Amorim *et al. Journal of the Brazilian Computer Society* (2015) 21:6
DOI 10.1186/s13173-015-0027-y

Journal of
the Brazilian Computer Society
a SpringerOpen Journal

RESEARCH Open Access

Performance evaluation of single- and multi-hop wireless networks-on-chip with NAS Parallel Benchmarks

Amanda M.P. Amorim*, Poliana A.C. Oliveira and Henrique C. Freitas

Abstract

Background: Parallel processing in the era of many-core processors demands high-performance networks-on-chip and parallel communication based on intra-chip message passing. In this context, wireless networks-on-chip (WiNoCs) emerge to improve inter-core communication, bringing high bandwidth and low power consumption. In order to increase application performance, WiNoCs can support single-hop or multi-hop communication. The main issue is related to the performance of each communication architecture in the face of different parallel workloads. For this reason, the goal of this work is to evaluate and compare single- and multi-hop WiNoC architectures using parallel applications.

Methods: The methodology is based on simulations done using the well-known simulator Network Simulator 2 (NS-2) and applications from NAS Parallel Benchmarks (NPB). The WiNoC architectures are based on 2-D mesh topology and ultra wide band (UWB) radio technology. The inter-core transmission is evaluated concerning unicast communication (1:1 and N:1) and broadcast communication (1:N and N:N). This work has, as contribution to the state-of-the-art, the evaluation of both WiNoC designs (single- and multi-hop architectures) with parallel applications.

Results: Based on our results, the single-hop architecture has lower communication delay than the multi-hop version, and for some workloads, there were no packet losses. However, to achieve high-performance communication, the single-hop architecture consumed 63.12 J for the 256-node network, versus 0.22 J consumed by the multi-hop version.

Conclusions: Although single-hop WiNoCs reduce network bottlenecks and increase communication parallelism, they are recommended when energy consumption is not a critical factor.

Keywords: Performance evaluation; Wireless networks-on-chip; Single- and multi-hop architectures; Parallel workloads

Background

The search for better performance in computers and the limits of single-core architectures, like power consumption and restrictions in instruction-level parallelism, favored the emergence of multi- and many-core architectures [1–3]. In this type of architecture, a processor is composed by several cores and each core is able to process more than one instruction flow (thread) belonging to an application. Therefore, parallel applications make better use of the performance capabilities of multi- and many-core architectures, since concurrent parts of a program can be executed simultaneously, reducing the program's total execution time.

In order for results to be generated during parallel processing, information exchange between threads is often needed. Messages can be exchanged in the following communication patterns: 1 to 1, 1 to N, N to 1, and N to N [4, 5]. An interconnection between cores must exist for communication to occur.

Busses and crossbar switches are vastly used to connect the cores in a multi-core processor [6, 7]. These solutions are not applicable to many-core architectures (processors with a very high number of cores), because the increased number of cores demands an increase in

*Correspondence: amanda.amorim@sga.pucminas.br
Department of Computer Science, Pontifícia Universidade Católica de Minas Gerais (PUC Minas), R. Walter Ianini, 255, São Gabriel, 31980-110, Belo Horizonte, Brazil

wire length [8]. Lengthy wires generate more latency and electrical resistance and make routing packets between cores difficult, making the use of buffers and repeaters necessary.

As a means for providing a more efficient core interconnection, networks-on-chip (known as NoCs) were proposed [9, 10]. NoCs are responsible for establishing packet transmission-based communication between cores. In this approach, each core is associated to a router connected to other routers using wires through which packets travel to their destination. Networks-on-chip mitigate wire-related problems and provide communication-level parallelism.

The introduction of NoCs has reduced message delivery delay and network power consumption. However, since wires were not completely eliminated, they can still cause problems. Due to these wire-related problems, wireless networks-on-chip (WiNoCs) were proposed. In WiNoCs [11, 12], routers are associated to cores and radio signals are emitted by antennas (transceiver/receiver) to provide communication. Thus, high-bandwidth message exchange between cores is possible, with reductions in power and energy consumption [13, 14].

In parallel processing, performance is expected to increase in the same proportion as the increase in number of processor cores. However, this is not always possible in practice, due to scalability limitations in applications, communication limitations generated by programs during execution, and network restrictions.

It is believed that single-hop WiNoCs may favor some applications by reducing packet delivery latency, as a result of a reduction in router workloads for packet hops in communication between non-adjacent nodes. Additionally, network overloads caused by unnecessary retransmissions so broadcasts can reach every node are also reduced. On the other hand, by using multi-hop communication, transmitters, receivers, and signals can use less power, reducing total energy consumption.

Our previous work [15] described the performance evaluation of single-hop WiNoCs. It was extended in order to deepen the evaluation encompassing the comparison with multi-hop WiNoCs. In this regard, the goal of this work is to evaluate the performance of single-hop and multi-hop WiNoCs by simulating the execution of parallel workloads in these architectures. Another goal is to compare the results for both architectures, pointing out pros and cons for each one. The contribution of this article, therefore, is the proposition of single-hop and multi-hop WiNoCs as an intra-chip communication alternative in the context of parallel applications and the WiNoC design method and evaluation for the related state-of-the-art.

Networks-on-chip overview

NoCs are communication networks based on packet exchange, used to interconnect the cores of a many-core chip [9, 16]. The main components of a NoC are:

a) *routers* – responsible for packet forwarding, according to a chosen routing protocol, until packets reach their destination;
b) *network interfaces* – responsible for connecting each core to a router;
c) *links* – short wires between routers that consist of an interconnection medium.

Thread-level parallel applications explore the performance of a multi-core architecture the best. When executed, these applications are split into many instruction flows (threads) that will be processed by the architecture's cores. During execution, threads communicate using the network-on-chip according to collective communication patterns (from an application behavior point of view) defined by Duato [4]. Figure 1 presents the main communication patterns, which are:

a) *1:1 (one-to-one)* – a core in the network sends a message to a single other core (unicast);
b) *1:N (one-to-all)* – a core sends a message to every other core in the network (broadcast);
c) *N:1 (all-to-one)* – all of the cores send a message to a designated core (every core sends a unicast message);
d) *N:N (all-to-all)* – all of the cores send a message to all the other cores (every core sends a broadcast message).

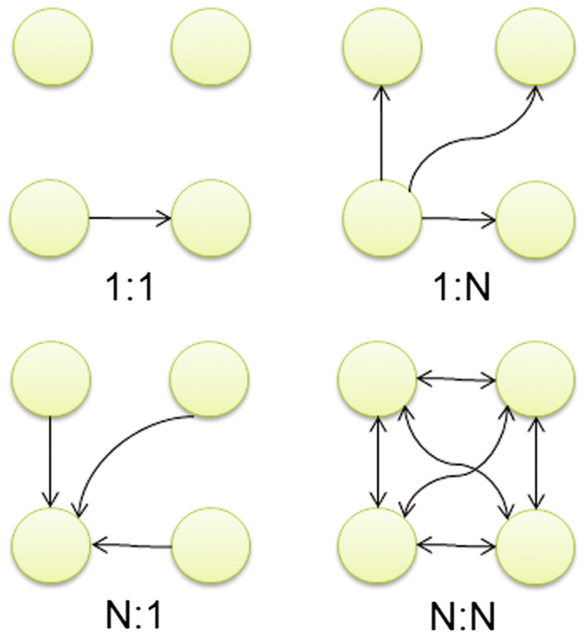


Fig. 1 Main collective communication patterns [11]

Freitas et al. [5] highlight the importance of knowing how parallel applications explore the collective communication patterns. This concept is regarded as important in the inception of general or specific-purpose NoCs, so that programs can obtain the best possible performance in a given architecture.

Ganguly et al. [17] show the increasing level of integration between cores in a NoC, which originates many-core architectures. This makes the limitations that arise from the use of wires become relevant again. These restrictions represent a hindering factor for NoCs, because communication between distant cores occurs in multiple hops, which naturally use more wires in message exchange, increasing latency and energy dissipation. WiNoCs were proposed in an attempt to solve the latency and energy problems, eliminating wire-related obstacles and increasing bandwidth at the same time.

Like NoCs, WiNoCs have routers physically connected to cores. As shown in Fig. 2, the difference is the network interface between routers, which in WiNoCs is an antenna (transceiver/receiver) that emits radio frequency signals to provide communication, reducing energy dissipation and packet transmission latency.

Related work

WiNoCs were created with the goal of solving the high-energy dissipation and high communication latency problems that emerged from the evolution of multi-core architectures into many-core architectures. Because the establishment of communication between cores in a chip is still a recent paradigm, many research works with distinct purposes are being conducted by the scientific community, targeting the development of more efficient and scalable WiNoC architectures.

Our previous works [15, 18, 19] conducted WiNoC studies using Network Simulator 2 (NS-2). Oliveira et al. [18] executed simulations based on synthetic workloads to generate communication between cores. In the other aforementioned works [15, 19], network traffic was generated by means of existent communication in parallel applications.

Li [12] reports some of the existing differences in, for example, WiNoC protocols, hardware, and topology. Concerning topologies, according to the author, both “pure” and hybrid (mixing wired and wireless connections) topologies can be found.

