HAL open science

CoSenS reference guide

Bilel Nefzi, Ye-Qiong Song

To cite this version:

Bilel Nefzi, Ye-Qiong Song. CoSenS reference guide. [Technical Report] 2009. inria-00443594v1

HAL Id: inria-00443594

https://inria.hal.science/inria-00443594v1

Submitted on 31 Dec 2009 (v1), last revised 16 Apr 2010 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CoSenS reference guide

BILEL NEFZI AND YE-QIONG SONG

NANCY UNIVERSITY INPL, CAMPUS SCIENTIFIQUE - BP 239 – 54506

VANDOEUVRE-LES-NANCY CEDEX, FRANCE

{NEFZI, SONG}@LORIA.FR

INTRODUCTION

This report provides a reference guide to the CoSenS protocol stack. CoSenS provides a simple but efficient improvement of the MAC layer of IEEE 802.15.4 [1] in terms of reliability, delay and throughput. The network layer uses static routing or hierarchical tree routing of ZigBee standard [2].

The objective of CoSenS besides the improvement of IEEE 802.15.4 standard is to allow quality of service (QoS) support in wireless sensor networks. The MAC layer enables the implementation of scheduling policies and management mechanisms like congestion control and data aggregation.

PROTOCOL DESIGN

GENERAL PARAMETERS AND NETWORK ARCHITECTURE

We consider a wireless network composed of simple nodes and router nodes (named router in the rest of the document). A simple node is a device that generally has measurement sensors built in and has limited routing decisions capabilities. A router is a device that implements full routing and network management protocols. However, it can act like a simple device when its routing and management capabilities are disabled.

The network may be statically defined or dynamically organized into a tree. In the latter case, it uses the tree addressing scheme presented in ZigBee standard.

CHANNEL ACCESS PROTOCOL

GENERAL DESCRIPTION

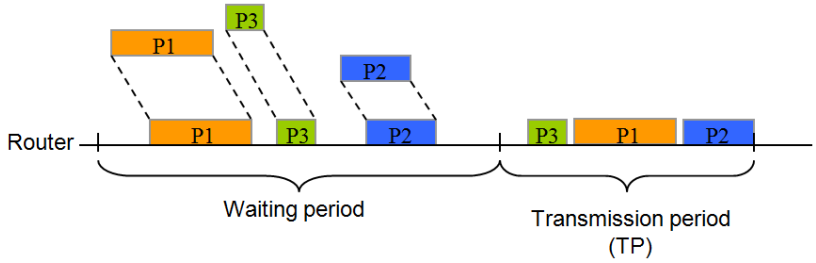

Fig. 1. An example of how the collecting and sending burst scheme operates

All nodes (simple nodes and routers) use CSMA/CA protocol to access the medium but with different maximum backoff intervals in order to give routers higher chance to access the medium; we used the unslotted version of CSMA/CA proposed by IEEE 802.15.4 (used in the non beacon enabled mode). CoSenS operates on the top of the "2450 MHz DSSS IEEE 805.15.4" physical layer; the data rate is equal to 250 kbps.

Routers use on the top of CSMA/CA protocol a collecting and sending burst scheme. In fact, a router does not transmit packets one by one upon their arrival. Instead, it waits for a period of time that we call *waiting period* (WP) and collects data either from its children or from its neighbor routers. After the end of the WP, the router starts transmitting all packets queued in its buffer in a single burst. We call this period of time the *transmission period*. Finally, at the end of the TP, the router starts another cycle and goes again to the waiting for reception state. WP and TP are illustrated in the example given by Fig. 1. Full details of the WP and TP are given next.

WAITING PERIOD

We define

- $\mathrm{d_S}$ and $\mathrm{d_R}$ as the maximum amount of time needed by a simple node and a router, respectively, to perform collision avoidance procedure (d_{BP}), sense the channel (t_{CCA}) and detect that it is Idle (at the first attempt), change the antenna state from receiver to sender (t_{tat}), send the packet (d_{pkt}), wait for an acknowledgment (t_{ack}), and receive it. See Fig. 2.

- N_{max} as an integer which has a minimum value of 1. Its value depends on the amount of received traffic.

- N_c^r as the number of children associated to the router *r*.

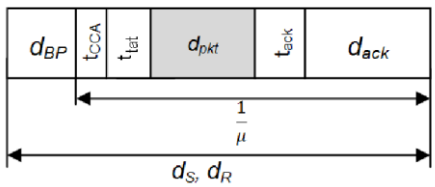

Fig. 2. $\mathrm{d_S}$, $\mathrm{d_R}$ and $\frac{1}{\mu}$

WP is given by the equation 2. When a router has no children, data packets come only from neighbor routers. Hence, we use $\mathrm{d_R}$ as a basis to calculate WP duration. In the other case, data packets can come either from children or neighbor routers. However, since $\mathrm{d_S}$ is greater than $\mathrm{d_R}$ (because d_{BP} is bigger for simple nodes), it is used as a basis to calculate WP duration.

$$WP = \begin{cases} N_{max} \cdot d_{\mathrm{S}} & \text{if } N_c^r \neq 0 \\ N_{max} \cdot d_{\mathrm{R}} & otherwise \end{cases} \quad (2)$$

ESTIMATION ALGORITHM

Let's define first the set $\Omega \in \mathrm{IN}$ as the set of WPs during which the router receives data and $WP_k,\ k \in \Omega$ these WPs. We define also, $U_k,\ k \in \Omega$ as the ratio giving the sum of the service times of received packets ($\sum_{i \in E_k} \frac{1}{\mu_i}$, E_k is the set of received packets during WP_k) divided by WP_k duration. $\frac{1}{\mu_i}$ is the service time of packet i (Fig. 2). U_k is shown in (3). We note that U_k can exceed 1 in the case where the router is still receiving a packet from a child or data from a neighbor router while the WP ends (recall that the router is not preemptive).

$$U_k = \frac{\sum_{i \in E_k} \frac{1}{\mu_i}}{WP_k} \quad (3)$$

Let us denote by S_k the exponential moving average of $U_k,\ k \in \Omega$

$$S_k = (1-\alpha)S_{k-1} + \alpha U_{k-1} \quad (4)$$

α is the smoothing factor. We used a non linear filter where α is bigger when $U_{k-1} \geq S_{k-1}$ allowing S_k to adapt more swiftly to traffic increase. After a number of trials, we found that values of α, $\alpha_1 = 0.008$ and $\alpha_2 = 0.01$, gave the best of all-around performance. The computation of S_k resembles that of the average queue length in RED [3] and the round trip time in TCP [4].

S_k is evaluated before the beginning of the kth WP and used in the Algorithm 1 to determine the value of N_{max} for the kth WP (thus its duration). In the algorithm, Thr_{max}, Thr_{min} and $NMAX$ designate the maximum threshold of S_k, the minimum threshold of S_k and the maximum value of N_{max}, respectively. Th_{max} allows the increase of the WP's duration. When set too high, the WP will not respond rapidly to the network load growth. When set too low, bandwidth will be wasted since the WP will increase too fast. Th_{min} is the minimum utilization ratio under which the protocol performance will degrade significantly. Th_{min} is used to decrease the WP's duration : set too high, WP decreases fast which diminish the benefits of our algorithm. When set too low, WP's duration decreases too slowly compared to the network load growth. The values of Thr_{max} and Thr_{min} are set to 0.28 and 0.75, respectively (tuned using some preliminary simulations with fixed WPs). $NMAX$ bounds the value of N_{max} to prevent a high increase in the WP duration. It is set to 15 in all simulations.

```
/* Update S_k */
α = α_1
if ( U_{k-1} ≥ S_{k-1}) then
  α = α_2
end
S_k = (1 − α)S_{k-1} + αU_{k-1}

/* Update N_max */
if (S_k ≥ Thr_max ) then
  N_max = N_max + 1
else
   if (S_k ≤ Thr_min ) then
      N_max = N_max − 1
   end
end
N_max = max(1, min(N_max , NMAX))
```

Algorithm 1. Estimation algorithm

Intuitively, S_k calculates the average utilization ratio of the WP. If S_k exceeds Thr_{max}, the value of the N_{max} is increased which increases the duration of the WP. If S_k decreases below Thr_{min}, the value of N_{max} is decreased. We can see that changing the WP duration, affects the value of U_k (3) and so S_k which in its turn affects the value of N_{max}. Besides, S_k depends also on the traffic load. Hence, our algorithm appears like an iterative process where the objective is to find the good value of N_{max} that reflects the incoming traffic load while satisfying an utilization ratio between Thr_{min} and Thr_{max}.

TRANSMISSION PERIOD

The collected data during the waiting period are transmitted in a burst. CSMA CA protocol is used to transmit only the first packet. Then, the remaining packets are transmitted directly upon the reception of the acknowledgement (ack)[1]. If an ack is not received for a given packet, it is retransmitted using CSMA/CA protocol. Then, the remaining packets will be sent directly.

CoSenS needs a finite amount of time to process data received by the PHY. So, a separation period between the acknowledgment frame and the next transmission shall be used (however, they are fixed to zero in the simulations).

NETWORK LAYER

The network layer may use static routing or hierarchical tree routing (HTR) protocol defined in ZigBee standard. In this section we describe the tree formation procedure and the HTR protocol.

NETWORK FORMATION

ASSOCIATION

We denote by the root, the router that starts the network and specifies all its parameters (like the parameters needed for address attribution). The root is equivalent to the the ZigBee coordinator in the ZigBee standard.

The association procedure works as follows. A node that wants to join the network, checks immediately its neighbor table to find the best candidate for association (the one that has the minimum depth and number of children). If a candidate is found, an association request is sent to this router. The node goes then to the waiting for reply state. If the neighbor table is empty, the node sends an “advertise_yourself” message, waits a defined amount of time and then checks the neighbor table again. The procedure is repeated until a candidate is found.

A router that receives an “advertise_yourself” message sends a hello message containing its address, depth, the number of its associated routers and the number of its associated simple nodes.

A router that receives an “association_request” message and accepts the association, sends an “association_response” message to the originator of the request containing the assigned address and the network parameters.

The association messages are not acknowledged. A node that issued an “association_request” message waits a given amount of time for the “association_response” message. If it is not received, the whole procedure is repeated. During the association procedure, all application packets are destroyed.

[1]Acknowledgments are enabled in CoSenS

ADDRESS ATTRIBUTION

The network addresses are assigned using a distributed address allocation scheme that is designed to provide every potential parent with a finite sub-block of network addresses to distribute to its children. During network establishment, the root determines the maximum number of children per parent (Cm), the maximum number of routers (Rm) between these children as well as the maximum depth of the network (Lm) (the root has a depth of 0). A function called $Cskip(d)$ (equation 4) is used after that to calculate the size of the address sub-bloc being distributed by each parent located at depth d.

$$Cskip(d) = \begin{cases} 1 + Cm \cdot (Lm - d - 1), & if\ \ Rm = 1 \\ \frac{1+Cm-Rm-Cm \cdot Rm^{Lm-d-1}}{1-Rm}, & otherwise \end{cases} \tag{4}$$

The network address distribution is as follows. The root has always the address 0. For routers, the address assignment uses the value of $Cskip(d)$ as an offset: if the node is the first served, its address is one greater than its parent address. Otherwise, the addresses are separated from each other by $Cskip(d)$. For simple nodes, network addresses are assigned in a sequential manner using the following rule :

$$A_n = A_{parent} + Cskip(d) \cdot Rm + n \tag{5}$$

Here $1 \leq n \leq (Cm - Rm)$ and A_{parent} represents the address of the parent.

