Enabling IEEE 802.15.4 Cluster-Tree Topologies in OMNeT++

J. Hurtado-López E. Casilari A. Ariza-Quintana
Dpto. Tecnología Electrónica, University of Málaga
Campus de Teatinos, 29071 Málaga (Spain),
Tfno.: 34-952132755; FAX 34-952131447
ecasilari@uma.es

ABSTRACT
The IEEE 802.15.4 standard jointly with the ZigBee specification is becoming one of the most popular technologies for the development of Low-Rate Wireless Personal Area Networks (LR-WPANs). 802.15.4 Cluster-Tree topologies appear as an appealing and standard compliant solution for many applications of home automation, medical sensing, tele-monitoring, etc. In this paper we present the enhancements made to the current IEEE 802.15.4 model in OMNeT++ to support this topology as well as different algorithms to schedule and dimension the Superframe Durations in order to avoid Beacon collisions in the network. An intuitive visual idea and a check of the whole process can be performed with OMNeT++'s Plove tool.

Categories and Subject Descriptors
I.6.5 [**Simulation and Modeling**]: Model development

C.2.1 [**Computer Communication Networks**]: Network architecture and design

General Terms
Performance, Design

Keywords
OMNeT++, Zigbee, 802.15.4, MAC, beacon.

1. INTRODUCTION
IEEE 802.15.4 standard [1] defines Physical (PHY) and Medium Access Control (MAC) layers while the ZigBee specification [2] completes the protocol stack designed to satisfy the market's need for low cost, low rate, and energy efficient wireless embedded devices. The Physical Layer operates in the Industrial Scientific and Medical (ISM) radio bands with a transfer rate of 250 kbps at 2.4 GHz (with 16 available channels) or 20/40 kbps at 868/915 MHz (10/1 channels). The standard also defines Guaranteed Time Slots (GTS) to provide quality of service to real time flows.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
OMNeT++ 2009, Rome, Italy.
Copyright 2009 ICST, ISBN 978-963-9799-45-5.

There are two possible modes of operation for the MAC sublayer: (1) the nonbeacon-enabled mode or point to point in which unslotted CSMA/CA is used to communicate, (2) the beacon-enabled mode which uses slotted CSMA/CA and communications are synchronized through special frames, periodically emitted by specific nodes (the coordinators), called Beacons. In the first mode, all nodes must be continuously listening to the radio channel, which leads to a useless waste of energy. As an advantage, this nonbeacon-enabled mode does not present any scalability problem as it allows nodes to transmit at any moment (so, obviously, GTS is not possible). In the second mode, a node should be active only to receive the Beacon from its coordinator in order to keep synchronized with the network and during the Contention Access period (CAP), i.e. a special period (defined by the coordinator) just after the Beacon, when data transmissions take place. The rest of the time, between these periods, the nodes can turn into a low consumption state reducing their duty cycle and consequently saving battery. Additionally GTS slots are allowed. On the other hand, this mode of operation is more complex and needs specific algorithms to correctly design the different parameters that regulate Beacon and data transmission in order to achieve a good network capacity. Furthermore, if no specific scheduling mechanism is adopted for the Beacon transmissions of the different coordinators, collisions may occur. This is a great challenge for the so-called Cluster-Tree topology. Cluster-Trees are a special case of peer to peer network in which a device is allowed to communicate only with its 'parent' (coordinator) or 'children' (coordinated) nodes. IEEE 802.15.4 and Zigbee admit the formation of the Cluster-Trees but the detailed analysis of the characteristics of this kind of topology is missing in the literature.

This paper presents the modifications made to the existing 802.15.4 model in OMNeT++ [3] to implement several algorithms that organize the activity periods of the coordinators in an IEEE 802.15.4 Cluster-Tree network. The performed simulations show that a wrong election of the beacon-enabled mode parameters may severely affect the global network behavior.