According to Ganguly et al. [17], if some or all wired connections in a NoC are replaced with wireless high-bandwidth single-hop connections, there will be a reduction in energy dissipation and latency, since these are caused by the multiple hops necessary to accomplish message exchange between distant cores. In order for the distance between cores not to increase a lot, it is suggested to split the network into smaller sub-networks as it grows. The cores in a sub-network can be interconnected via wires, since they will be physically close, and the communication between sub-networks is wireless. Also, Ganguly et al. [17] present a performance evaluation comparing NoC architectures with two different hybrid architectures (one was designed based on mesh topology and the other had a ring-star topology). For the comparisons, a 64-core architecture was simulated in a cycle-precise network simulator. The workload was generated by packet injection, with uniformly distributed spatial traffic. The authors concluded that the proposed hybrid networks can achieve higher performances than the compared NoC architectures.

Carloni et al. [20] describe the opportunities and challenges of three emerging core interconnection technologies. The discussed technologies are 3-D topology, nanophotonic (optical) communications, and wireless connections. Regarding wireless connections, the advantages of hybrid solutions were emphasized [17]. The authors finished by stating that the three studied technologies are promising solutions to the traditional NoC problems, but more research is necessary to overcome the multiple challenges like system architecture and device manufacturing.

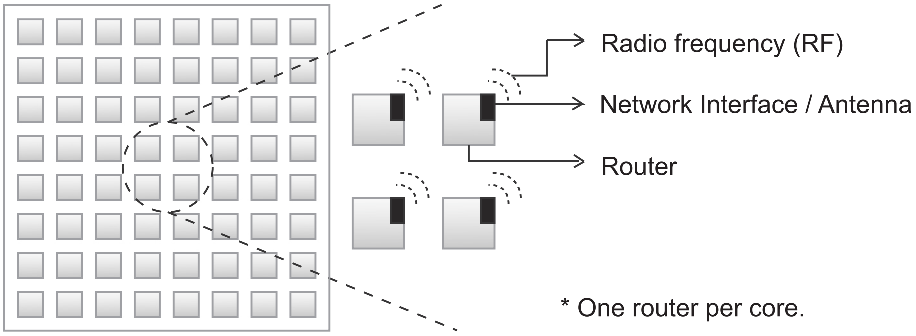


Fig. 2 Main WiNoC components, adapted from [18]

Pande et al. also analyzed hybrid NoC architectures [21]. The simulated networks had 128, 256, and 512 cores, and for all of them the number of cores in each sub-network was set to be 16, and only the number of sub-networks varied. Sub-network cores were connected in a mesh topology using wires. Wired and wireless connections were used to connect sub-networks. The number of wireless connections varied between 1, 6, and 24. Initially, the wireless links were randomly distributed and the distributions that obtained the best results were later determined. The conclusion was that inserting long-ranged wireless connections in a NoC significantly improves its performance. Additionally, the performance gains are more significant as the system grows in size.

A comparative study between wireless and optical NoCs is presented by Deb et al. [22]. The experiments were conducted in architectures of three sizes: 128, 256, and 512 cores; and with two workloads: uniform random traffic and specific traffic applications. The latter is composed by four distributions: HotSpot, Transpor, Fast Fourier Transform, and Matrix. The authors concluded that WiNoCs have higher bandwidth for all evaluated traffic patterns.

Another work proposing long-distance single-hop wireless communications in a NoC is presented by Wang et al. [23]. To analyze the proposed network, an architecture with 100 cores split into 4 sub-networks was simulated in a System-C-based simulator. Aiming to apply the temporal and spatial distribution behaviors of practical applications, the authors used three traffic-generating techniques: burstiness, injection with Gaussian distribution, and hop distance. Simulation analysis indicated that the addition of wireless routers between the sub-networks generated throughput improvements, in addition to reducing energy dissipation and latency.

Zhao et al. [24] propose an architecture they denominated McWiNoC (multi-channel wireless network-on-chip), consisting of a WiNoC with ultra-short connections using multi-channel UWB radio technology. In this approach, communication between adjacent cores is multi-hop. The authors also developed a routing algorithm based on core location and a method to prevent deadlocks. In order to evaluate the designed architecture, the authors developed a simulator. In the simulations, synthetic workloads were used. Results showed that McWiNoC had better performance in comparison with traditional NoCs.

A network split into sub-networks and interconnected using wired and wireless connections is also shown by Ganguly et al. [14]. The proposal consists of sub-networks using mesh topology in which the cores are connected to each other and to a central hub by wires. In the second level of the network, all of the sub-networks are also connected by wires in a ring topology. Initially, the available wireless connections are probabilistically distributed among pairs of hubs, based on the distance between them, measured in number of hops. After network start-up, the simulated annealing heuristic is used to reassign wireless connections for performance optimization. Tests were conducted with 128, 256, and 512 simulated cores, varying the number of sub-networks and the number of cores per sub-network. Results showed that the proposed network achieved better transfer rates and lower energy dissipation and delay.

No studies were found in the literature that evaluated multi-hop and exclusively single-hop WiNoC architectures using workloads generated by parallel applications. Only synthetic workload-based evaluations were found, and the single-hop WiNoC proposals are part of hybrid architectures. Thus, this work has, as state-of-the-art contributions, the two proposed WiNoC architectures (multi-hop and single-hop) with parallel workloads from NAS Parallel Benchmarks (NPB), along with their design method and evaluation. Besides, the results indicate which approach to use for prototyping WiNoCs.

Methods

The performance evaluation methodology for single- and multi-hop WiNoCs was based on the simulation model and divided into three stages. The first stage involved the selection and preparation of workloads, while the second stage consisted in configuring an environment capable of simulating WiNoC architectures. In the third and final stage, the performance evaluation of simulated WiNoCs with parallel workloads was made.

Selection and preparation of workloads

To evaluate the behavior of the proposed WiNoCs regarding communication in existing parallel applications, we decided to use some of the NPB kernels [25]. This choice was due to these applications being indicated to evaluate the performance of parallel supercomputers.

The chosen kernels were the following: Conjugate Gradient (CG), Embarrassingly Parallel (EP), Fast Fourier Transform (FT), Integer Sort (IS) and Multi-Grid (MG), configured with class A problem sizes [26], and Message Passing Interface (MPI) programming model.

These applications accurately represent the traffic pattern in a WiNoC, since they cover all collective communication patterns. As shown in Table 1, the 1:1 communication pattern is well represented by the CG and MG workloads. The N:N communication pattern is significantly present in the EP and IS workloads. N:1 communication makes up 42.38 % of the FT workload, which is the most balanced between unicast (N:1) and broadcast (N:N) communication. Although the 1:N communication pattern is not abundant in the workloads, it is part of N:N communication. In addition to the aforementioned kernels, NAS is also composed, for instance, by the Scalar

Table 1 Workload collective communication patterns

		Workloads				
Communication patterns		CG	EP	FT	IS	MG
Unicast	1:1	99.98 %	0 %	0 %	2.69 %	88.39 %
	N:1	0.02 %	0 %	42.38 %	5.56 %	0.13 %
Broadcast	1:N	0.0007 %	0.78 %	1.10 %	0.09 %	0.03 %
	N:N	0 %	99.22 %	56.51 %	91.68 %	11.46 %

Penta-diagonal (SP), Lower Upper Gauss-Seidel (LU) and Block Tridiagonal (BT) applications with similar characteristic and distribution of collective communication. They were not used due to their size, rendering them not viable to be run on the simulator for comparisons between single-hop and multi-hop WiNoCs.

Simulator choice

This stage had the goal of looking for alternatives to simulate WiNoC architectures. Due to the recent proposition of this interconnection model, there are no simulators available that focus exclusively on this type of architecture. Some authors [12] that conducted simulations using WiNoCs opted to develop their own simulation tools, which were designed to do very specific tasks and therefore are not applicable to this work.

For this reason, the alternatives were to either use NS-2 (widely used network simulation tool) and adapt it to the intra-chip context or to adapt an existing NoC simulator to work with wireless communication. Although NS-2 has been used in other works in the last few years to simulate NoCs [27, 28], we found no studies that used it in WiNoC simulations. In spite of that, NS-2 (version 2.29) was proven to be more favorable due to the associated difficulties to implement, in a short period of time, all the layers and protocols that constitute a wireless network in a NoC-specific simulator.

Network simulator

NS-2 is a network simulator based on discrete events, popular among academics for being free and open source. The project started at Berkeley University and has received the collaboration of many researchers. It is based on the OSI model network layers, and several technologies were implemented on it, such as queuing policies, routing protocols, transport agents (TCP - Transmission Control Protocol and UDP - User Datagram Protocol), wireless network protocols, and traffic-generating applications. Furthermore, it also has a graphical interface for network visualization, called Network Animator (NAM) [29].

NS-2 was implemented in two programming languages: C++ and Otcl. Its core was written in C++, because it is a more robust and dependable language, allowing for efficient, lower-level code to be developed. The simulation scenario configuration module was written in Otcl, which is an object-oriented, interactive, interpreted TCL script language. With it, configuration parameters can be altered for new simulations without recompiling the entire simulator code.

To configure an NS-2 simulation, a TCL script is used. In this script, users define the number of nodes in the network, network topology, link type, transport agent, routing protocols, type of traffic, among others.

NS-2 generates a network trace file as output, which is like a log. In this file, information about every packet that traveled through the network is stored. This information encompasses the type of event (packets being sent, received, or lost), timestamps, origin and destination addresses, energy consumption by the node involved in the event, identifier, packet size and type, an identifier for the message flow to which the packet belongs, among others.

Evaluation metrics

The WiNoC simulations were evaluated according to the well-known, established metrics in network research: packets sent (percentage of sent packets), packets lost (percentage of lost packets), injection rate (average number of bits injected in the network per second), throughput (average number of bits received by the network nodes per second), delay (packet delay from the origin to the destination in milliseconds), and energy consumption (per node and for the entire network, in joules).