HTR PROTOCOL

End devices (simple nodes in our case) always send their data to their father. For routers, HTR protocol is based on some comparison rules: for a ZigBee Router with address A at depth d, if the following logical expression is true, then a destination device with address D is a descendant.

$$A < D < A + Cskip(d - 1) \tag{6}$$

If it is the case, the Next Hop is given by the following rule:

- $N = D \quad for\ ZigBee\ end\ devices\ where\ D > A + Rm \cdot Cskip(d)$
- $N = A + 1 + \left\lfloor \frac{D-(A+1)}{Cskip\ (d)} \right\rfloor \times Cskip(d) \quad , otherwise$

If the expression is not true, the destination is not a descendant and the message should be routed through A's parent.

SIMULATION MODEL

NODE MODEL

CoSenS model is implemented within OPNET simulator [5]. The structure of the CoSenS sensor nodes (Fig. 3) used in the simulation model is composed of four functional blocks:

- The **Physical Layer** consists of a wireless radio transmitter (wireless_tx) and receiver (wireless_rx), operating at the 2.4 GHz frequency band and a data rate equal to 250 kbps. The transmission power is set to 0.05 W and the modulation technique is QPSK.

- The **MAC Layer** (mac) consists of an active queue where the server implements the CSMA/CA mechanism and the collect then send burst.
- The **Network Layer** (Nwk) implements the routing protocols and the association procedure.
- The **Application Layer** implements the traffic generator (gen) and the traffic sink (sink).

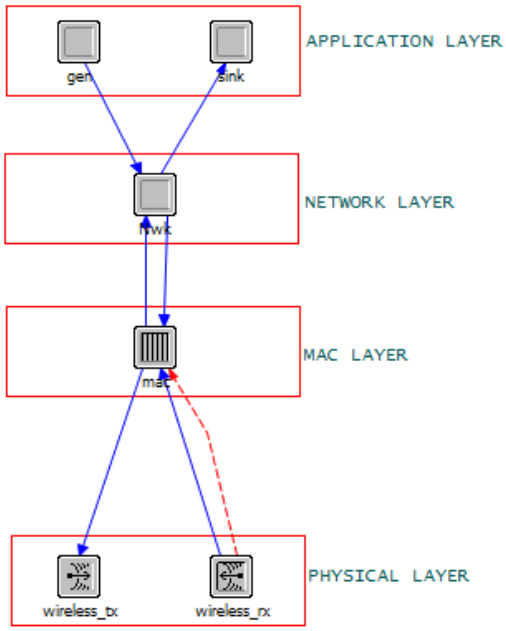

Fig. 3. Node model

Routers and simple nodes have the same node model but with different attributes. Table. 1 gives a summary of node attributes of routers and simple nodes.

Attribute		Router	Simple node
Acknowledgment		Enabled	Enabled
CSMA/CA	Minimum Backoff Exponent	2	3
	Maximum Number of Backoffs	4	5
Collect then Send burst Scheme		Enabled	Disabled
Data generation		Disabled	Enabled

Table. 1. Node attributes

MAC MODEL

The MAC process model is illustrated in Fig. 4. It consists of two states, **init** and **active**. In the init state, all the model parameters are initialized. The active state handles all the received events. Every transition handles an event. The pair *"(my_event)/my_function()"* means that the event *"my_event"* is handled by the function *"my_function"*. Here the list of the events.

- **PHY_RECEPTION**: Reception of a packet from the physical layer.
- **TX_DONE**: CSMA/CA process returns a transmission success.
- **TX_FAILED**: CSMA/CA process returns a transmission fail.
- **SEND_TIMER**: WP expired.
- **ACK_TIMER**: Acknowledgment timer expired
- **NWK_RECPTION**: Reception of a packet from the network layer

- **Default**: all other events

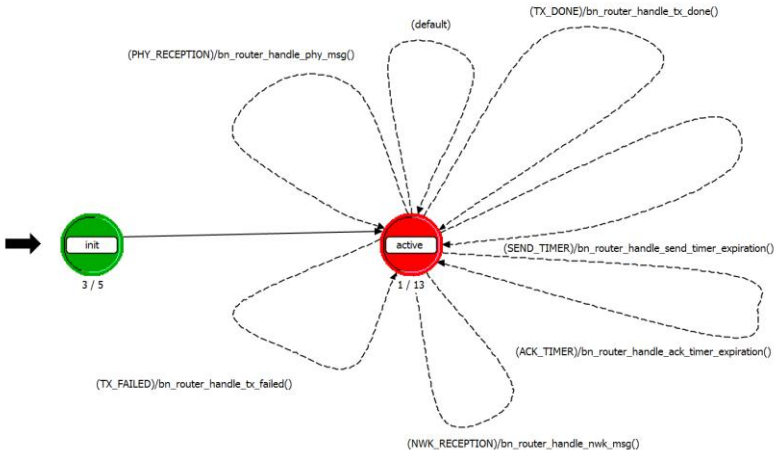

Fig. 4. MAC process model

CSMA/CA protocol is implemented as a process child of the MAC process. It is illustrated in Fig. 5. It is invoked either by *bn_router_handle_nwk_msg()* or *bn_router_handle_tx_done()* for simple nodes, *bn_router_handle_send_timer_expiration()* for routers and *bn_router_handle_tx_failed()* for both node types.

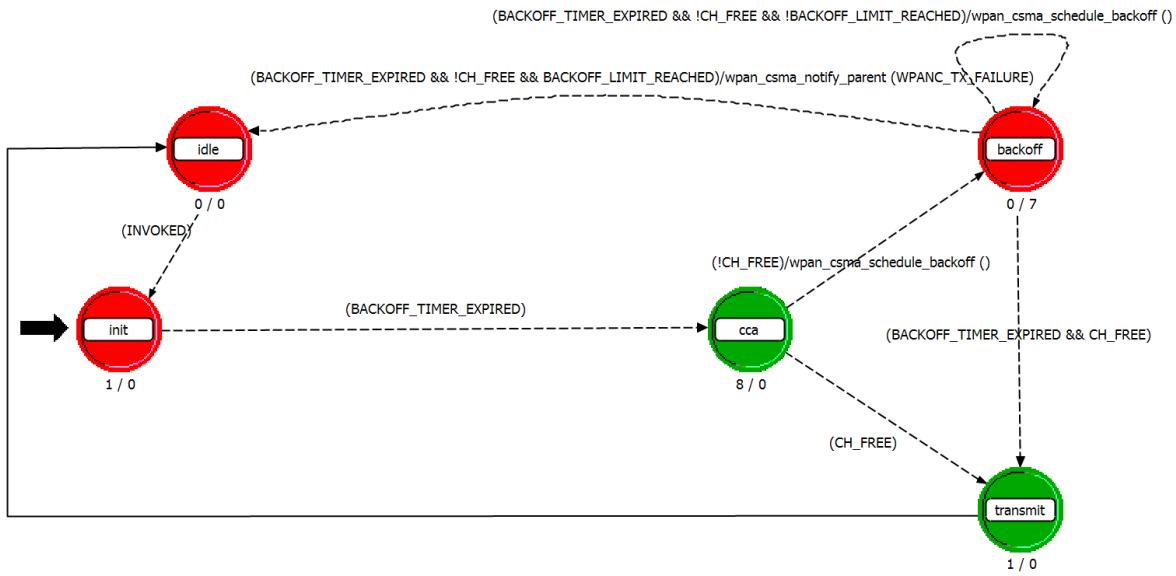

Fig. 5. CSMA/CA child process

NWK MODEL

The network process is illustrated in Fig. 6. It consists of three states; **init**, **association** and **active**. After the initialization phase (init) and depending on the routing protocol used, the model goes to the idle state or association state. The idle state handles all network events which are in our model, reception of an application layer packet (**APL_RECEPTION**) or an MAC packet (**MAC_RECEPTION**).

The association procedure is performed only when tree routing is used. The association model is implemented as a child process. It is illustrated in Fig. 7. It is invoked in the association state. The association protocol is described in the association subsection

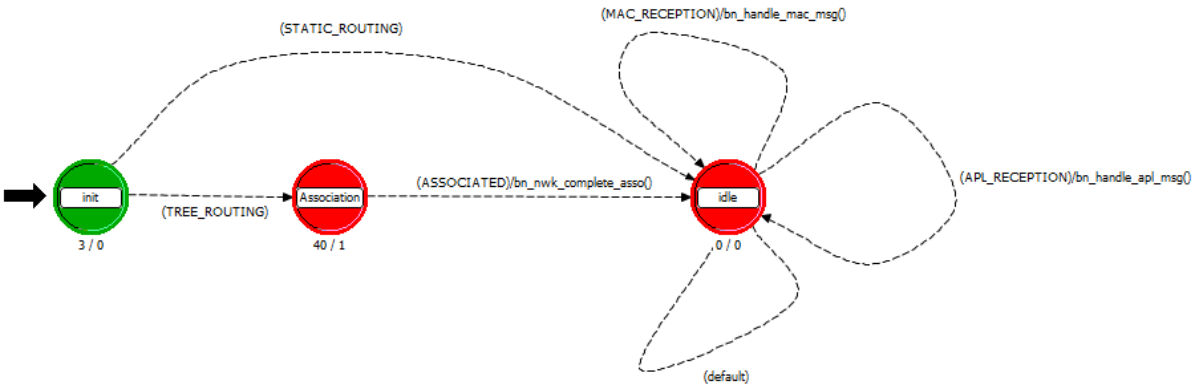

Fig. 6. Network process model

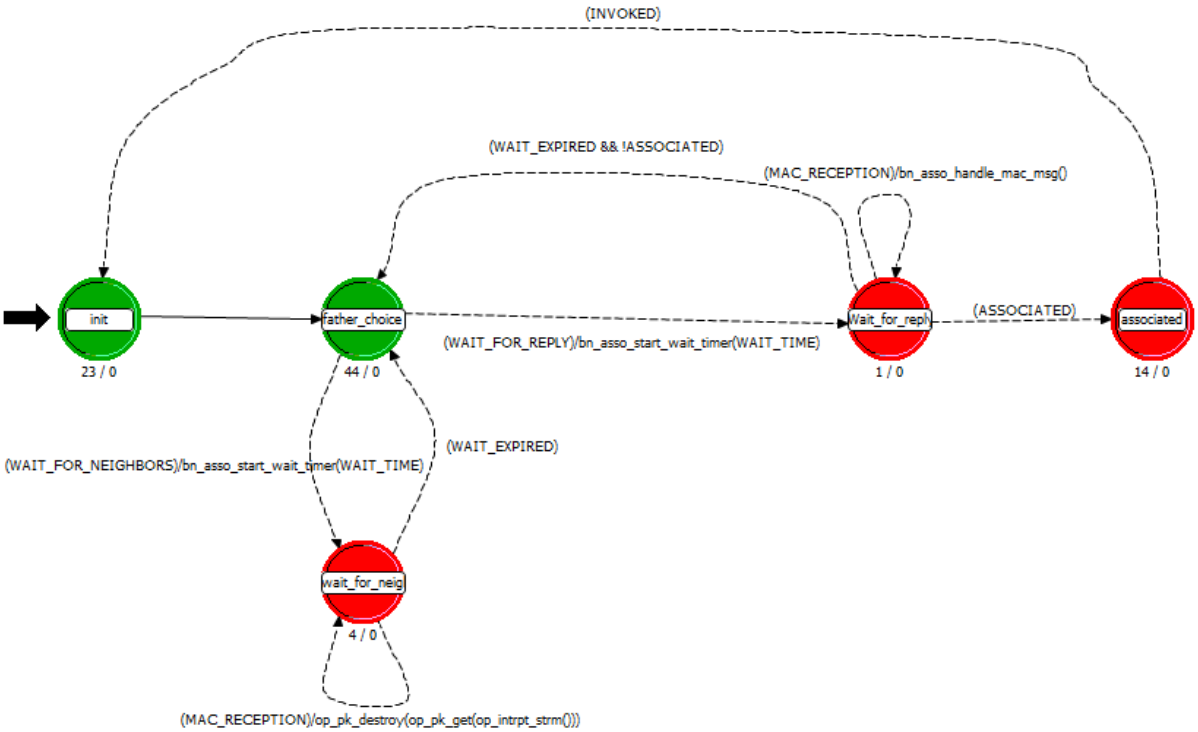

Fig. 7. Association child process model

PERFORMANCE ANALYSIS

SIMULATION DESCRIPTION

In this set of scenarios, we will analyze the CoSenS performance regarding the self-synchronization between routers, end to end delay and throughput. The self-synchronization between routers is defined as the percentage of the simulated time where their TPs do not overlap.