2. BEACONS MANAGEMENT IN CLUSTER-TREE NETWORKS
Two types of nodes are defined by the IEEE 802.15.4 standard: the Full-Function Devices (FFD) and the Reduced-Function Devices (RFD). An RFD can only talk to an FFD node, i.e. it always behave as a simple device or a leaf node, while an FFD can talk to any other device and may act as the PAN Coordinator, a coordinator or a leaf node. A coordinator is in charge of the communications of the set of leaf nodes associated with it and, in

the beacon-enabled mode, it is responsible for transmitting Beacons so that the depending devices can synchronize. The time between two consecutive Beacons of the same coordinator is called the Beacon Interval (*BI*) and its structure is called the Superframe (see Figure 1). A Superframe, which is bounded by the transmission of a Beacon frame, has both an active and an inactive period. The coordinator may enter a low-power (sleep) mode during the inactive period to achieve a better energy efficiency. The structure of the Superframe is described by the values of macBeaconOrder (*BO*) and macSuperframeOrder (*SO*). *SO* describes the length of the active portion of the Superframe or Superframe Duration (*SD*), which includes the Beacon frame. *BO*, *BI* and *SO*, *SD* are related as follows:

$BI = a \cdot 2^{BO}$ for $0 \leq BO \leq 14$ *(1)*

$SD = a \cdot 2^{SO}$ for $0 \leq SO \leq BO$ (2)

where *a* is the Base Superframe Duration (15.36, 24 or 48 ms depending on the employed bit rate 250, 40 or 20 kbps respectively). Finally, the *SD* is divided into sixteen equally sized slots that can be separated into two different periods: The Contention Access Period (CAP) and the Contention Free Period (CFP). During the first one, a node shall contend for the slots by using the CSMA/CA mechanism. Although the CSMA/CA algorithm tries to avoid packet collisions they may still occur. These collisions together with the backoff periods introduced by CSMA/CA may induce delays or even packet losses; anyway they may cause the underutilization of the CAP slots. To guarantee some quality of service, up to seven slots (GTS) can be reserved forming the CFP.

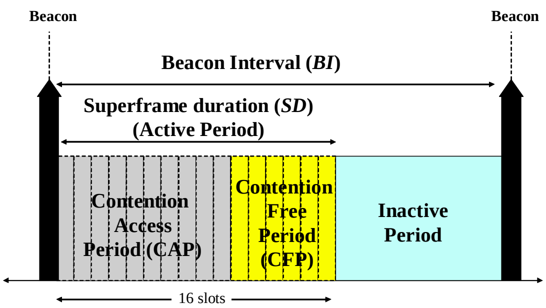

Figure 1. Example of the IEEE 802.15.4 Superframe structure.

Clearly, most of the advantages of IEEE 802.15.4 strongly depend on an adequate choice of the MAC parameters; the behavior of the whole network relies on this. For example, if *SO*=*BO* there will be no inactive period in the Superframe meaning that the nodes could not enter into the low power state and, on the other hand, if *SO* is set too low (and so does the duty cycle), the data rate has to be decreased.

As outlined before, 802.15.4/ZigBee allows the formation of Cluster-Tree topologies. In that case, one of the coordinators must become the PAN Coordinator. This entity, called the ZigBee Coordinator (*ZC*) in the ZigBee specification, must be unique in the whole network. The other coordinators or ZigBee Routers (*ZR)* will manage the communications and synchronization of their associated leaf nodes (following a star topology). Both IEEE 802.15.4 standard and ZigBee specification propose the Cluster-Tree topology concept but none of them impose any algorithm or protocol to create or organize it. If self organization is one of the main attractiveness of IEEE 802.15.4/ZigBee capabilities it is also one of its greatest challenges. In fact, most of the existing commercial 802.15.4/ZigBee-compliant modules do not support the formation of Cluster-Tree topologies. Actually, many of them only implement the Physical Layer of the stack (MICAz or TelosB [4]). Therefore, this is an open research issue still far from being solved.