Preparation of parallel workloads

The kernels selected from NPB were set up and executed with 4, 8, 16, 32, 64, and 256 processes in a multiprocessor cluster in order to record communication information during execution (sender node, destination node, timestamp, type of transmission, etc). These traces were manipulated to generate the input traffic files for NS-2 simulations.

Simulated architectures

The draft of the multi-hop WiNoC architecture was conceived by Oliveira et al. [11]. To design the single-hop WiNoC architecture, we opted to change only what was necessary so that communications stopped occurring in multiple hops and were carried out in a single hop. The performance of both architectures can then be compared based only on this difference. Table 2 presents the main features of both architectures and their differences.

Both the single- and multi-hop architectures use mesh network topologies. The radio technology for the WiNoCs is ultra wide band (UWB). Buffer size was fixed as 10 packets and each packet have a maximum size of 38 bytes according to the state-of-the-art [24]. What differentiates

Table 2 Characteristics of the simulated architectures

Parameter	Value - multi-hop WiNoC	Value - single-hop WiNoC
Topology	Mesh	Mesh
Distance between nodes	1 mm	1 mm
Node size (core/router)	0.5 mm^2	0.5 mm^2
Ad hoc approach	Multi-hop	Single-hop
Radio Technology	UWB	UWB
Central frequency	3.6 GHz	3.6 GHz
Bandwidth	1 GHz	1 GHz
Antenna	omnidirectional	omnidirectional
Antenna height	3 mm	3 mm
Transmitter power	0.9 mW	Varies with network size
Transmitter signal power	0.9 mW	Varies with network size
Receiver power	1.6 mW	Varies with network size
Data rate	1.16 Gbps	1.16 Gbps
Queue policy	FIFO	FIFO
Maximum queue size	10 packets	10 packets
Maximum packet size	38 bytes	38 bytes

the single-hop WiNoC from the multi-hop network is the fact that the single-hop architecture does not use routing protocols, while the multi-hop architecture uses the XY protocol. Power expenditure was also altered, since communication range in the single-hop network must be larger due to the fact that communication must occur in only one hop, with no intermediate nodes. Thus, the power used in transmitters, receivers, and signals is higher in the single-hop WiNoC.

Power usage in multi-hop and single-hop communication

After some test simulations with the multi-hop WiNoC, the values of 0.9 mW for transmitter and signal power and 1.6 mW for receivers were shown to be viable for simulations in NS-2. These values were enough for the emitted signal to reach neighboring nodes in a mesh topology network with a 1-mm distance between nodes.

For communication between nodes in a WiNoC to be single hop, the employed transmitting signal (Impulse Radio UWB) must be propagated to the entire network, reaching every node. This way, communication between all of the nodes can be achieved in a single hop. For such, we identified three WiNoC configuration parameters that needed to be altered: transmitter power, receiver power, and signal power. We decided to maintain the proportion between the previously established values (multi-hop version), so we multiplied them by the same constant when increasing them.

Aiming to find the lowest possible power values so that the signal could still reach every node in the network, no matter where the origin and destination nodes were located, tests were conducted using synthetic workloads to test packet transmission between nodes, especially unicast communication between the nodes that were the farthest apart. Broadcast communication was also tested. These tests were conducted for all of the different network sizes, and power values were altered until the desired values were determined according to the aforementioned criterion. Table 3 presents the final values obtained through the simulations for each network size.

Simulator adaptations

The following adaptations were made to NS-2 in order to better support WiNoC simulations:

a) *Integration of a UWB radio technology implementation* We integrated into NS-2 (version 2.29) the only available implementation of Impulse Radio UWB (IR-UWB) from the simulator's contributed code website [30, 31]. This implementation is an NS-2 plug-in. It adds the Dynamic Channel Coding MAC (DCC-MAC) layer and the IR-UWB physical layer (InterferencePhy - PHY) to the simulator. The physical layer uses time hopping to enable simultaneous transmissions in sub-channels, pulse-position modulation (PPM) for signal modulation, convolutional code for channel encoding, and a calculation based on bit error rate (BER) to obtain the error rate for each packet. A propagation model for UWB channels was also incorporated [32, 33].

b) *Addition of a Fast Broadcast module* For broadcast communication to occur in an ad hoc network with multihop routing, each node must, upon receiving a broadcast packet, forward it to its neighbors so every node in the network will receive the packet. This strategy causes network overloads due to the high number of packets generated in every broadcast forwarding. This is why the Fast Broadcast module [34] was added to the simulator. The module is an NS-2 extension and implements an algorithm designed to reduce the number of forwardings. To initiate communication, Fast Broadcast uses a module-specific application to generate network traffic. This was not viable in WiNoC simulations, due to the use of an

Table 3 Powers altered for single-hop context (in milliwatts)

Number of processes	Receiver power	Transmitter/signal power
4	1.6	0.9
8	4.16	2.34
16	7.36	4.14
32	23.52	13.23
64	39.69	22.32
128	110.88	62.37
256	181.92	102.33

application (TrafficTrace) that schedules packet transmissions by reading trace files. As a solution, a modification was made so that Fast Broadcast was triggered directly from TrafficTrace by an agent [11].

c) *Modification of the file-based traffic generation mechanism* The TrafficTrace application generates network traffic by reading binary input files containing two fields to represent transmissions. The first field specifies the time interval, in milliseconds, after which transmissions can occur; the second field defines the size of packets that will be sent. The application proved to be adequate to generate traffic compatible with NPB application workloads. To accomplish this, some modifications [19] needed to be made. The first one consisted in altering the first field of the file so that it would represent the exact moment in which the packet would be sent. The application was also altered so it would generate transmissions from the files only once instead of for the entire duration of the simulation, as was done previously. Further adaptations included the following: the field representing the time in which to send a packet has its unit changed to seconds; the input file type was altered to text from binary, so larger time moments could be represented; a third field was added to the file to determine what kind of agent must be used when sending a packet according to communication type (0 for unicast and 1 for broadcast).

d) *Adjustments in the physical layer* Some adaptations were made to the class Phy/WirelessPhy/InterferencePhy. In its original state, the class randomly picks a packet to be received from the synchronization list. This is not viable when there is intense traffic in the network, because rescheduling problems may happen if the predicted acquisition time of a packet expires, which generates a fatal error causing the simulation to crash. To avoid this problem, the class was altered to always choose the last packet to be inserted into the synchronization list, since it has a smaller probability of having an expired predicted reception time. Another necessary modification was the value of the sync_thresh variable. According to Merz et al. [33], this variable sets the sensitivity level for the network, relative to the energy needed to detect reception signals. Its value was changed to 10 dBm from the original −84 dBm, because it is the maximum value [35] recommended by the IEEE 802.15.4a standard document.

e) *Adaptations to the Fast Broadcast module to simulate broadcast transmissions in single-hop WiNoCs* The Fast Broadcast module that was integrated into the simulator is responsible for making broadcast transmissions in multi-hop networks. Using an optimization algorithm, some routers are selected to retransmit received packets that correspond to broadcast messages to their neighbors, making packets get to every node in the network. For the module to also be used in single-hop WiNoC simulations with broadcast communication with no packet retransmissions, we studied the module and found out that retransmissions were generated in the process_data_BroadcastMsg method, called after receiving a packet with the recv method of the BroadcastAgent class. The recv method was altered to no longer call the process_data_BroadcastMsg method, so retransmissions would stop. Synthetic workloads were used in simulations in order to test the modifications. Results showed that the modifications did not affect broadcast communication and that packets were no longer being retransmitted. The Fast Broadcast module could then remain being used to generate 1 to N communications without packet routing in single-hop WiNoC simulations. It is important to emphasize that the modifications were only made for single-hop WiNoC simulations. For multi-hop WiNoCs, the unaltered Fast Broadcast module was used.

Results and discussion

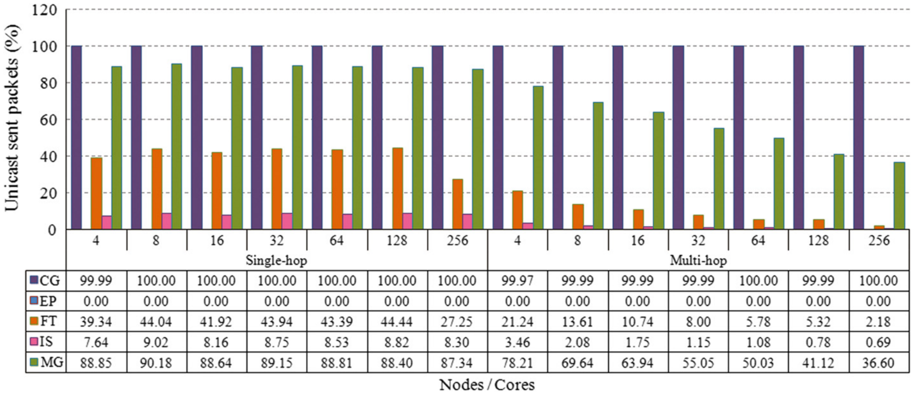
Our performance evaluation of single-hop WiNoCs is based on unicast communication used for 1:1 and N:1 communication patterns and broadcast communication for 1:N and N:N patterns for the different workloads.