For our simulations, the packet length is constant and equal to 400 bits. The generation is Poisson distributed. Simulation duration is equal to 15min and all simple nodes start transmitting at t = 10 seconds.

In the first seven sections we used static routing and varied the topology and number of destinations. The objective is to study the influence of these parameters on protocol performance. The eighth section considers a random network with the use of HTR. The goal is to study the protocol performance under a more general topology. The last section analyzes the obtained results.

SCENARIO 1: STAR TOPOLOGY

DESCRIPTION

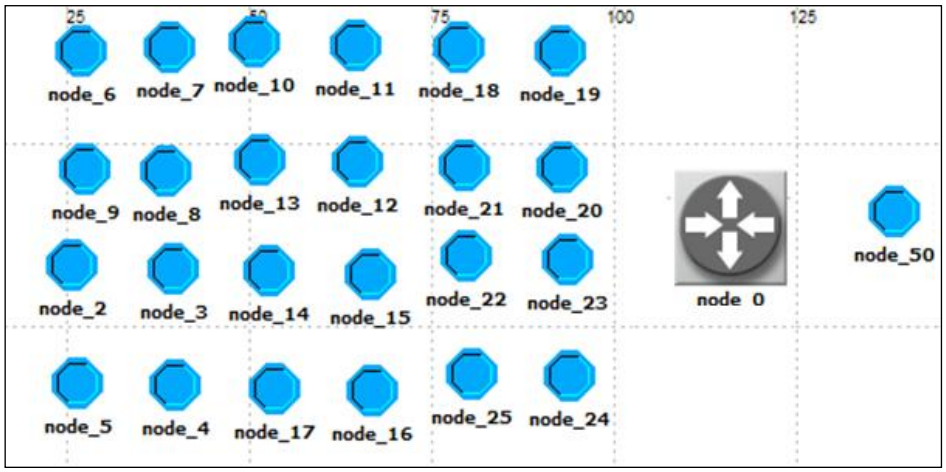

Fig. 8. Star topology

Nodes 6 to 24 send data to node 50 through the router 0. In this scenario we varied the inter arrival time and measured the transmission success rate and end to end delay.

RESULTS

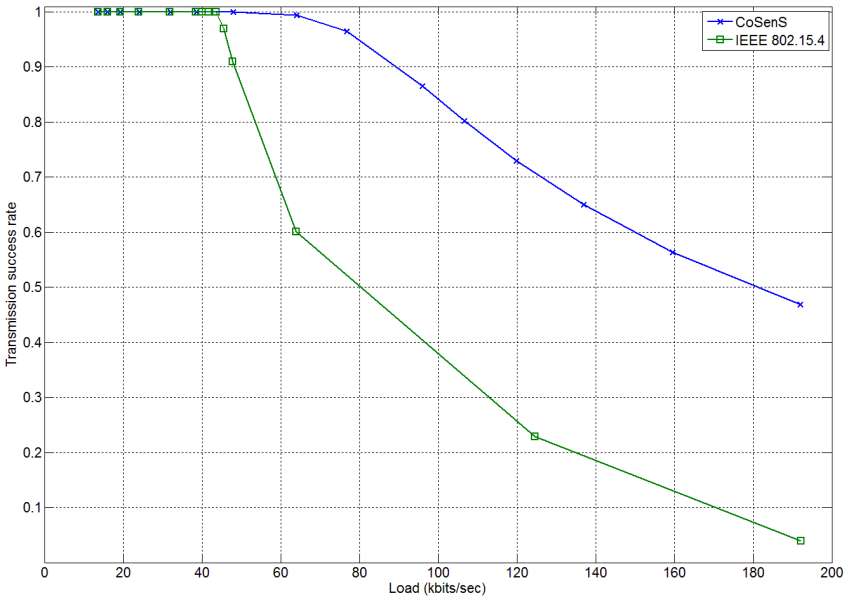

Fig. 9. Successful transmission rate comparison between CoSenS and IEEE 802.15.4 for star topology.

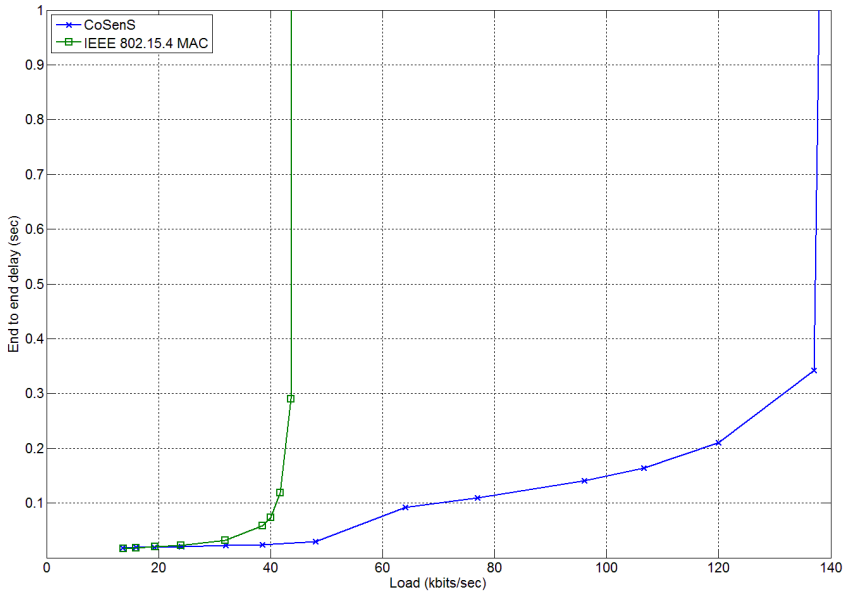

Fig. 10. End-to-end delay comparison between CoSenS and IEEE 802.15.4 for star topology.

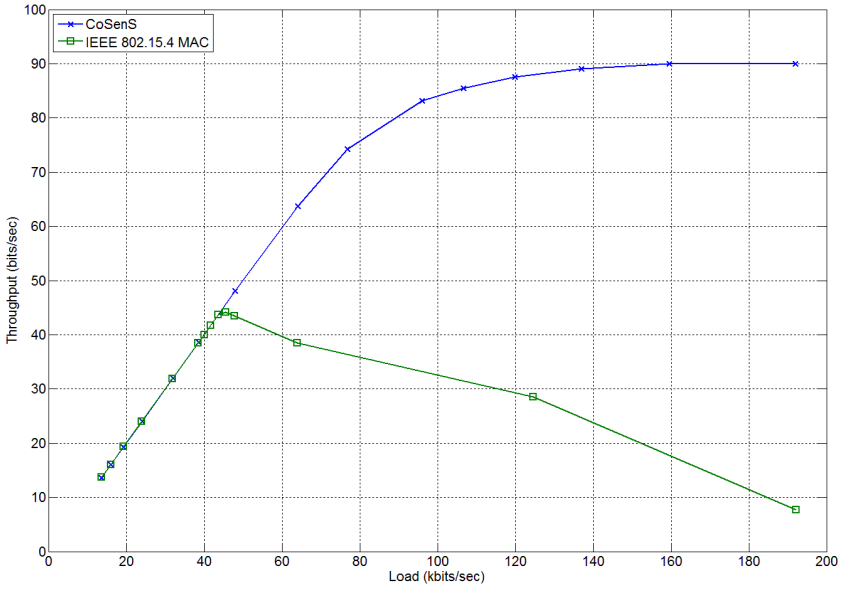

Fig. 11. Throughput comparison between CoSenS and IEEE 802.15.4 for star topology.

SCENARIO 2: TWO ROUTERS – SINGLE BROADCAST ZONE – TWO DESTINATIONS

DESCRIPTION

Nodes 6 to 24 are attached to router 0 and send data to node 50. Nodes 26 to 48 are attached to router 1 and send data to node 51. The inter arrival time is fixed to 0.3 seconds (~1.3 kb/s per node).

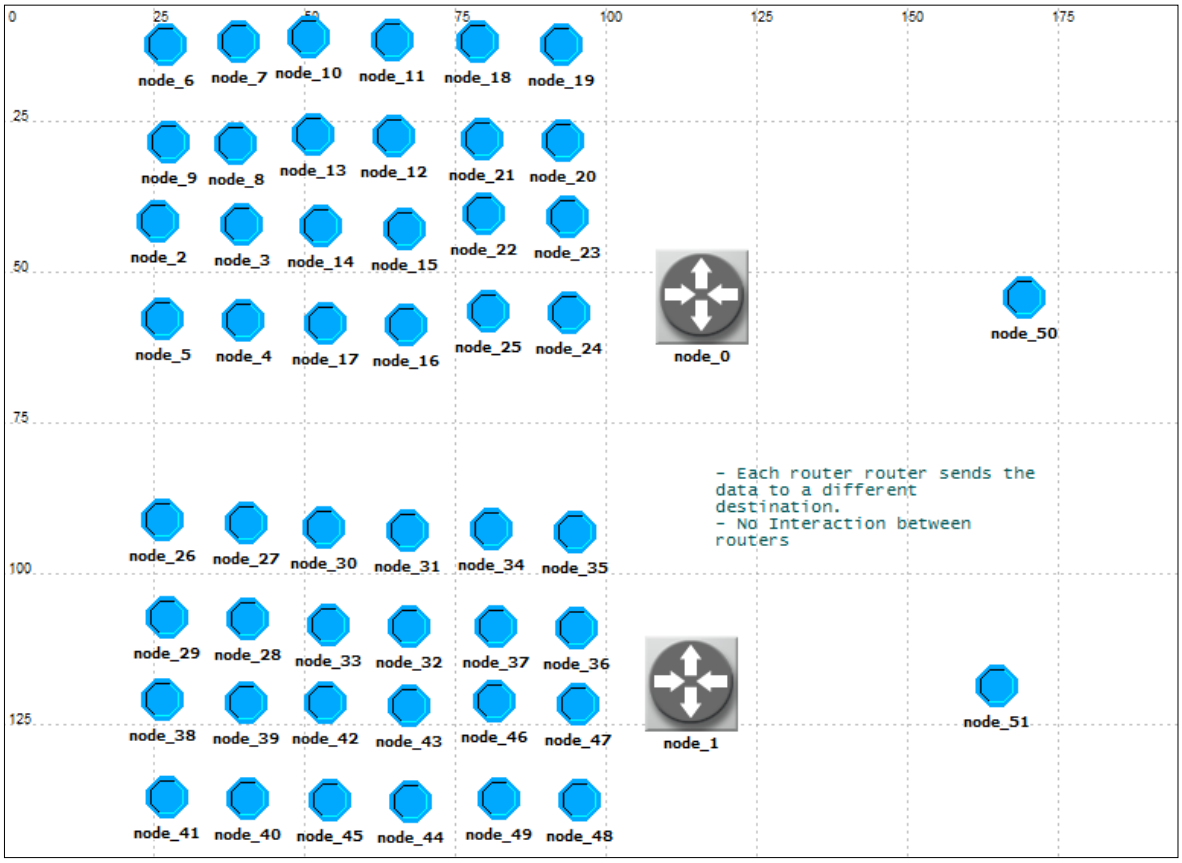

Fig. 12. Scenario 2.