Maybe one of the most important problems related to Cluster-Tree topologies derives from the coexistence of different coordinators. If a coordinator's Beacon collides with another Beacon or data packet, the synchronism of the nodes associated to that coordinator could be lost. Bearing in mind that the Cluster-Tree topology can be considered as the one formed by the association of beaconing coordinators of different star networks it can be inferred that there is a high probability of Beacon collision if no special mechanism is implemented to avoid it. This problem was studied by the IEEE 802.15 Task Group 4b [5] which discussed several strategies to cope with it [6]. Finally, the solution that they found consists in the possibility of shifting the beaconing time of each coordinator in order to avoid collisions. However the way to determine the values for the Beaconing times is still missing. On the other hand, as the activity of a coordinator obliges its neighbors to stay inactive, scalability may become a serious problem if the active periods of the coordinators are defined in a very restrictive way.

In the next section we propose several algorithms (also presented in [7]) to determine the superframe duration of each coordinator. The general goal of the algorithms is to optimize the utilization of the network beacon interval (*BI*).

3. ALGORITHMS FOR DEFINING THE SUPERFRAME DURATIONS

In our study we assume that any coordinator (or any node) can interfere with the rest so a Beacon shall not be transmitted during the *SD* of any other coordinator. In other words, different *SDs* cannot overlap in time. Our goal is to maximize the use of the *BI* of the network. Only leaf nodes generate traffic and the sink will always be the PAN Coordinator. In order to set an upper bound to the delay the *BI* is fixed. The algorithms must determine the *SOs* and offsets of each coordinator as seen on Figure 2.

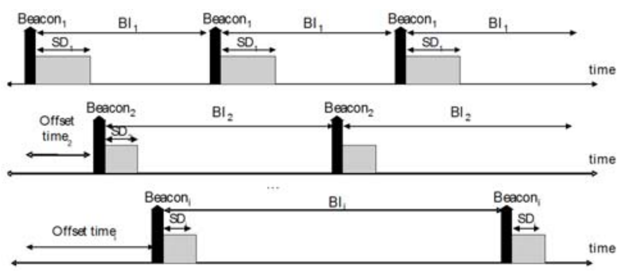

Figure 2. Collision avoidance by distributing the *SDs*.

Following this *BI* distribution technique only the children of one coordinator will contend during their father's CAP. Clearly, the larger the coordinator's CAP the higher the traffic load it can support. So, the main objective is to fit as best as possible the *SD* of each coordinator to the traffic load it is supporting.

The following non-strict inequality must always be guaranteed:

$$BI = a \cdot 2^{BO} \geq \sum_{i=1}^{N_C} SD_i = \sum_{i=1}^{N_C} a \cdot 2^{SO_i} \tag{3}$$

being N_c the number of coordinators present in the network (including the PAN coordinator) and SD_i, SO_i the Superframe Duration and Superframe Order of the *i*-th node, respectively. If equation (3) can be satisfied, it would imply that a correct distribution of the *SDs* with no overlapping is possible; if not, one or more *SOs* have to be reduced or *BO* increased. If this situation occurs when *BO* is 14 and all *SO* values are zero then the only left possibility is to drop some coordinator(s) from the network.

3.1 Equidistribution of the Beacon Interval

This is the easiest solution to the distribution of the *BI*. In this case we set all *SOs* to the same value that can be derived from equation (3) obtaining finally:

$$SO = \left\lfloor \log_2\left(\frac{2^{BO}}{N_C}\right)\right\rfloor = \left\lfloor BO - \log_2\left(N_C\right)\right\rfloor \tag{4}$$