The results presented in the following subsections depict single-hop and multi-hop WiNoC simulations with 4, 8, 16, 32, 64, 128, and 256 cores. These results were generated according to the previously defined evaluation metrics and evaluated taking into consideration the characteristics of the simulated architectures, along with the behavior of each workload. The proposed WiNoCs were evaluated with different sizes to investigate the architectures' scalability. Multi-hop simulations have a non-deterministic behavior due to the Fast Broadcast module. For this reason, each scenario (architecture/workload) was simulated 33 times. All scenarios have a 99 % confidence interval.

Packets sent and lost

Figure 3 allows us to infer, from the high percentage of unicast packets sent for the CG workload, that point-to-point communications are dominant. Even though the figure shows that, for networks with eight or more nodes and single-hop architecture, communication is 100 % unicast, broadcast communication (Fig. 4) is also present in these scenarios but had their percentages rounded down to zero for being too small compared to the total. As expected for this workload, the percentage of sent packets of both types is very close for both architectures (single- and multi-hop), which can be explained by the almost complete lack of broadcast communication.

Communication in the CG workload occurs predominantly between adjacent network nodes. Figure 5 shows a maximum packet loss of 0.02 % for single-hop WiNoCs, which shows that even though communication was concentrated around a few node pairs, no bottlenecks were



	4	8	16	32	64	128	256	4	8	16	32	64	128	256
	Single-hop							Multi-hop						
CG	99.99	100.00	100.00	100.00	100.00	100.00	100.00	99.97	99.99	99.99	99.99	100.00	99.99	100.00
EP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
FT	39.34	44.04	41.92	43.94	43.39	44.44	27.25	21.24	13.61	10.74	8.00	5.78	5.32	2.18
IS	7.64	9.02	8.16	8.75	8.53	8.82	8.30	3.46	2.08	1.75	1.15	1.08	0.78	0.69
MG	88.85	90.18	88.64	89.15	88.81	88.40	87.34	78.21	69.64	63.94	55.05	50.03	41.12	36.60

Fig. 3 Packets sent via unicast

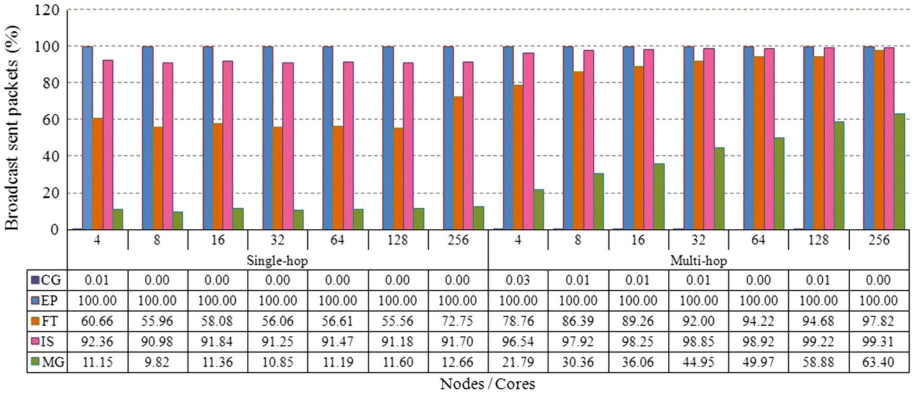
generated for the workload. For the multi-hop architectures with non-quadratic mesh topologies (8, 32, and 128 cores), loss rates of up to 50 % were observed, which occur due to nodes being busier making packets hop.

In spite of the amount of communication for the EP workload not being very significant when compared to the remaining studied workloads, it is important to analyze them because communication is 100 % broadcast, as shown in Fig. 4. The single-hop WiNoC architecture is shown to be extremely favorable, relative to packet losses, to workloads that operate exclusively with broadcast communication, since no losses were detected in simulations of these types of workload (Fig. 6). Differently, due to routers being burdened with retransmissions and greater network overload, 9.92 % of the packets were lost in the 256-core multi-hop architecture. Still in accordance with Fig. 6, the multi-hop WiNoC architecture is not scalable regarding packet losses.

The FT workload is composed by over 55 % of broadcast transmissions. Unicast transmissions are generated by the N:1 collective communication pattern. In this pattern, N unicast transmissions are carried on at the same time for a single destination. As can be seen in Figs. 3 and 4, for single-hop WiNoC scenarios, the percentage of sent packets of each type is more balanced than in the multi-hop architecture, in which each broadcast communication generates many packet transmissions, vastly increasing the percentage of broadcast packets sent.

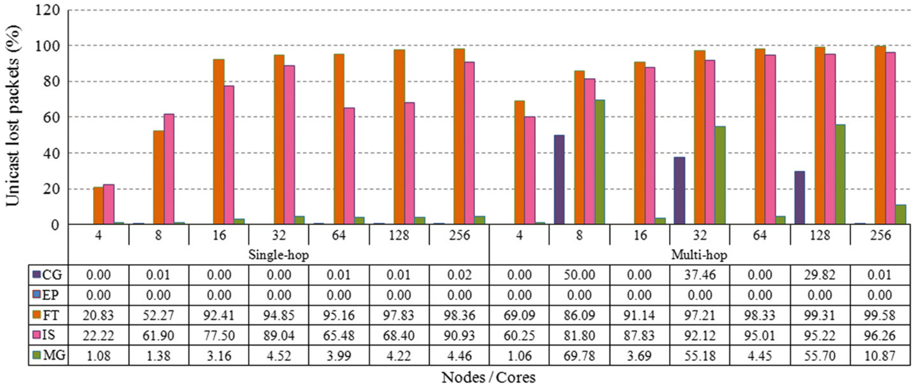
Like with the EP workload, single-hop WiNoCs have excellent broadcast performance with the FT workload since there were no packet losses, as seen in Fig. 6. However, unicast communication (Fig. 5) suffered from elevated, increasing losses. Such losses derived from the nodes competing to send packets to the destination node. A node cannot receive more than one packet at a time, leading to just one of the N packets arriving at a given moment in an N:1 communication to be received. Even then, the single-hop WiNoC had fewer lost packets than the multi-hop, which has increased competition for routers and network traffic due to packets being resent.

In the IS workload, communication is predominantly of the broadcast type. As Fig. 4 indicates, this predominance



	4	8	16	32	64	128	256	4	8	16	32	64	128	256
	Single-hop							Multi-hop						
CG	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.01	0.01	0.01	0.00	0.01	0.00
EP	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
FT	60.66	55.96	58.08	56.06	56.61	55.56	72.75	78.76	86.39	89.26	92.00	94.22	94.68	97.82
IS	92.36	90.98	91.84	91.25	91.47	91.18	91.70	96.54	97.92	98.25	98.85	98.92	99.22	99.31
MG	11.15	9.82	11.36	10.85	11.19	11.60	12.66	21.79	30.36	36.06	44.95	49.97	58.88	63.40

Fig. 4 Packets sent via broadcast



	Single-hop 4	8	16	32	64	128	256	Multi-hop 4	8	16	32	64	128	256
■CG	0.00	0.01	0.00	0.00	0.01	0.01	0.02	0.00	50.00	0.00	37.46	0.00	29.82	0.01
■EP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
■FT	20.83	52.27	92.41	94.85	95.16	97.83	98.36	69.09	86.09	91.14	97.21	98.33	99.31	99.58
■IS	22.22	61.90	77.50	89.04	65.48	68.40	90.93	60.25	81.80	87.83	92.12	95.01	95.22	96.26
■MG	1.08	1.38	3.16	4.52	3.99	4.22	4.46	1.06	69.78	3.69	55.18	4.45	55.70	10.87

Fig. 5 Packet loss in unicast communication

is greater than 90.98 % for the single-hop WiNoC scenarios. The workload's unicast communication (Fig. 3) belongs to the 1:1 and N:1 collective communication patterns, the latter being the most common. Broadcast communication rates are even higher for the multi-hop WiNoC, with a minimum rate of 96.54 %.

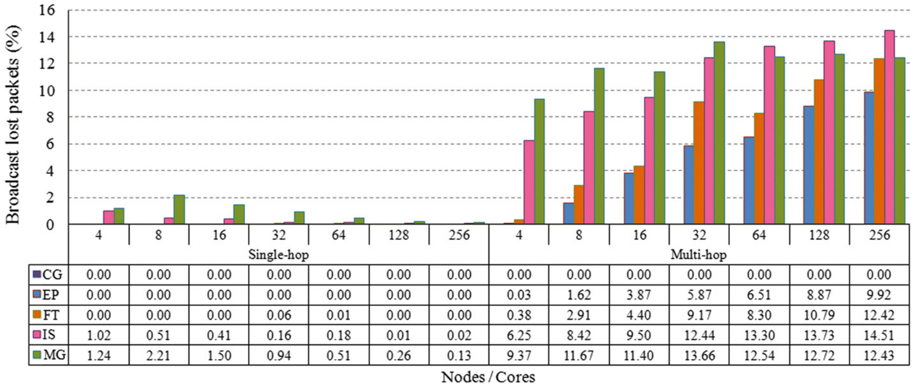
The IS workload also suffers from high packet losses, as a consequence of transmissions pertaining to the N:1 collective pattern, as evidenced by Fig. 5. For this workload, the increase in communication incurs a loss, albeit discrete, of broadcast-derived packets. Packet losses for the single-hop WiNoCs were smaller for both unicast and broadcast communication when compared to the multi-hop networks.

Still according to Figs. 3 and 4 and as opposed to the FT and IS workloads, the MG workload is predominantly comprised by unicast communications, a small portion of which are N:1. Such predominance is more evident in the single-hop WiNoC, in which broadcast communications occur in a single hop.