RESULTS

SELF-SYNCHRONIZATION PERCENTAGE

Load (kb/s)	19.13	38.23	95.7
Self-synchronization	99.99	99.99	99.83

END TO END DELAY

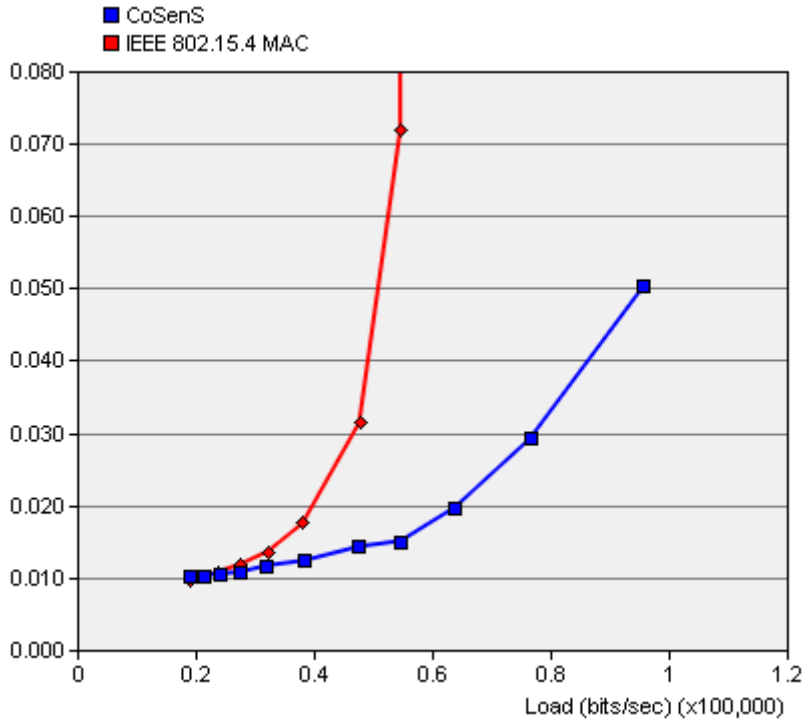

Fig. 13. End to end delay (s) for scenario 2.

WP DURATION (IN SECONDS)

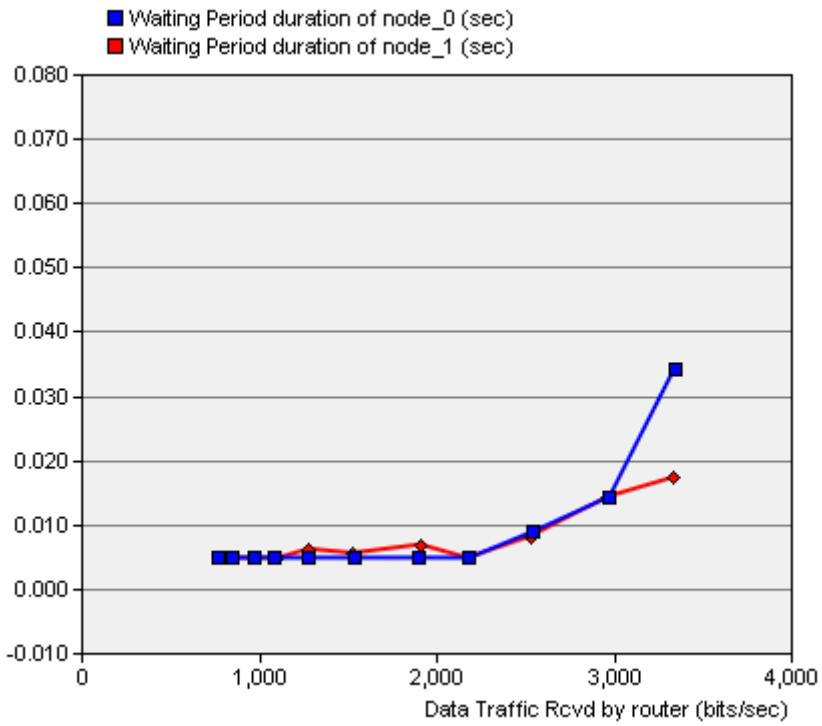

Fig. 14. Scenario 2 - WP duration.

THROUGHPUT

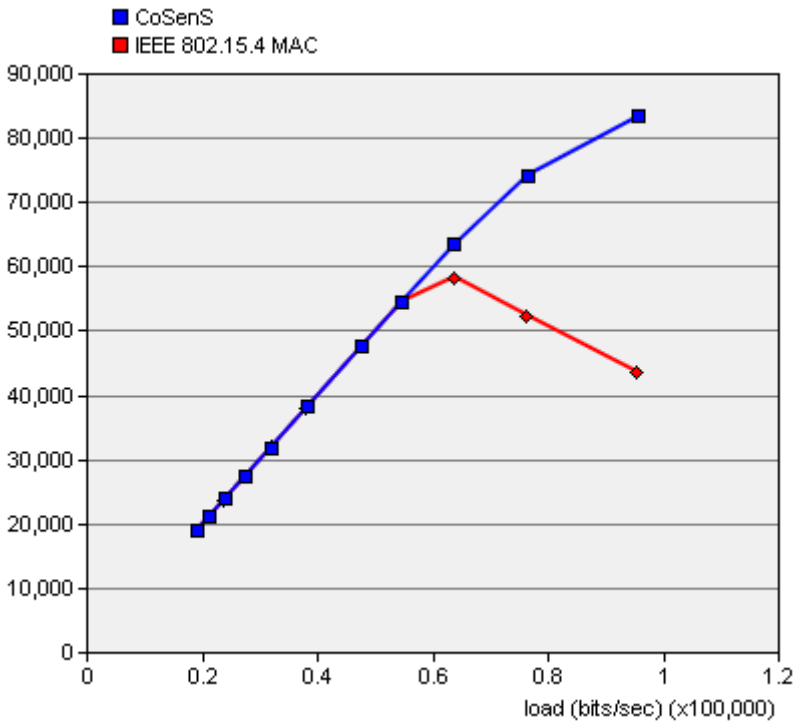

Fig. 15. Throughput (bits/s) results for scenario 2.

SCENARIO 3: TWO ROUTERS – SINGLE BROADCAST ZONE – ONE DESTINATION

DESCRIPTION

Two groups of nodes, each one attached to a router, send data to node_50 which is associated to node_0. So, node_1 forwards data to node_0. The inter arrival time is fixed to 0.3 seconds (~1.3 kb/s per node).

RESULTS

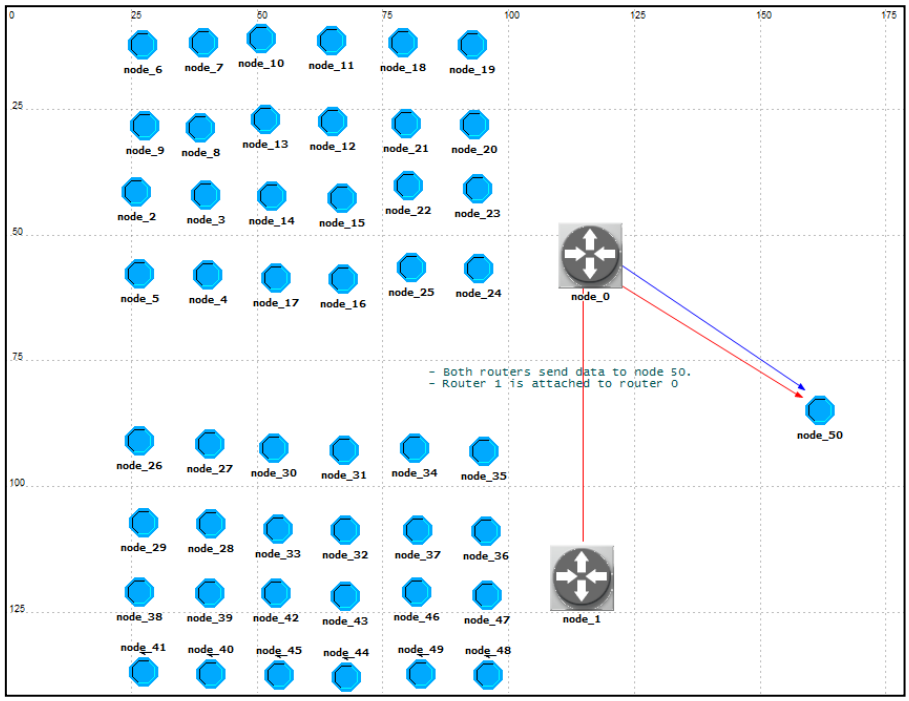

Fig. 16. Scenario 3.

SELF-SYNCHRONIZATION PERCENTAGE

Load (kb/s)	18.93	38.22	95.73
Sef-synchronization	99.99	99.97	99.59

END-TO-END DELAY

The end-to-end delay increases in comparison with the first scenario since data generated by nodes 26 to 48 go through both routers to reach the destination.

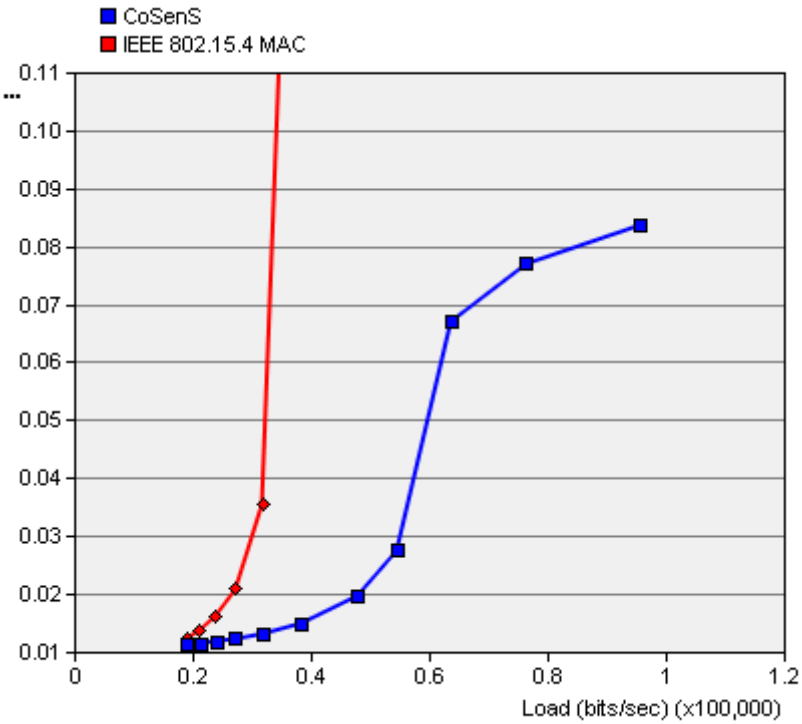

Fig. 17. End-to-end delay (s) for scenario 3.

WP DURATION (FIG. 18)

Observe the increase of the WP duration for node_0 in comparison with scenario 2 as the data traffic received increases.