3.2 PAN coordinator prioritization

Many practical applications, for example, Wireless Sensor Networks (WSNs) consist of a group of nodes that collects data and send them to a central processing node. In this kind of situations we can slightly modify the previous policy in order to privilege the role of the PAN Coordinator (which will receive all the packets generated in the tree) by increasing its *SD*. As a first option, the *SO* of the PAN coordinator is set to twice the value of that of the rest of coordinators:

$$SO_i = SO \quad \forall i \in [2, N_C]; \quad SO_1 = 2 \cdot SO \tag{5}$$

Substituting in (3) and after some calculus we obtain:

$$SO = \left\lfloor \log_2\left(1 - N_C + \sqrt{(N_C - 1)^2 + 4 \cdot 2^{BO}}\right) - 1\right\rfloor \tag{6}$$

As *SD* depends on a power of *SO* (following equation (2)), the above policy overestimates the *SD* of the PAN Coordinator in most cases. Therefore, as a second option we propose:

$$SO_i = SO \quad \forall i \in [2, N_C]; \quad SO_1 = SO + 1 \tag{7}$$

Using these new values in equation (3) we obtain:

$$SO = \left\lfloor \log_2\left(\frac{2^{BO}}{N_C + 1}\right)\right\rfloor = \left\lfloor BO - \log_2\left(N_C + 1\right)\right\rfloor \tag{8}$$

3.3 Beacon Interval distribution based on topology

The previous policies were rough solutions to the problem of distributing the *BI.* There are many applications where each coordinator has to support a different number of nodes (or traffic load). Next, we propose a more flexible strategy allowing the design of each *SD* for each coordinator as a function of the traffic load it is expected to support. Figure 3. shows the flow diagram of the algorithm (see [7] for more details):

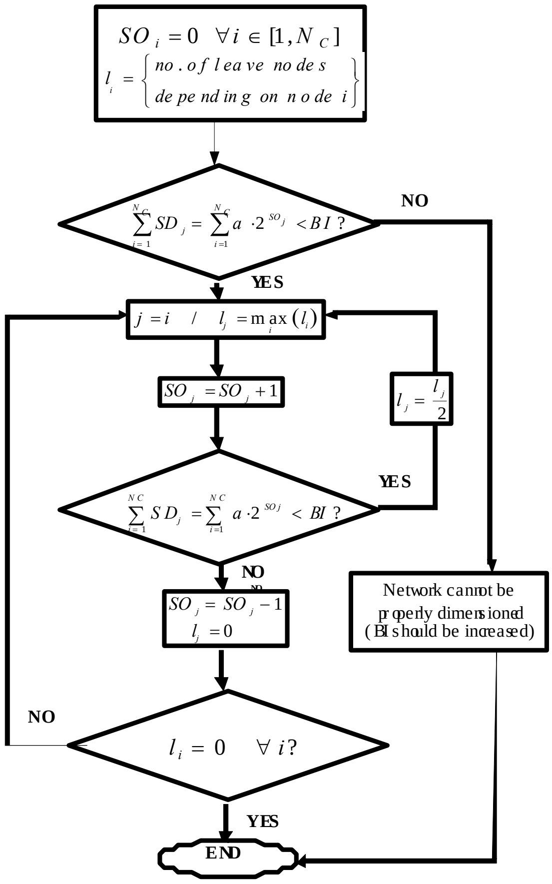

Figure 3. The Beacon Interval distribution based on topology algorithm flow diagram

It has been assumed that all data sources (i.e. the leaf nodes) do generate the same traffic. The assignation of the *SOs* is made iteratively so that coordinators with the higher number of children are favored. This policy adjusts better to diverse traffic loads and topologies than the previous ones.

4. IMPLEMENTATION'S FUNCTIONAL DETAILS

In order to implement the aforementioned policies for 802.15.4 Cluster-Trees, we extended the capabilities of the existing IEEE 802.15.4 model in OMNeT++ [8]. This model was adapted from a version for ns- 2 by Feng Chen and Falko Dressler in the University of Erlangen-Nürnberg (Germany).

The model, which is described in more detail in [9] implements the IEEE 802.15.4 protocol stack (version of 2006). It includes a Routing Module that permits to configure star topologies. To support Cluster-Tree topologies, our extension to the model defines the role of the ZigBee Coordinator (ZC) as a special class of FFD node acting in the beacon-enabled mode.