Packet losses referent to the unicast communications for the MG workload, as seen in Fig. 5, come from N:1 transmissions. The broadcast-derived losses, on the other hand, happen as a consequence of the increased number of communications, which provokes more packet collisions and increases competition for the transmission medium. For the MG workload, the single-hop WiNoC presented a smaller loss percentage than the multi-hop architecture. This fact can be justified by the less intense network overload caused by single-hop broadcasts.

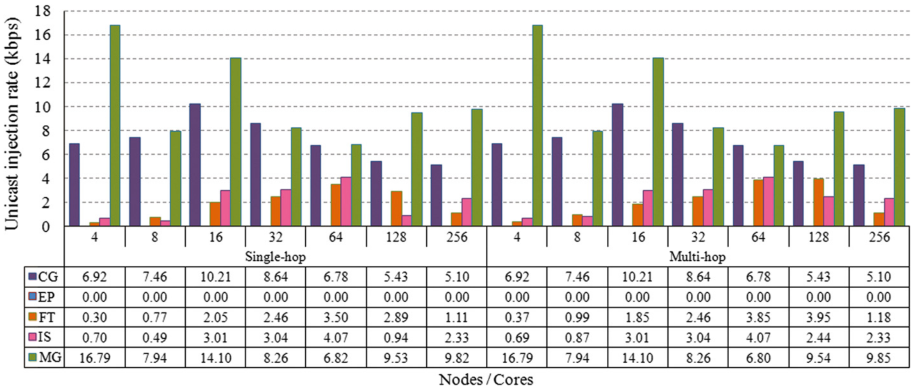
Injection rate and throughput

For the CG workload in scenarios with more than 16 nodes, unicast communication is concentrated in the beginning of simulations and becomes more uniform and sparse during the rest of the time. The fact that this peak exists causes communication to be reduced during most of the time, consequently reducing the average injection rate in the network. This explains the drop in injection rate for both architectures observed in Fig. 7. Concerning broadcast communication (Fig. 8), the scenarios for the single-hop architecture have a smaller injection rate because retransmissions that would cause more packets to be injected into the network are avoided.



	Single-hop 4	8	16	32	64	128	256	Multi-hop 4	8	16	32	64	128	256
■CG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
■EP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	1.62	3.87	5.87	6.51	8.87	9.92
■FT	0.00	0.00	0.00	0.06	0.01	0.00	0.00	0.38	2.91	4.40	9.17	8.30	10.79	12.42
■IS	1.02	0.51	0.41	0.16	0.18	0.01	0.02	6.25	8.42	9.50	12.44	13.30	13.73	14.51
■MG	1.24	2.21	1.50	0.94	0.51	0.26	0.13	9.37	11.67	11.40	13.66	12.54	12.72	12.43

Fig. 6 Packet loss in broadcast communication



	4	8	16	32	64	128	256	4	8	16	32	64	128	256
	Single-hop							Multi-hop						
■CG	6.92	7.46	10.21	8.64	6.78	5.43	5.10	6.92	7.46	10.21	8.64	6.78	5.43	5.10
■EP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
■FT	0.30	0.77	2.05	2.46	3.50	2.89	1.11	0.37	0.99	1.85	2.46	3.85	3.95	1.18
■IS	0.70	0.49	3.01	3.04	4.07	0.94	2.33	0.69	0.87	3.01	3.04	4.07	2.44	2.33
■MG	16.79	7.94	14.10	8.26	6.82	9.53	9.82	16.79	7.94	14.10	8.26	6.80	9.54	9.85

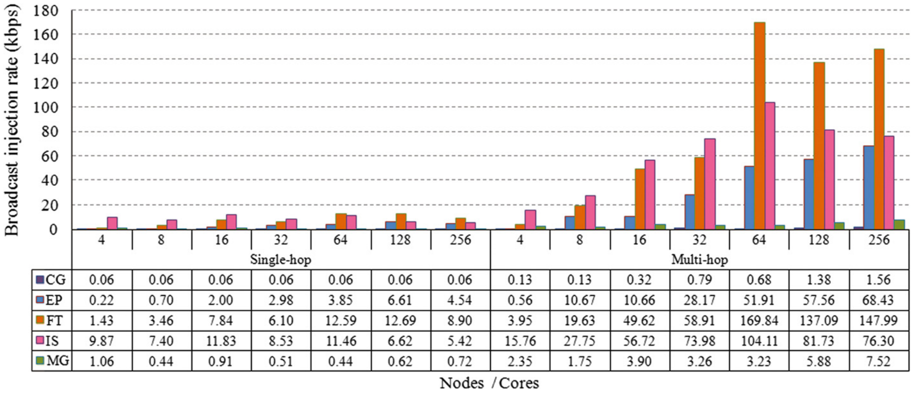
Fig. 7 Unicast injection rate

Figure 9 shows that, concerning network throughput for the CG workload, the single-hop WiNoC had excellent performance by keeping throughput rates equal to the injection rates for unicast communication. For broadcast communication (Fig. 10), the throughput is naturally higher because each transmitted message is received by every node in the network. In the scenarios with 4, 8, 16, 32, 64, and 128 routers, throughput for the single-hop architecture is smaller than for multi-hop because there are no retransmissions for packets to reach every node. The network with 256 routers, however, has higher throughput in the single-hop architecture due to the higher number of packets from broadcast communication being received at the same time by more nodes.

Figure 8 allows us to observe that, for the EP workload, the increase in communication as the number of network nodes grow causes the injection rate to also grow. The reduction in injection rate for the 256-node single-hop WiNoC occurs due to the slight peaks in communication distribution during simulation time, which mildly reduces average injection rates. As evidenced by Fig. 8, packet retransmissions generated by the multiple hops in multi-hop architectures cause the injection rates to be much higher in the corresponding scenarios in comparison with single-hop architectures.

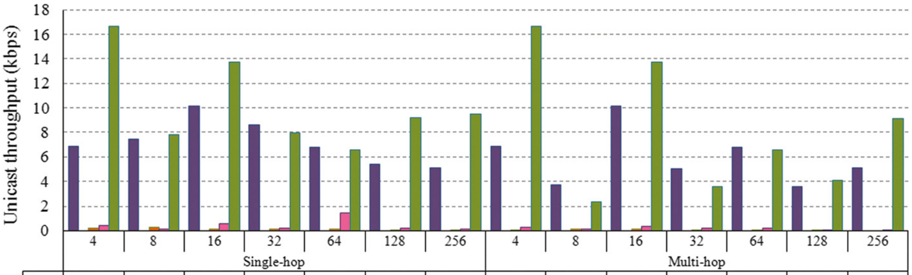
Throughput deriving from broadcast communication for the EP workload also grows as the number of nodes is augmented. In the single-hop WiNoC, this growth is up to 200 times higher than the injection rate. This behavior is explained by every node receiving every packet sent in the network. All of the routers in a single-hop WiNoC simultaneously receive a broadcast message when it is transmitted. Therefore, throughput is higher than in a multi-hop network for sizes starting from 64 nodes, considerably increasing with the addition of more nodes. For the scenarios simulating networks with 4–32 nodes, packet retransmissions make the multi-hop WiNoC have more throughput.

Figure 7 shows, additionally, that injection rates stemming from unicast communication for the FT workload are extremely low, reaching a maximum of 3.5 kbps in the single-hop WiNoC. This occurs due to the small amount of communication of this type. Since broadcast communication is also not abundant in this workload, injection



	4	8	16	32	64	128	256	4	8	16	32	64	128	256
	Single-hop							Multi-hop						
■CG	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.13	0.13	0.32	0.79	0.68	1.38	1.56
■EP	0.22	0.70	2.00	2.98	3.85	6.61	4.54	0.56	10.67	10.66	28.17	51.91	57.56	68.43
■FT	1.43	3.46	7.84	6.10	12.59	12.69	8.90	3.95	19.63	49.62	58.91	169.84	137.09	147.99
■IS	9.87	7.40	11.83	8.53	11.46	6.62	5.42	15.76	27.75	56.72	73.98	104.11	81.73	76.30
■MG	1.06	0.44	0.91	0.51	0.44	0.62	0.72	2.35	1.75	3.90	3.26	3.23	5.88	7.52

Fig. 8 Broadcast injection rate



	Single-hop							Multi-hop						
	4	8	16	32	64	128	256	4	8	16	32	64	128	256
■CG	6.92	7.46	10.21	8.64	6.78	5.43	5.10	6.92	3.73	10.21	5.08	6.78	3.57	5.10
■EP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
■FT	0.24	0.34	0.13	0.13	0.16	0.06	0.02	0.11	0.13	0.16	0.08	0.07	0.03	0.00
■IS	0.45	0.17	0.58	0.26	1.45	0.22	0.15	0.28	0.13	0.35	0.23	0.21	0.12	0.09
■MG	16.65	7.86	13.78	7.99	6.62	9.25	9.52	16.66	2.35	13.73	3.60	6.59	4.12	9.12

Nodes / Cores

Fig. 9 Unicast throughput

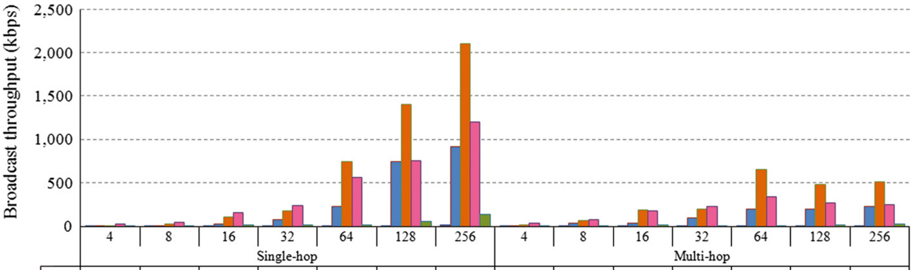
rates (Fig. 8) derived from them are also not elevated, although greater, reaching 12.69 kbps. For the multi-hop WiNoC, injection rates for broadcast communication are notably higher due to the multiple hops.