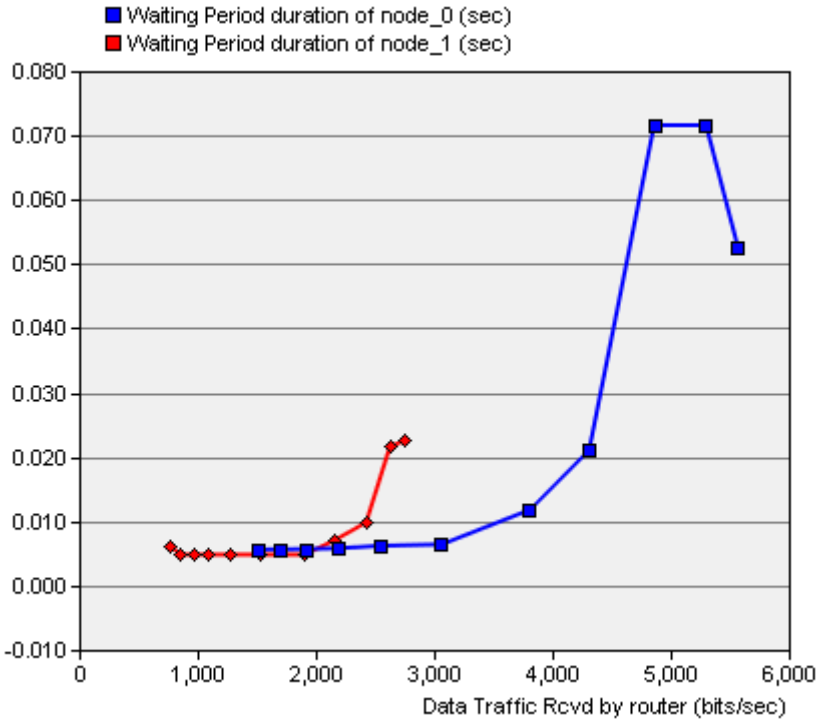

Fig. 18. WP duration for scenario 3.

THROUGHPUT (FIG. 19)

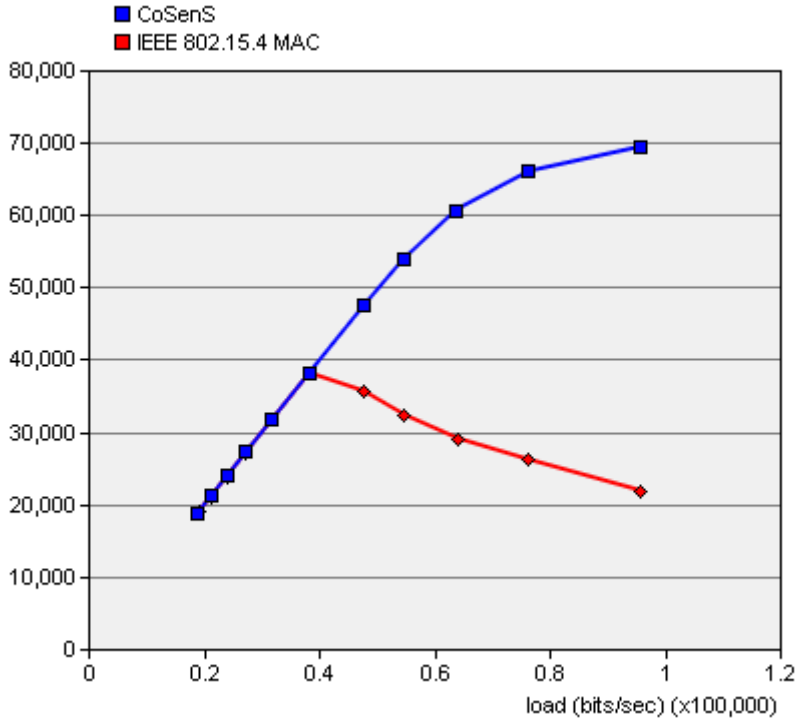

Fig. 19. Throughput (bits/sec) for scenario 3.

SCENARIO 4: FOUR ROUTERS – ONE BROADCAST ZONE – FOUR DESTINATIONS

DESCRIPTION

This scenario resemble scenario 2 but we added two node groups each one associated to one new router (Fig. 20). The inter arrival time is fixed to 0.5 seconds (~0.87 kb/s per node).

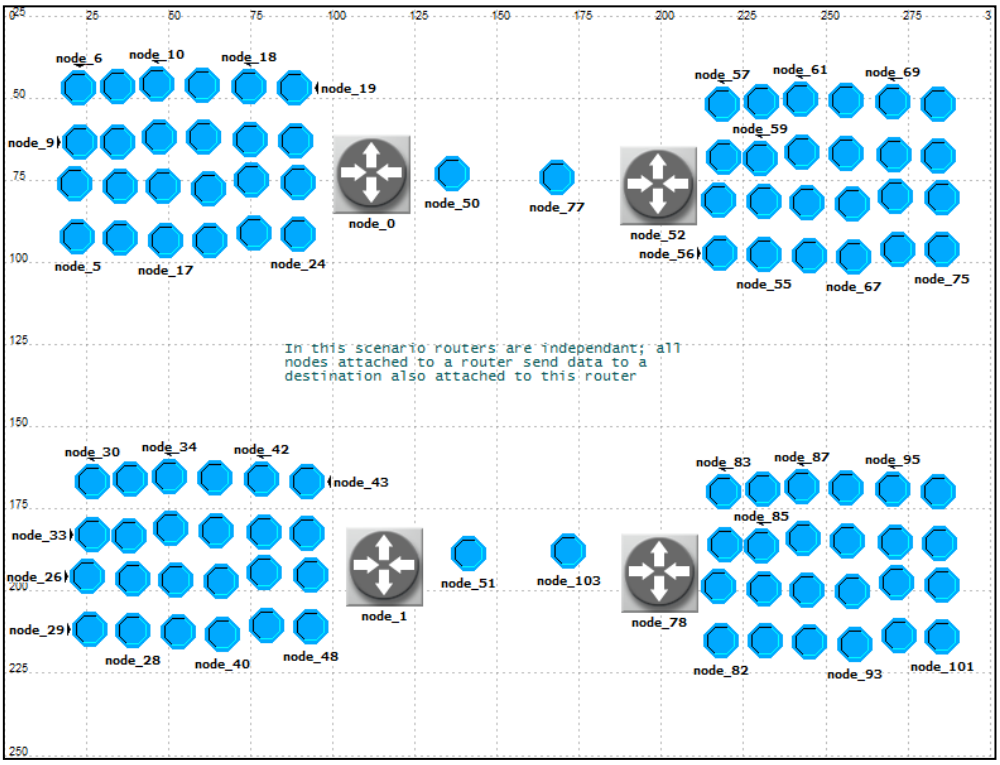

Fig. 20. Scenario 4.

RESULTS

SELF-SYNCHRONIZATION PERCENTAGE

Load (kb/s)	19.25	54.25	95.86
Sef-synchronization (%)	99.98	99.96	98.27

END-TO-END DELAY

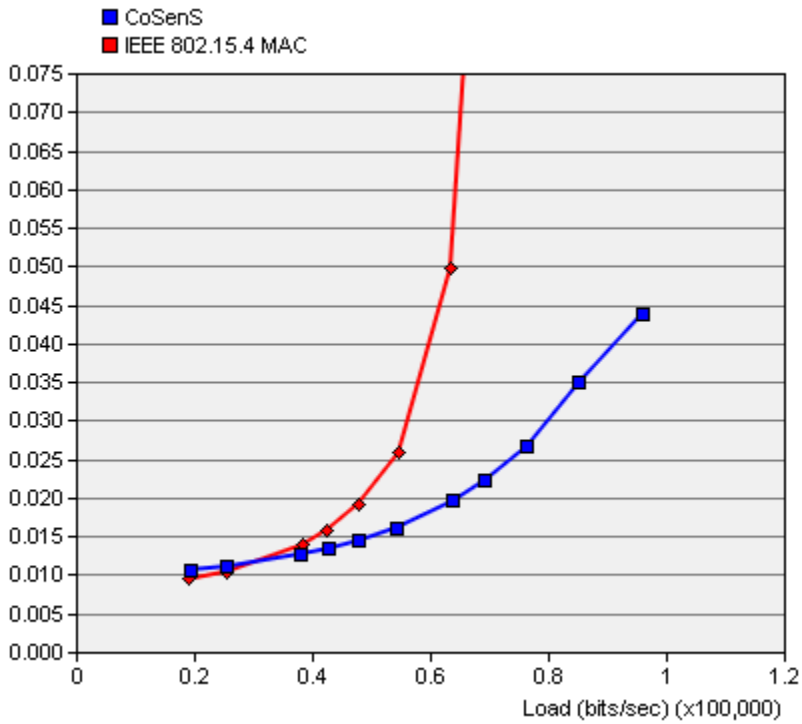

Fig. 21. End-to-end delay for scenario 4.

WP DURATION (FIG. 22)

It is the same for the four routers since they receive the same amount of data.

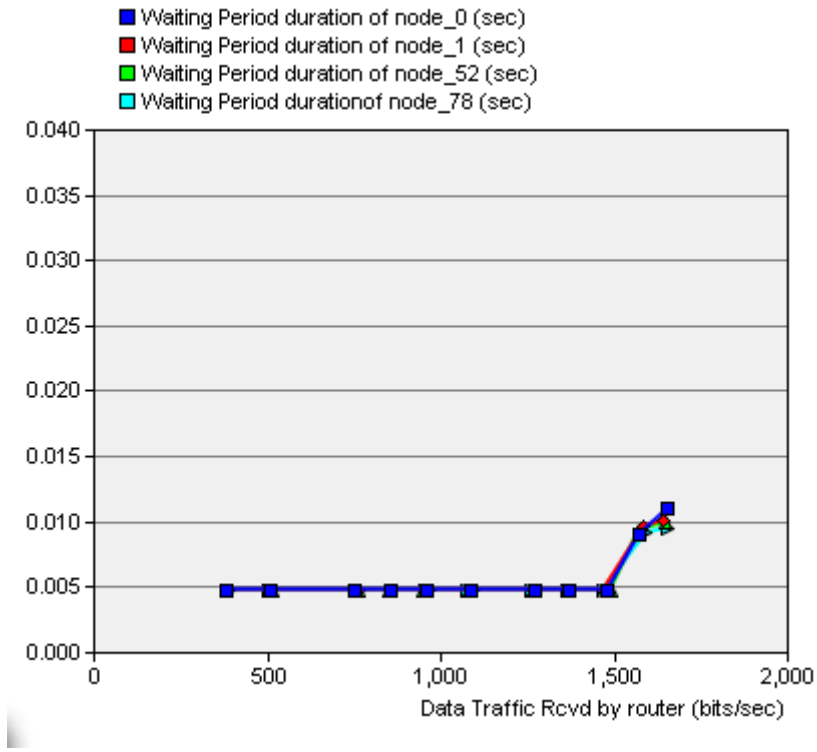

Fig. 22. WP duration for scenario 4.

THROUGHPUT (FIG. 23)

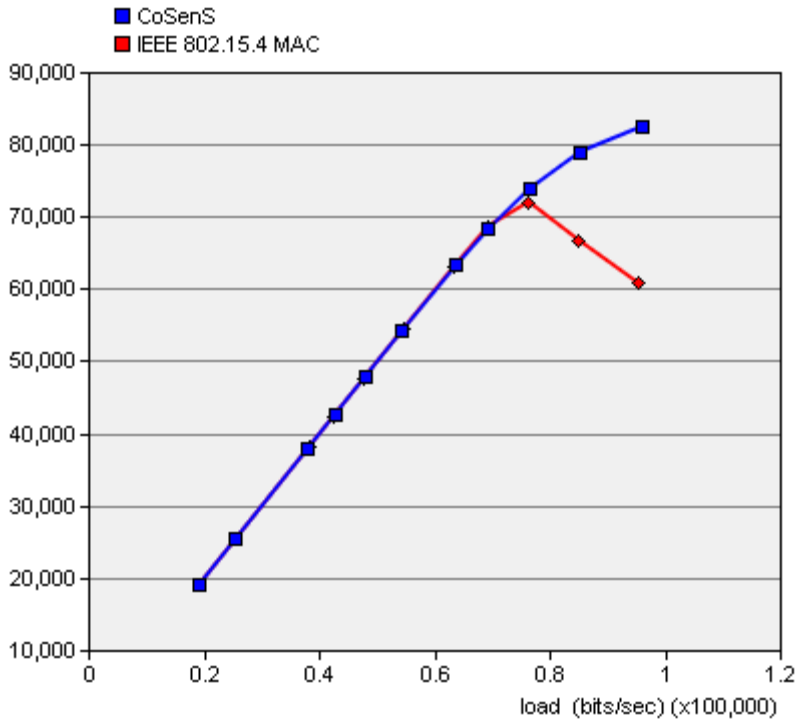

Fig. 23. Throughput for scenario 4.