In a real scenario, the ZC is supposed to be in charge of programming and communicating the MAC parameters that regulates the Beacon emission of the ZigBee Routers (ZRs), mainly the Beacon Order (*BO*) of the whole network and the Superframe Orders (*SOs*) of the routers [2]. To simplify this procedure in our implementation, the configuration of these MAC parameters is defined through the file *omnetpp.ini*.

In this file, following the structure of the original code, a value for the *BO* and the *SO* is defined for every node. The role of the FFDs (ZRs and the ZC) is assigned by fixing a value to *BO* different from 15. On the contrary, the RFDs (the final devices that do not emit Beacons) are identified by setting the *BO* to 15. A special Boolean field is added to select the ZC among all the ZRs (just one node can behave as the ZC, in other case the simulation will not work properly). The ZC initiates the Beacon emission as soon as the simulation begins. Conversely, ZRs are programmed so that they only emit their Beacons after associating to the ZC. After this association, the only functional difference between the ZC and the ZRs is that a ZR alternatively acts as an associated node (during the active period of the Superframe of the ZC) and as the coordinator of a set of surrounding RFD nodes (see the example in Figure 4).

Our implementation is specifically intended for topologies supporting upstream traffic, that is to say, traffic flowing from the final RFD leaf nodes to the ZC (typical case of a sensor network). If so, ZRs receive and buffer the packets coming from the RFDs until they can be retransmitted to the ZC during the active period of the ZigBee Coordinator.

To prevent the overlapping of the active parts of the Superframes of the ZRs and the ZC, the emission of the Beacons (and consequently the distribution of the time between the Superframes) are governed by an offset time. The offset for each coordinator is set in a special *StartTime* parameter. The particular value of the offset (which is null for the ZC) for each ZR depends on the utilised policy to coordinate the Superframe Durations (the policy is selected in the program configuration file). The different proposed policies are implemented in a single routine which is executed at the beginning of the simulation. Basing on the elected algorithm, the *BO* of the network (that of the ZC) and the number of ZRs and leaf nodes associated to each ZR, the routine calculates the values of the *StartTime* and *SO* for the ZRs. These parameters resulting from the application of this routine can be optionally shown in the screen or saved in a file.

5. SIMULATION EXAMPLE

Figure 4 shows an example of a 802.15.4 Cluster-Tree network configured with our architecture. The network consists of 1 ZC (host[0]) and 3 ZRs (host[1], host[2] and host[3]). The three ZRs have 2, 4 and 1 associated final leaf nodes, respectively. The *BI* of the network was set to 5 while the radio band is 2.4 Ghz. For this scenario, the results of the different proposed policies for the dimensioning of the Superframe Durations are those tabulated in Table 1. These values are also stored in a file which is compatible with Scalars tool of OMNeT++.

To illustrate that Beacon emissions follow the configured values of these parameters, Figure 5 represents the network activity (captured with OMNeT++ Plove tool) when the topology based algorithm is employed. In the figure, a value of 2 corresponds to a Beacon emission (i.e., the beginning of the CAP of a coordinator) while the intervals with a constant value of 1 represents the different Superframe Durations. From the figure, we can observe that the activity periods of the Superframes do not overlap while the Superframe Durations follow the calculated offsets (the values defined in the last column of Table 1).