Throughputs for unicast communication in the FT workload, as seen in Fig. 9, were naturally smaller relative to injection rate, due to high packet losses. Despite low unicast throughput rates, broadcast communication achieved a high growth for the scenarios with the most nodes. The increase is drastic when compared to injection rates. On the 256-node single-hop WiNoC, it was over 237 times higher, which represents an increase by a factor of almost the number of existing nodes. This corresponds to the expected behavior for broadcast communication. As was the case with the EP workload, throughput for the FT kernel in single-hop networks with 64 or more routers is higher than in corresponding multi-hop networks, because of broadcast communication.

Figure 7 indicates that injection rates derived from unicast communication for the IS kernel are low, but are higher than the ones obtained with FT, which is justified by the presence of 1:1 communication. Broadcast communication obtained higher rates due to being predominant in this workload. Multi-hop WiNoCs have higher injection rates with IS because of the multiple hops.

Throughput for the IS workload has the same behavior as for FT. The reasons behind this are aforementioned N:1- and broadcast-related characteristics. For this workload, the single-hop WiNoC obtained better throughput rates in scenarios with more than 16 routers. Such a result is natural due to the increase in broadcast packet reception caused by the greater number of network nodes.

For MG workloads, communication was more concentrated during the beginning of the simulations and tended to be more sparse in larger, longer-lasting network simulations. This is the reason why injection rates for unicast communication, shown in Fig. 7, are irregular throughout the various network sizes. The same pattern can be observed in broadcast communication (Fig. 8). Single- and multi-hop WiNoCs had practically the same injection rate for unicast communication, due to low interference from



	Single-hop							Multi-hop						
	4	8	16	32	64	128	256	4	8	16	32	64	128	256
■CG	0.19	0.45	0.96	1.98	4.03	8.13	16.32	0.38	0.50	1.65	4.59	4.37	9.32	10.79
■EP	0.68	4.94	26.28	72.81	226.07	750.65	922.63	1.52	34.21	34.97	98.20	198.20	198.33	229.08
■FT	4.13	24.10	106.75	175.95	749.31	1405.77	2112.62	11.45	66.18	191.32	200.37	660.17	483.96	513.29
■IS	28.66	50.10	157.44	236.51	569.13	760.18	1203.50	39.07	80.21	181.72	233.30	339.29	271.53	250.85
■MG	3.07	2.63	11.08	12.21	21.10	58.75	135.64	5.31	4.42	11.36	9.21	9.98	18.47	24.89

Nodes / Cores

Fig. 10 Broadcast throughput

broadcast communication. Regarding broadcasts, injection rates were lower in the single-hop WiNoC than in the multi-hop network.

In the single-hop WiNoC, unicast communication from the MG workloads had very close throughput and injection rate values. Throughput is the smallest of the two, due to packet losses in N:1 communication. Concerning broadcast, throughput was similar to what was observed for other workloads, that is, it grows as the network grows, for the same aforementioned reasons. For the MG workload, throughput for networks with more routers is also higher in the single-hop architecture, since more packets are received simultaneously.

Communication delay and energy consumption

Figure 11 shows average delay values for unicast communication in both single- and multi-hop WiNoCs. We can see both architectures presented a non-uniform behavior pertaining to average delay values for unicast communication in the CG workloads, relative to node quantity. This happens because communication occurs among just a few nodes in the network, and the distance between these nodes can vary according to the way in which they were distributed in the architecture. It can also be noted that the single-hop WiNoC had better performance, because it relays messages directly to the destination nodes.

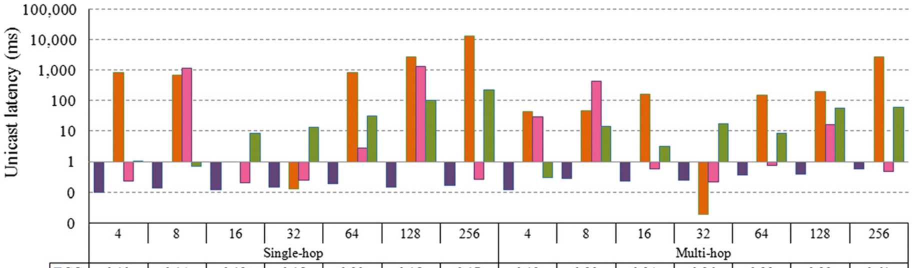
For the FT and IS workloads, delay behavior in unicast communication is also non-uniform. This is justified based on characteristics of N:1 communication. During packet exchanges, if more than one node wishes to send a packet to a destination node at the same time, a selection to decide which packet will be received is carried out, since the router can only receive one packet a time. The other packets are then discarded, which affects packet loss percentages, as previously stated. When the selected packet originates from the receiving router itself, the delay is practically null but otherwise a negotiation between the involved nodes takes place, increasing communication delay.

Smaller delay values were obtained by the multi-hop WiNoC in unicast communication for these workloads. The high delay in the single-hop WiNoC for this type of communication refers to more transmissions, since the multi-hop architecture had higher packet loss rates. Besides, delay also increases with high medium utilization, caused by unicast messages being emitted to the entire network, independently of the positions of the origin and destination routers.

Delay values for unicast communication using the IS workloads were higher in the networks of sizes 8 and 128, which occurs due to negotiations to select which requesting node will be granted permission to send a packet in N:1 communication. Possibly, since communications of this type were lost in networks of other sizes, delay values were lower, pertaining only to 1:1 communication.

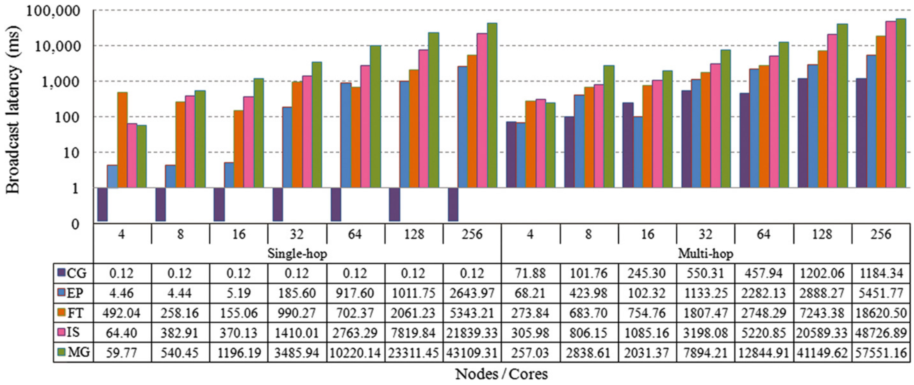
For the MG workload, unicast delay values for the single-hop WiNoC begin increasing with network sizes in networks with 8 or more routers. It is an expected increase, due to the high number of packet exchanges. This fact is also observed in the multi-hop WiNoC, although it is less expressive in the quadratic mesh topology (4, 16, 64, and 256 routers) scenarios. Even though the CG workload has more communication, delays for the MG workload are bigger because it is executed in less time, which increases competition for the transmission medium. Like in the FT and IS workloads, delays in the single-hop architecture surpassed those in the multi-hop WiNoCs.

Figure 12 shows the average delay values for both single- and multi-hop architectures in broadcast communication. Delay values for every workload increased with network size and were proportional to the amount of communication that took place. Delays for the CG workload are significantly lower than others in the single-hop WiNoC, due to the small number of broadcast transmissions. The



	4	8	16	32	64	128	256	4	8	16	32	64	128	256
	Single-hop							Multi-hop						
■CG	0.10	0.14	0.12	0.15	0.20	0.15	0.17	0.12	0.29	0.24	0.26	0.38	0.39	0.61
■EP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
■FT	864.59	690.56	0.00	0.13	836.93	2769.44	13254.47	45.12	48.54	161.31	0.02	159.60	201.08	2820.61
■IS	0.24	1154.56	0.21	0.25	2.93	1375.45	0.28	29.63	445.42	0.61	0.22	0.78	16.83	0.48
■MG	1.10	0.74	8.58	14.15	33.27	107.06	223.87	0.31	14.76	3.17	17.67	8.98	58.75	62.90

Fig. 11 Unicast communication delay



	Single-hop							Multi-hop						
	4	8	16	32	64	128	256	4	8	16	32	64	128	256
■CG	0.12	0.12	0.12	0.12	0.12	0.12	0.12	71.88	101.76	245.30	550.31	457.94	1202.06	1184.34
■EP	4.46	4.44	5.19	185.60	917.60	1011.75	2643.97	68.21	423.98	102.32	1133.25	2282.13	2888.27	5451.77
■FT	492.04	258.16	155.06	990.27	702.37	2061.23	5343.21	273.84	683.70	754.76	1807.47	2748.29	7243.38	18620.50
■IS	64.40	382.91	370.13	1410.01	2763.29	7819.84	21839.33	305.98	806.15	1085.16	3198.08	5220.85	20589.33	48726.89
■MG	59.77	540.45	1196.19	3485.94	10220.14	23311.45	43109.31	257.03	2838.61	2031.37	7894.21	12844.91	41149.62	57551.16

Fig. 12 Broadcast communication delay

single-hop WiNoC had considerably smaller delay values from broadcast transmissions for all workloads, as a consequence of carrying them out in a single hop.

