requirements

[2] Anis Koubâa, Mario Alves, Melek Attia, Anneleen Van Nieuwenhuyse, Collision-Free Beacon Scheduling Mechanisms for IEEE 802.15.4/Zigbee Cluster-Tree Wireless Sensor Networks

[3] Robert Holte, AI Mok, Louis Rosier, Igor Tuichinsky, and Donald Varve1, The Pinwheel:A Real-Time Scheduling Problem

[4] Jin-Woo Kim, Yeonwoo Lee and Seong Ro Lee, Interference Aware MAC scheduling for Collision Avoidance using Energy Detection Scan, *Vol. 7, No. 3, May, 2013, International Journal of Software Engineering and Its Applications*

[5] Lun-Wu Yeh, Meng-Shiuan Pan, and Yu-Chee Tseng, Two-Way Beacon Scheduling in ZigBee Tree-BasedWireless Sensor Networks, *2008, IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing.*

[6] D. B. West. *Introduction to Graph Theory* (Prentice Hall 2001)

[7] M. A. Batalin, G. S. Sukhatme, and M. Hattig, Mobile robot navigation using a sensor network, *In IEEE International Conference on Robotics and Automation, 2004.*

[8] M.-S. Pan, C.-H. Tsai, and Y.-C. Tseng. Emergency guiding and monitoring applications in indoor 3d environments by wireless sensor networks. *1(1/2):2–10, 2006.International Journal of Sensor Networks (IJSNet)*

[9] C.-Y. Lin, W.-C. Peng, and Y.-C. Tseng. Efficient innetwork moving object tracking in wireless sensor networks. *IEEE Trans. on Mobile Computing, 5(8):1044–56, 2006.*

[10] Terrestrial ecology observing systems. *http://research.cens.ucla.edu/areas/2007/Terrestrial/.*

[11] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler. An analysis of a large scale habitat monitoring application. *In Proc. of ACM Int'l Conference on Embedded Networked Sensor Systems (SenSys), 2004.*

[12] Chi-Ming Wong, A Novel Beacon Frame Scheduling Algorithm Based on Cluster-Tree IEEE 802.15.4 Wireless Sensor Networks, *Proceedings of the 2012 IEEE ICCS*

[13] T. G. 15.4b: *http://grouper.ieee.org/groups/802/15/pub/TG4b.html.*

[14] Emanuele Toscano, Lucia Lo Bello, A Multichannel Approach to Avoid Beacon Collisions in IEEE 802.15.4 Cluster-Tree Industrial Networks, *978-1-4244-2728-4/09/$25.00 ©2009 IEEE*