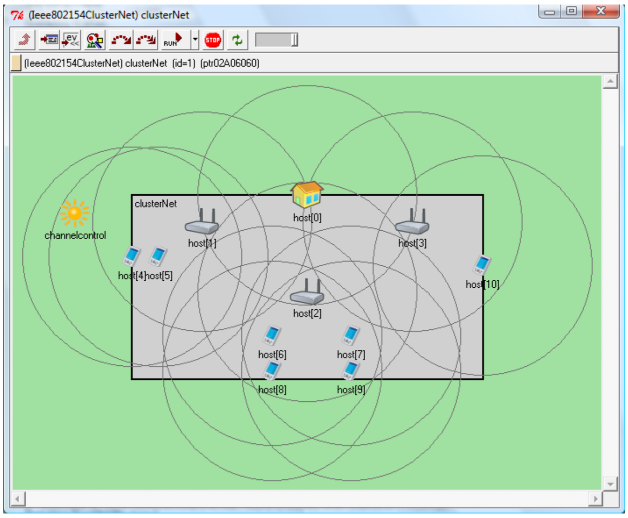

Figure 4. Example of Cluster-Tree topology

Table 1. Resulting *SOs* and *StartTimes*

Node	Applied Policy				
	Parameter	**Same SO**	**$SO_0 = 2 \cdot SO_i$**	**$SO_0 = SO_i + 1$**	**Based on topology**
ZC host[0]	***SO***	**3**	**4**	**3**	**4**
	StartTime	**0**	**0**	**0**	**0**
host[1]	***SO***	**3**	**2**	**2**	**2**
	StartTime	**0.123 s**	**0.246 s**	**0.123 s**	**0.246 s**
host[2]	***SO***	**3**	**2**	**2**	**3**
	StartTime	**0.246 s**	**0.307 s**	**0.185 s**	**0.307 s**
host[3]	***SO***	**3**	**2**	**2**	**2**
	StartTime	**0.369 s**	**0.369 s**	**0.246 s**	**0.430273**

As it refers to the performance of the simulator, just 70 seconds are required to simulate 1800 seconds in the previous scenario with a traffic load of 10 packets (of 102 bytes) per second and per final leaf node.

6. CONCLUSIONS

IEEE 802.15.4/ZigBee standard is a promising technology that enables the creation of low cost and low consumption multi-hop Cluster Trees. An efficient utilization of networks based on this technology may require the definition of algorithms that optimize the activity period of the so called coordinators (in charge of timing and centralizing the communications of a set of nodes by means of special Beacon messages).

This paper has described the implementation on OMNeT++ of several policies aiming at coordinating the beacon emission of the coordinators in 802.15.4 Cluster-Trees. The policies have been programmed as an extension of the capabilities of an existing 802.15.4 model for OMNeT++.

7. ACKNOWLEDGMENTS

This work was partially supported with public funds by the Spanish National Project No.TEC2006-12211-C02-01.

8. REFERENCES

[1] IEEE TG 15.4, "Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)", 2006, IEEE Standard for Information technology, URL: http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf

[2] ZigBee-Alliance, "ZigBee Specification", URL: http://www.zigbee.org/

[3] OMNeT++ homepage, URL: http://www.omnetpp.org

[4] Hardware platforms available for TinyOS, URL: http://www.tinyos.net/

[5] IEEE 802.15 Task 4b, URL: http://grouper.ieee.org/groups/802/15/pub/TG4b.html

[6] A. Koubâa, A. Cunha and M. Alves, "A Time Division Beacon Scheduling Mechanism for IEEE 802.15.4/Zigbee Cluster-Tree Wireless Sensor Networks", in Proceedings of the 19th Euromicro Conference on Real-Time Systems (ECRTS 2007), Pisa (Italy), July 2007, pp. 125-135.

[7] E. Casilari, A. Flórez-Lara, J.M. Cano-García, "Analysis of the scalability of hierarchical IEEE 802.15.4/Zigbee networks", Third International ICST Conference on Scalable Information Systems (Infoscale 2008), Vico Equense (Italy), June 2008

[8] IEEE 802.15.4 model for OMNeT++homepage, URL: http://www7.informatik.uni-erlangen.de/~fengchen/omnet/802154/

[9] F. Chen and Falko Dressler, "A Simulation Model of IEEE 802.15.4 in OMNeT++", in Proceedings of 6. GI/ITG KuVS Fachgespräch Drahtlose Sensornetze, Poster Session, Aachen, Germany, July 2007, pp. 35-38.