SCENARIO 5: FOUR ROUTERS – ONE BROADCAST ZONE – TWO DESTINATIONS

DESCRIPTION

This scenario resemble scenario 3 but we added two node groups each one associated to one new router (Fig. 24).

RESULTS

SELF-SYNCHRONIZATION PERCENTAGE

Load (kb/s)	19.15	54.57	95.49
Sef-synchronization (%)	99.96	99.75	97.79

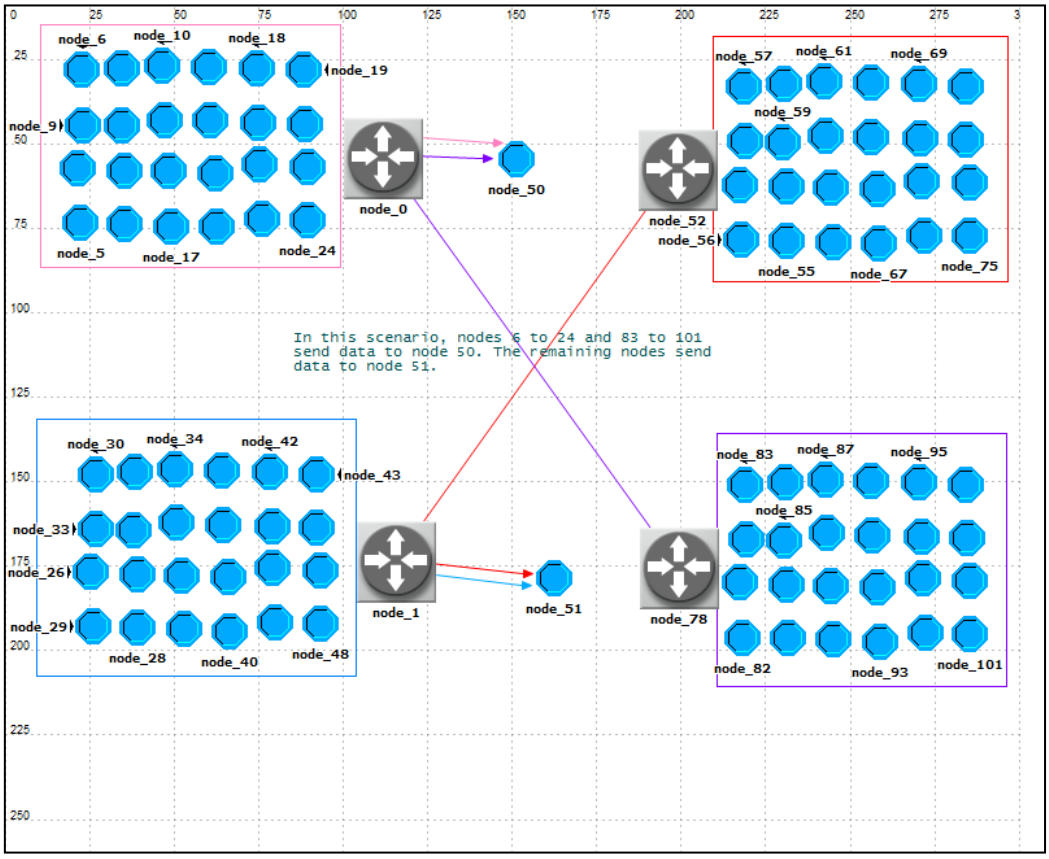

Fig. 24. scenario 5.

END-TO-END DELAY (FIG. 25)

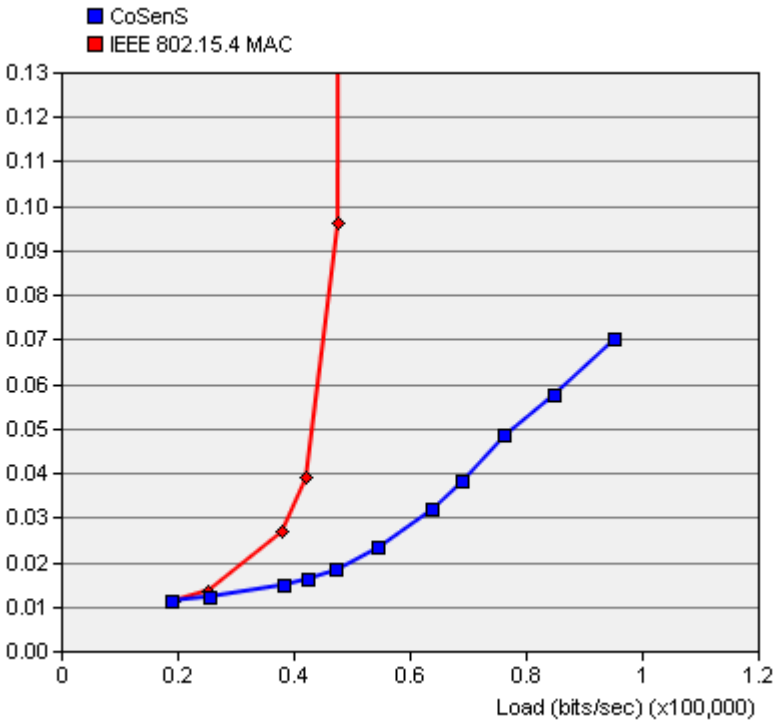

Fig. 25. End-to-end delay for scenario 5.

WP DURATION (FIG. 26)

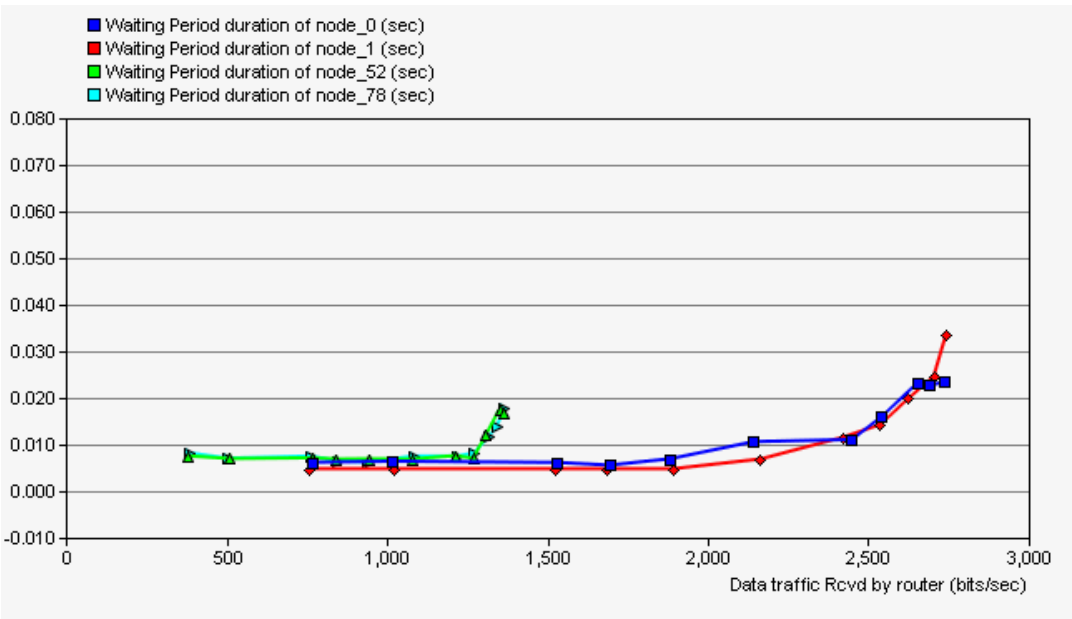

Fig. 26. WP duration for scenario 5.

THROUGHPUT (FIG. 27)

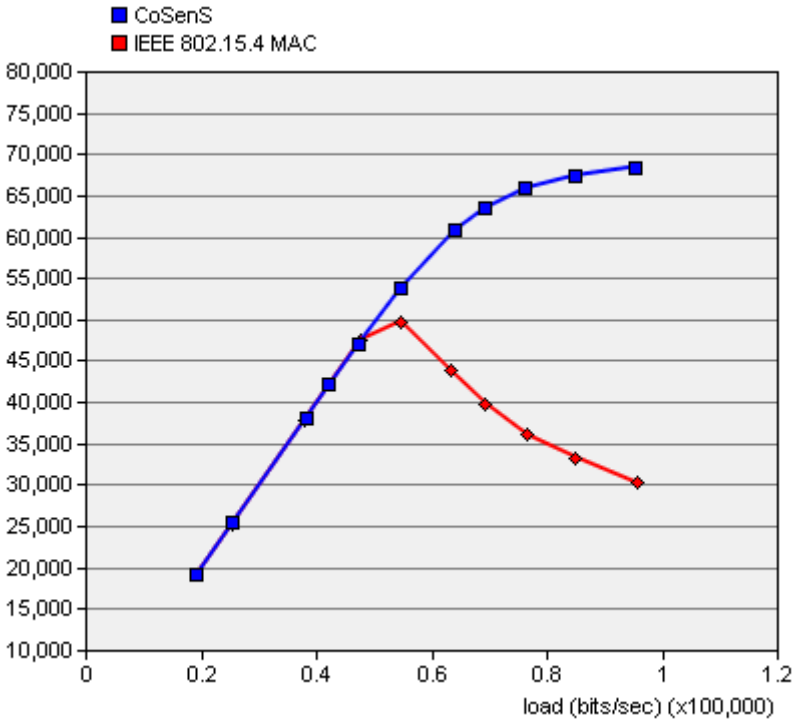

Fig. 27. Throughput for scenario 5.

SCENARIO 6: THREE ROUTERS – MULTIPLE BROADCAST ZONES – ONE DESTINATION

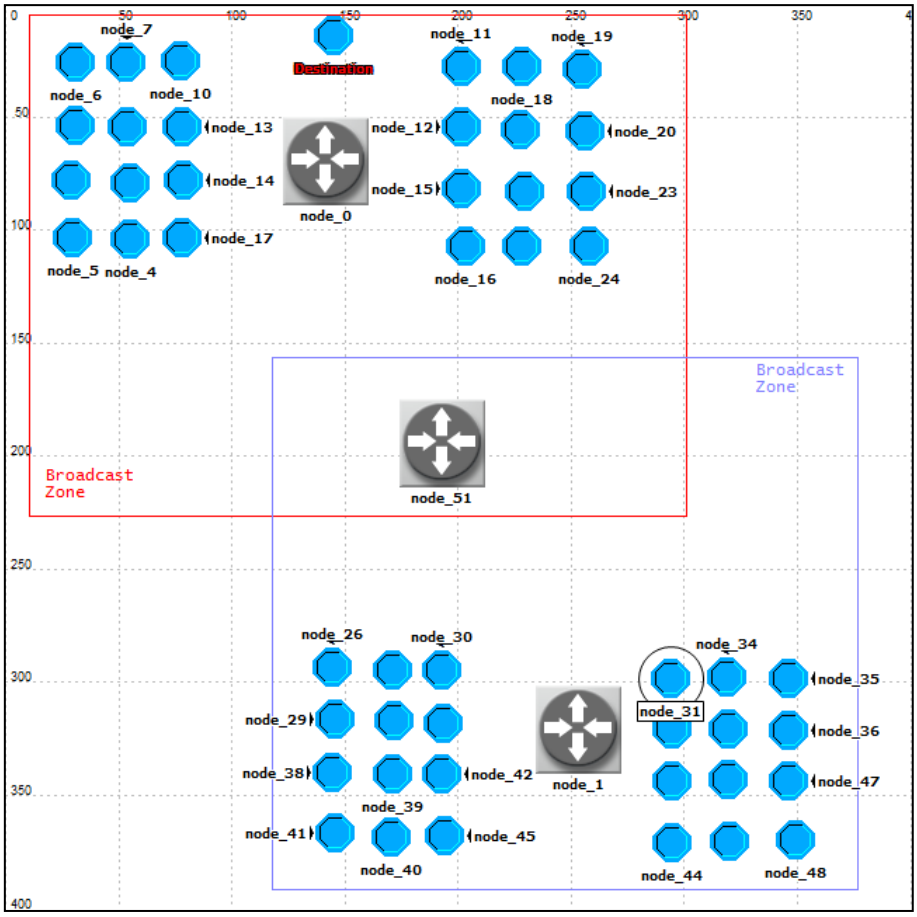

Fig. 28. Scenario 6.