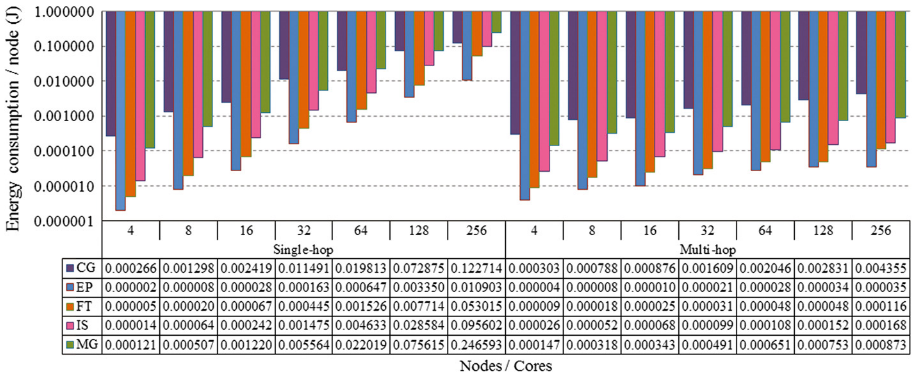
As a consequence of the increase in transmitter, receiver, and signal power, so that single-hop communication is viable, energy consumption values are considerably higher in single-hop WiNoCs when compared to multi-hop networks. Figure 13 shows average energy consumption values for each node in the single- and multi-hop WiNoCs. Energy consumption in the simulations increases with network size, and consumption per workload is associated with the number of packet transmissions in each of them. This occurs because power values are the same no matter what the distance between the nodes may be, so power values must be set to accommodate the highest distance.

Scalability evaluation

The single-hop WiNoC was more scalable relative to packet losses in 1:1 unicast communication, presenting no losses for the CG workload and slightly increased losses for the MG workload. In the multi-hop architecture, losses got bigger as the number of cores was increased, but scalability is more critical in nonsquare-shaped mesh networks, which have significantly larger packet loss rates. Pertaining to N:1 communication, losses grew significantly as networks got larger in both architectures, but the growth was steeper in multi-hop architectures, rendering them less scalable.

Scalability relative to predominantly 1:1 unicast communication latency (CG) is better in the single-hop WiNoC. Conversely, the MG workload scaled better in the multi-hop WiNoC, but since packet loss rates were considerably smaller in the single-hop architecture, the better scalability is not regarded as an advantage, because the increased latency in the single-hop architecture is explained by delays in the delivery of packets that were lost in the multi-hop network.

In broadcast communication, the scalability of the single-hop WiNoC relative to packet losses is excellent, given the absence of packet losses for the EP workload and, respectively, the low and decreasing loss rates for the IS and MG workloads as the number of network nodes increases. Loss rates on the multi-hop WiNoC, on the other hand, increase as the network grows. Scalability



	Single-hop							Multi-hop						
	4	8	16	32	64	128	256	4	8	16	32	64	128	256
■CG	0.000266	0.001298	0.002419	0.011491	0.019813	0.072875	0.122714	0.000303	0.000788	0.000876	0.001609	0.002046	0.002831	0.004355
■EP	0.000002	0.000008	0.000028	0.000163	0.000647	0.003350	0.010903	0.000004	0.000008	0.000010	0.000021	0.000028	0.000034	0.000035
■FT	0.000005	0.000020	0.000067	0.000445	0.001526	0.007714	0.053015	0.000009	0.000018	0.000025	0.000031	0.000048	0.000048	0.000116
■IS	0.000014	0.000064	0.000242	0.001475	0.004633	0.028584	0.095602	0.000026	0.000052	0.000068	0.000099	0.000108	0.000152	0.000168
■MG	0.000121	0.000507	0.001220	0.005564	0.022019	0.075615	0.246593	0.000147	0.000318	0.000343	0.000491	0.000651	0.000753	0.000873

Fig. 13 Node energy consumption

relative to broadcast communication in workloads that rely predominantly on this pattern (EP and IS) is better in the single-hop WiNoC, due to the smaller inclination of its curve as the number of cores increases.

In addition to smaller energy consumption, the multi-hop WiNoC's inclination curve is smoother as the number of cores increases, showing its better scalability compared to the single-hop network.

Overall evaluation

WiNoC simulation results for workloads using predominantly unicast communication (CG and MG) showed that the single-hop WiNoCs have excellent performance, with low packet loss and delay values. Even though the MG workload had higher delay for the single-hop architecture (Fig. 11), this was not considered a negative point compared to the multi-hop architecture, since delay values for broadcast packets (Fig. 12) still show a better performance for single-hop architectures.

For the FT and IS workloads, high packet losses in unicast communication were recorded for both architectures, being more accentuated in the multi-hop WiNoC. The registered delays were also high, with the single-hop architecture being the slowest of the two. The high latency values can be attributed to N:1 communication, whose packets arrive simultaneously, and the higher values observed in the single-hop architecture can be explained by its lower packet loss rates.

Regarding broadcast communication, the simulated single-hop WiNoC architecture was shown to be extremely favorable to them. For the EP workload, which works exclusively with broadcasts, not a single packet was lost, and for the MG workload, the maximum registered loss rate was 2.21 %. In all simulated scenarios, losses were lower for the single-hop architecture, considering equivalent scenarios for both. The single-hop WiNoC also had better broadcast delay results for every scenario and workload.

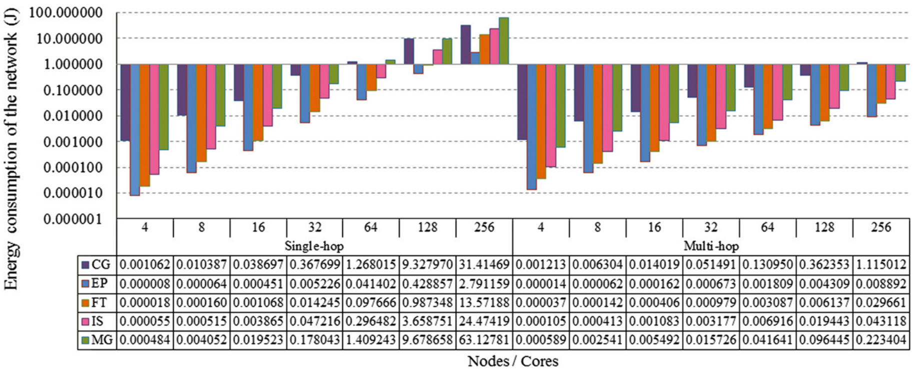
Energy consumption was observed to be a weak point of single-hop WiNoCs, demanding further studies seeking to find improvements. For the most broadcast-heavy workload (MG), energy consumption reached 63.12 J for the 256-node network (Fig. 14). Consumption was higher in the single-hop architecture in every scenario and for every workload. The higher energy consumption is justified by the use of higher power values in the transmitters, receivers, and signals, which is necessary to enable communications in a single hop. Besides, the same power values are used in every unicast communication, not taking distance between nodes into account.

Conclusions

Given the desire to execute increasingly complex programs in feasible time and since applications tend to explore parallelism more and more, it is speculated that tens or hundreds of cores may be incorporated to the same processor in the future (many-core architecture). As a way of assuring better performance for many-core processors, WiNoCs were conceived. This solution consists of connecting routers that communicate by means of radio antennas to processor cores. This way, packets are sent by routers to their neighbors until they get to their destination.

To improve the performance of traditional WiNoC architectures (multi-hop), especially concerning delays in packet delivery, single-hop WiNoCs were proposed, that is, WiNoCs in which communications are conducted in a single hop. Contrarily, in this new approach, transmitters, receivers, and signals must use more power, increasing total power consumption. In this article, two WiNoC architectures (one single-hop architecture and one multi-hop architecture) were presented and evaluated.

To accomplish that, a simulation environment created using the NS-2 simulator for the intra-chip context was used. Simulations were conducted using NPB kernels



	Single-hop							Multi-hop						
	4	8	16	32	64	128	256	4	8	16	32	64	128	256
■ CG	0.001062	0.010387	0.038697	0.367699	1.268015	9.327970	31.41469	0.001213	0.006304	0.014019	0.051491	0.130950	0.362353	1.115012
■ EP	0.000008	0.000064	0.000451	0.005226	0.041402	0.428857	2.791159	0.000014	0.000062	0.000162	0.000673	0.001809	0.004309	0.008892
■ FT	0.000018	0.000160	0.001068	0.014245	0.097666	0.987348	13.57188	0.000037	0.000142	0.000406	0.000979	0.003087	0.006137	0.029661
■ IS	0.000055	0.000515	0.003865	0.047216	0.296482	3.658751	24.47419	0.000105	0.000413	0.001083	0.003177	0.006916	0.019443	0.043118
■ MG	0.000484	0.004052	0.019523	0.178043	1.409243	9.678658	63.12781	0.000589	0.002541	0.005492	0.015726	0.041641	0.096445	0.223404

Fig. 14 Total network energy consumption

(CG, EP, FT, IS, and MG). The 1:1 and N:1 collective communication patterns existent in the workloads were simulated using unicast communication, and the 1:N and N:N patterns were simulated using broadcast communication.