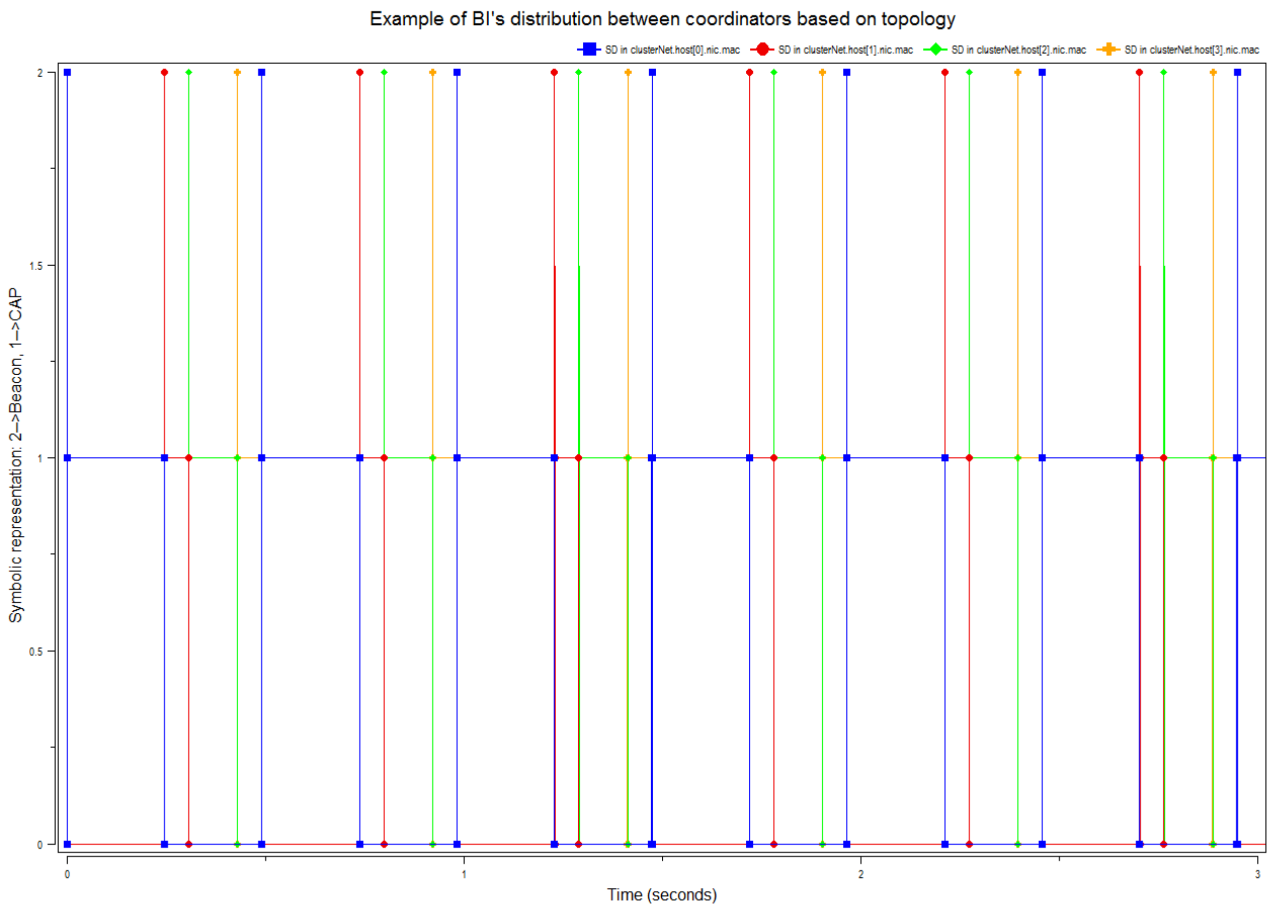

Figure 5. Visualization with OMNeT++ Plove tool of the distribution of the *BI* between coordinators