DESCRIPTION

This scenario is composed of three routers and four groups of children. Each two groups are associated to a router. The router in the middle does not have children and is used as gateway between the other ones. All nodes send data to a single destination (Fig. 28).

RESULTS

SELF-SYNCHRONIZATION PERCENTAGE

Load (kb/s)	(a revoir)19.15	54.57	95.49
Self-synchronization of node_0 and node_51 (%)	99.99	99.97	97.89
Self-synchronization of node_1 and node_51 (%)	99.99	99.95	97.53

END-TO-END DELAY

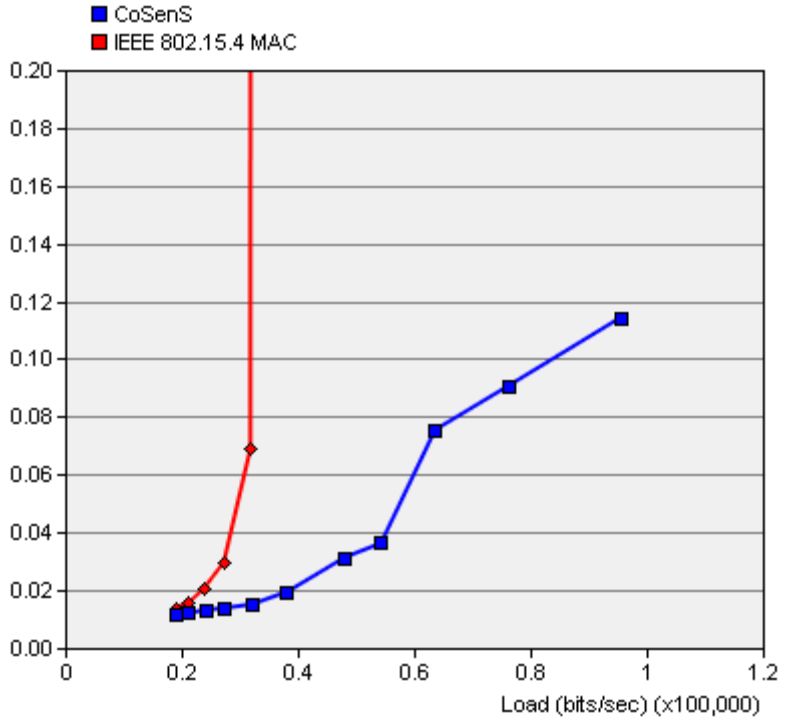

Fig. 29. End-to-end delay for scenario 6.

WP DURATION

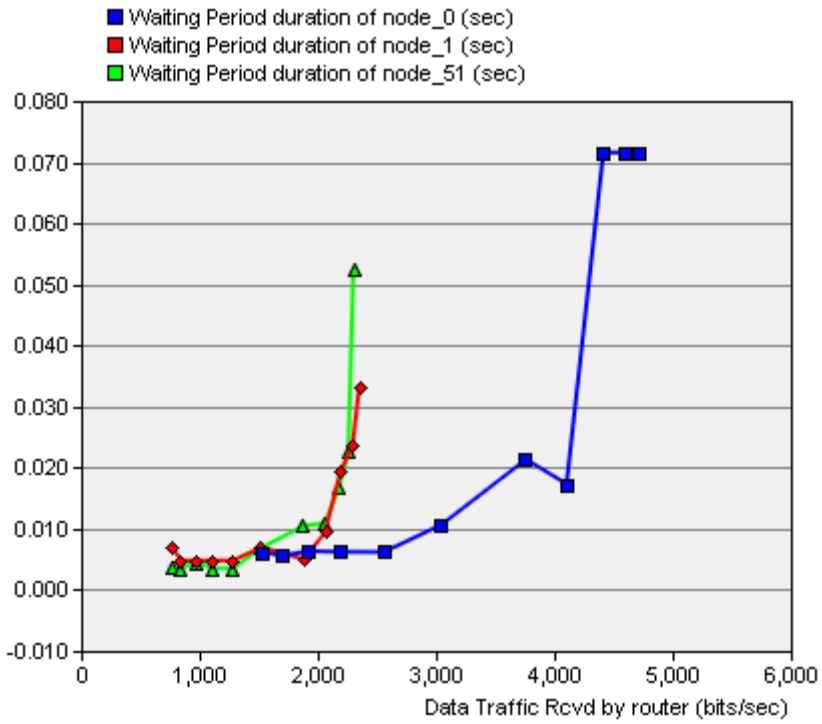

Fig. 30. WP duration for scenario 6.

THOUGHPUT (FIG. 31)

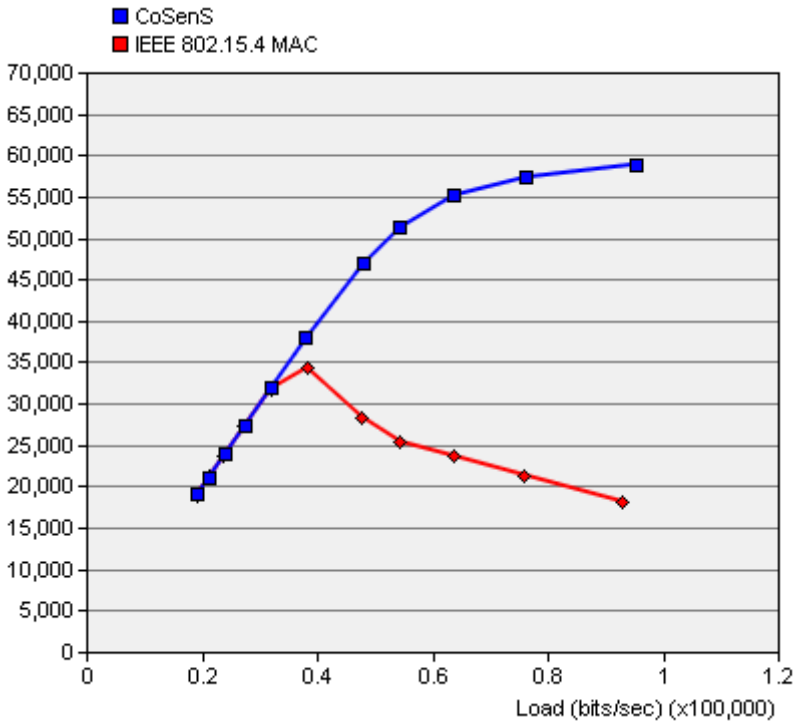

Fig. 31. Throughput (bits/sec) for scenario 6.

SCENARIO 7: MULTIPLE ROUTERS – MULTIPLE BROADCAST ZONES – MULTIPLE DESTINATIONS

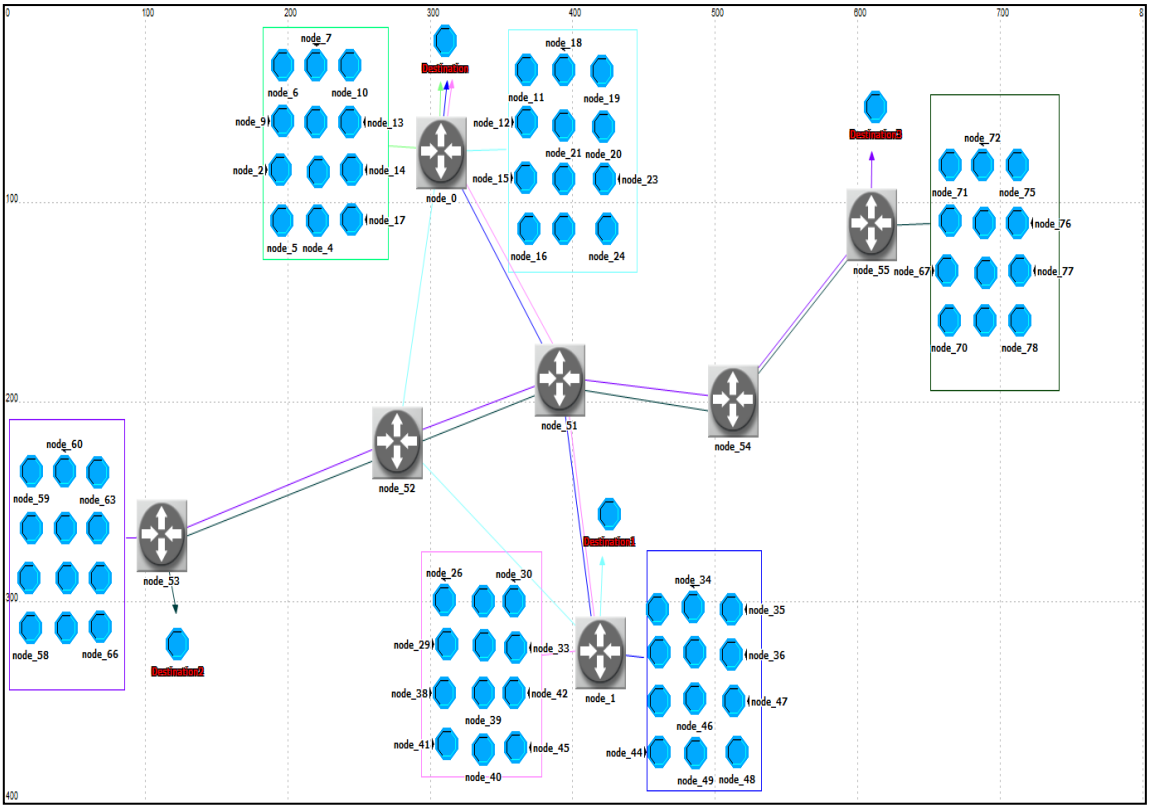

Fig. 32. Scenario 7.

DESCRIPTION

This scenario is composed of seven routers, six group of nodes (each one has 122 nodes) and four destinations organized as shown in (figure number).

RESULTS

END-TO-END DELAY

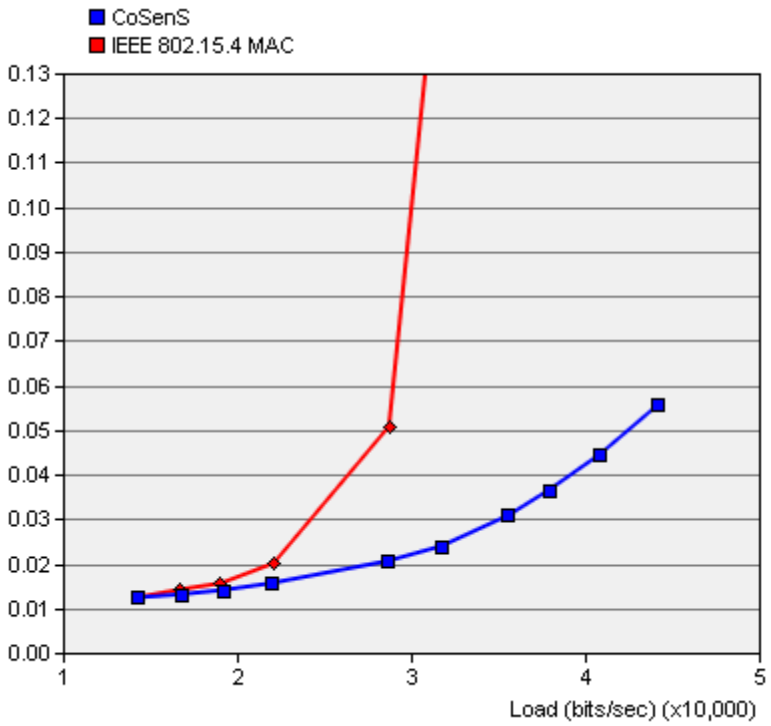

Fig. 33. End-to-end delay of scenario 7.