Due to the many-core processor architecture context, WiNoCs gained traction in the support of parallel application threads. In this way, it is important to highlight the correlation between performance and energy consumption with collective communication patterns, as discussed in the "Results and discussion" section, as a crucial factor for project decisions. Although both networks are scalable due to the nature of the architectural project, viability is linked to the scalability reachable by parallel applications. This is a very particular situation of the studied context, since a single parallel application uses the entire WiNoC, contrary to what happens in distributed systems in which different applications communicate through the wireless network. In the current high-performance computing scenario, energy efficiency is a vital factor for a many-core processor project to be viable. Although single-hop and multi-hop WiNoCs have shown different behaviors for the same workloads, it is not possible to discard the use of one or the other. Thus, design compromises can lead to a hybrid and reconfigurable WiNoC project or even a lower performance project in order to attain energy consumption gains.

As future work, ways of reducing single-hop WiNoC energy consumption must be investigated, testing other topologies such as 3-D mesh, which allows for a reduction in power by diminishing the distance between nodes. Other options to be explored are hybrid WiNoCs, which mix wired and wireless transmissions and higher-powered multi-hop WiNoCs that are able to reduce the number of hops in transmissions. Characteristics such as antenna size and frequency must also be taken into account as a way of reducing the power necessary to accomplish communication in a single hop, consequently reducing total energy consumption.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors read and approved the final manuscript.

Acknowledgements

This work was partially supported by FIP PUC Minas, FAPEMIG, and CNPq. Our special thanks to Matheus Queiroz who helped us translate and review the article.

Received: 25 January 2014 Accepted: 9 June 2015
Published online: 27 June 2015

References

1. Alves MAZ, Freitas HC, Navaux POA (2011) High latency and contention on shared l2-cache for many-core architectures. Parallel Process Lett 21(01):85–106
2. Borkar S, Chien AA (2011) The future of microprocessors. Commun ACM 54(5):67–77
3. Fan D, Zhang H, Wang D, Ye X, Song F, Li G, Sun N (2012) Godson-t: an efficient many-core processor exploring thread-level parallelism. Micro IEEE 32(2):38–47
4. Duato J, Yalamanchili S, Lionel N (2002) Interconnection networks. Morgan Kaufmann Publishers, San Francisco
5. Freitas HC, Schnorr LM, Alves MAZ, Navaux POA (2010) Impact of parallel workloads on NoC architecture design. In: Proc. 18th Euromicro Conference on Parallel, Distributed and Network-Based Processing. PDP, Pisa. pp 551–555. http://dx.doi.org/10.1109/PDP.2010.53
6. Kumar R, Zyuban V, Tullsen DM (2005) Interconnections in multi-core architectures: understanding mechanisms, overheads and scaling. In: Proc. 32nd International Symposium on Computer Architecture (ISCA). pp 408–419
7. Freitas HC (2009) Programmable multi-cluster NoC architecture to support collective communication patterns. Phd thesis, Universidade Federal do Rio Grande do Sul, Instituto de Informática, Programa de Pós-Graduação em Computação, Porto Alegre, Rio Grande do Sul, Brasil. http://hdl.handle.net/10183/16656
8. Ho R, Mai KW, Member S, Horowitz MA (2001) The future of wires. J Proc IEEE 89(4):490–504
9. Bjerregaard T, Mahadevan S (2006) A survey of research and practices of network-on-chip. Comput Surv 38(1):1–51
10. Sanchez D, Michelogiannakis G, Kozyrakis C (2010) An analysis of on-chip interconnection networks for large-scale chip multiprocessors. ACM Trans Archit Code Optim 7(1):4–1428
11. Oliveira PAC (2012) Performance evaluation and parallel workload characterization for wireless networks-on-chip. http://www.biblioteca.pucminas.br/teses/Informatica_OliveiraPAC_1.pdf
12. Li X (2012) Survey of wireless network-on-chip systems. Technical report, Auburn University, Technical report. Auburn University, Alabama, USA. http://www.eng.auburn.edu/~agrawvd/THESIS/LI/report.pdf
13. Zhao D (2008) Ultraperformance wireless interconnect nanonetworks for heterogeneous gigascale multi-processor SoCs. In: Proc. 2th Workshop on Chip Multiprocessor, Memory Systems and Interconnects. CMP-MSI, Beijing. pp 1–3
14. Ganguly A, Chang K, Deb S, Pande PP, Belzer B, Teuscher C (2011) Scalable hybrid wireless network-on-chip architectures for multi-core systems. J Trans Comput 60(10):1485–1502
15. Amorim AM, Freitas HC (2013) Avaliação de desepenho de redes-em-chip sem fio single-hop com nas parallel benchmarks. In: WSCAD-SSC 2013, Porto de Galinhas, Ipojuca, Pernambuco
16. Benini L, Micheli GD (2002) Networks on chips: a new SoC paradigm. J Comput 35(1):70–78
17. Ganguly A, Chang K, Pande PP, Belzer B, Nojeh A (2009) Performance evaluation of wireless networks on chip architectures. In: Proc. 10th International Symposium on Quality of Electronic Design. ISQED, San Jose. pp 350–355. http://dx.doi.org/10.1109/ISQED.2009.4810319
18. Oliveira PAC, Duarte-Figueiredo FLP, Martins CAPS, Freitas HC, Ribeiro CP, Castro M, Marangozova-Martin V, Méhaut J-F (2011) Performance evaluation of winocs for parallel workloads based on collective communications. In: Proc. IADIS Applied Computing. IADIS Applied Computing, Rio de Janeiro. pp 307–314
19. Amorim AMP, Oliveira PAC, Freitas HC (2012) Integrando traços de execução de aplicações paralelas ao network simulator para simulação de winoc. In: Anais.. WSCAD-WIC, Petrópolis. pp 1–4. Workshop de Iniciacao Cientifica, XIII Simposio em Sistemas Computacionais
20. Carloni LP, Pande P, Xie Y (2009) Networks-on-chip in emerging interconnect paradigms: advantages and challenges. In: Proc. 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip. IEEE Computer Society, Washington, DC, USA. pp 93–102. http://dx.doi.org/10.1109/NOCS.2009.5071456
21. Pande PP, Ganguly A, Chang K, Teuscher C (2009) Hybrid wireless network on chip: a new paradigm in multi-core design. In: Proc. 2th International Workshop on Network on Chip Architectures. NoCArc, New York. pp 71–76. http://dx.doi.org/10.1145/1645213.1645230
22. Deb S, Chang K, Ganguly A, Pande P (2010) Comparative performance evaluation of wireless and optical architectures. In: Proceedings of the IEEE International SOC Conference. SOCC '10. IEEE Computer Society, Las Vegas. pp 487–492. http://dx.doi.org/10.1109/SOCC.2010.5784675

23. Wang C, Hu WH, Bagherzadeh N (2011) A wireless network-on-chip design for multicore platforms. In: Proc. 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing. PDP, Ayia Napa. pp 409–416. http://dx.doi.org/10.1109/PDP.2011.37
24. Zhao D, Wang Y, Li J, Kikkawa T (2011) Design of multi-channel wireless NoC to improve on-chip communication capacity. In: Proc. 5th ACM/IEEE International Symposium on Networks-on-Chip. NOCS, Pittsburgh. pp 177–184. http://dx.doi.org/10.1145/1999946.1999975
25. NPB NAS Parallel Benchmarks. Available at http://www.nas.nasa.gov/publications/npb.html. Accessed in May.
26. NPB (2013) Problem sizes and parameters in NAS Parallel Benchmarks. Available at http://www.nas.nasa.gov/publications/npb_problem_sizes.html. Accessed in May.
27. Sun YR, Kumar S, Jantsch A (2002) Simulation and evaluation for a network on chip architecture using NS-2. In: Proceedings, 20th IEEE NorChip Conference, Copenhagen
28. Kourdy R, Yazdanpanah S, Rad MRN (2010) Using the NS-2 network simulator for evaluating multi protocol label switching in network-on-chip. In: Second International Conference on Computer Research and Development, Washington, DC, USA. pp 795–799
29. NS (2013) Network simulator (NS-2). Available at http://isi.edu/nsnam/ns/. Accessed in May.
30. NS (2013) Network simulator (NS-2) contributed code. Available at https://ant.isi.edu/nsnam/index.php/Contributed_Code#Wireless_and_Mobility. Accessed in May.
31. NS-UWB (2013) NS-2 ultra wide-band (UWB) MAC and PHY simulator. Available at http://uwb.epfl.ch/ns-2/index.html. Accessed in May.
32. Merz R, Boudec J-YL, Widmer J (2007) An architecture for wireless simulation in NS-2 applied to impulse-radio ultra-wide band networks. In: Proceedings Spring Simulation Multiconference. SpringSim, Norfolk. pp 256–263. http://dx.doi.org/10.1109/ICCRD.2010.145
33. Merz R (2008) Interference management in impulse-radio ultra-wide band networks. Phd thesis, Laboratoire pour les communications informatiques et leurs applications 2, École Polytechnique Fédérale de Lausanne (EPFL), Suisse. http://infoscience.epfl.ch/record/121463
34. FAST-BROADCAST Resources - Fast Broadcast modules for NS-2. Available at http://www.math.unipd.it/~cpalazzi/fastbroadcast.html. Accessed in May (2013)
35. IEEE 802.15 working group for WPAN. Available at http://www.ieee802.org/15. Accessed in August 2012