THROUGHPUT

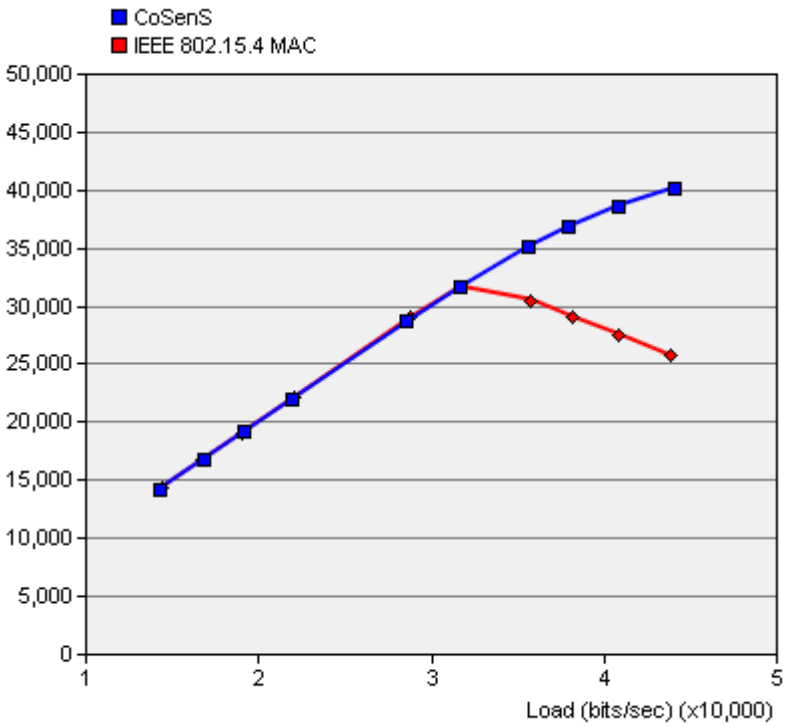

Fig. 34. Throughput of scenario 7.

SCENARIO 8: RANDOM TOPOLOGY – MULTIPLE BROADCAST ZONES – ONE DESTINATION

DESCRIPTION

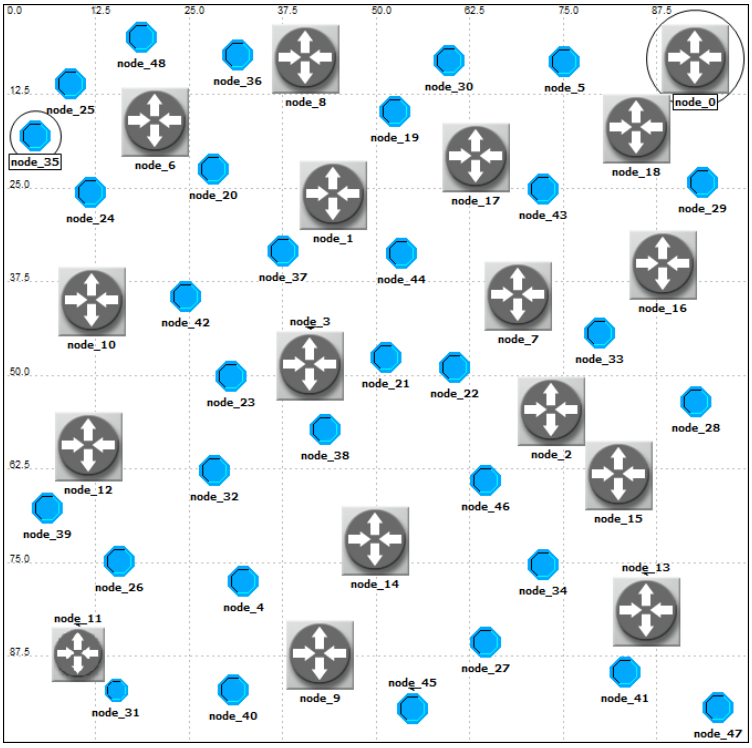

Fig. 35. Scenario 8.

RESULTS

TRANSMISSION SUCCESS PROBABILITY

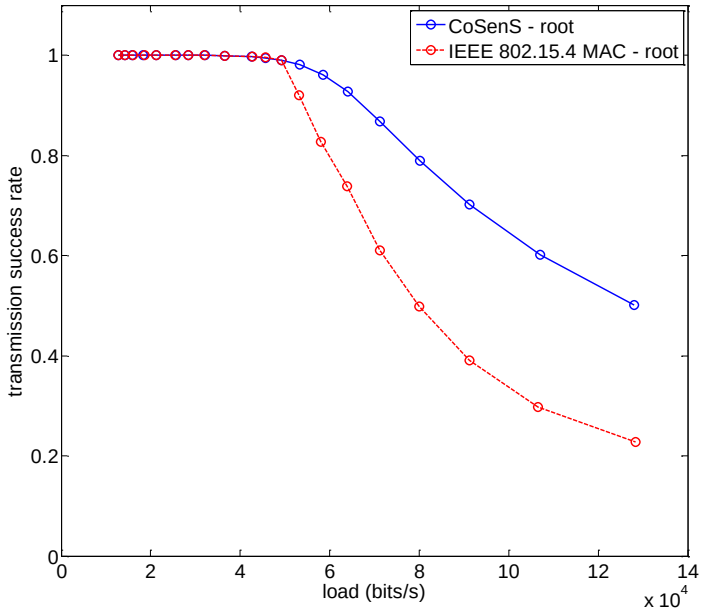

Fig. 36. Transmission success probability of scenario 8.

END-TO-END DELAY

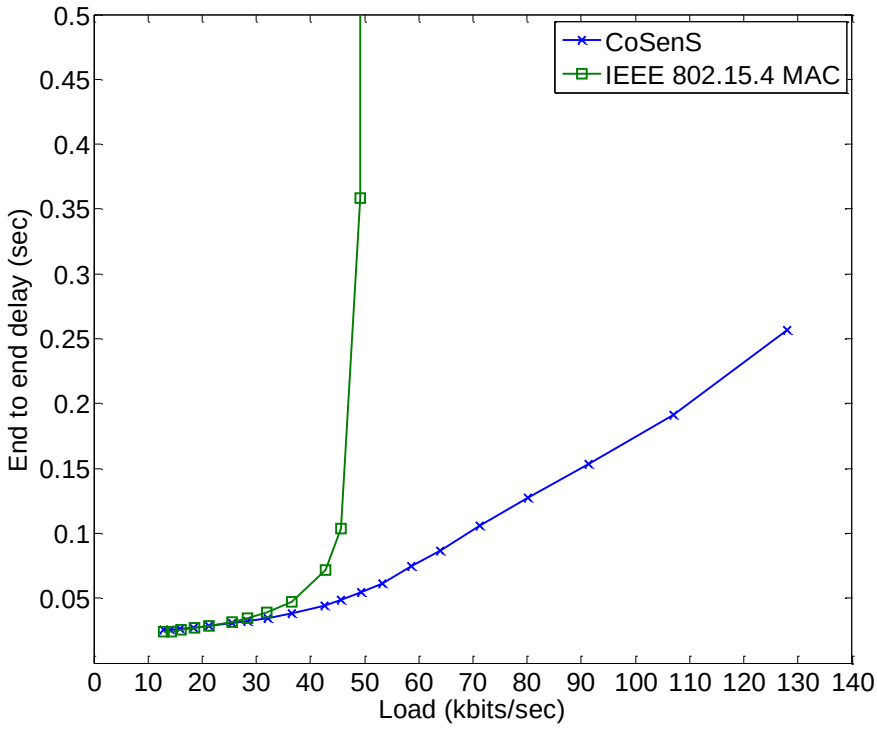

Fig. 37. End-to-end delay of scenario 8.

THROUGHPUT

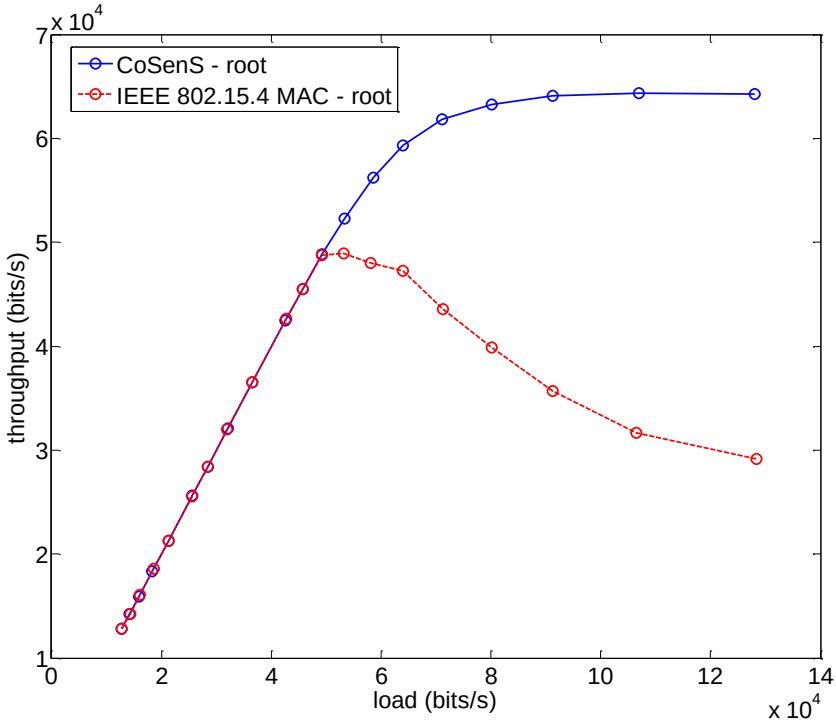

Fig. 38. Throughput of scenario 8.

RESULTS ANALYSIS

All simulations scenarios show that the CoSenS provides better throughput, end to end delay and reliability than IEEE 802.15.4 MAC for medium and high traffic load. The results are similar for light traffic load. CoSenS ameliorates the weakness of CSMA/CA protocol namely the throughput, end to end delay and transmission success probability while preserving its simplicity and good scaling property. CoSenS is a collecting and sending burst scheme, the fact to queue the packets at routers allows us to design efficient scheduling algorithms, a key feature for providing differentiated QoS.

CONCLUSION

This report describes CoSenS in details and gives the results of some simulation scenarios which have been carried out to analyze its performance in comparison to IEEE 802.15.4 MAC using the unslotted CSMA/CA protocol. The results show that CoSenS greatly enhance throughput, end-to-end delay and reliability.

BIBLIOGRAPHY [5]

[1] IEEE TG15.4 (2006), *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for low-rate Wireless Personal Area Networks (LR-WPANs). IEEE standard for information technology.*.

[2] ZigBee Alliance, *473-489 ZigBee Specification Document 053474r17.*.

[3] S. Floyd and V. Jacobson, "Random Early Detection (RED) gateways for Congestion Avoidance ," in *IEEE/ACM Transactions on Networking 1 (4)*, August 1993, p. 397–413.

[4] J. Postel, *Transport Control Protocol, RFC 793*. September 1981.

[5] OPNET Simulator, v14.5. http://www.opnet.com.