



A load-balanced congestion-aware routing algorithm based on time interval in wireless network-on-chip

Shokoofeh Mikaeeli Mamaghani¹ · Mohammad Ali Jabraeil Jamali²

Received: 3 September 2017 / Accepted: 29 August 2018 / Published online: 3 September 2018
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

Abstract

Network-on-chip (NoC) has been introduced to increase the performance of chip multiprocessors (CMPs) and execute parallel programs. Although NoC is known as a modular and scalable infrastructure for interconnections, there are still some challenges with conventional NoC such as high latency and power consumption due to the communication among long-distance (LD) cores. In this regard, wireless network-on-chip (WiNoC) is a potential solution that can provide high bandwidth and low latency by means of the unique features of wireless interconnects. However, wireless routers (WRs) are prone to congestion in WiNoC due to the limited number of wireless channels on a chip and shared use of these channels by all processing elements (PEs). In this study, a load-balanced time-based congestion-aware (LTCA) routing algorithm is proposed to eliminate the congestion of WRs and distribute the traffic load on the wired and wireless networks in a balanced way. LTCA is a deadlock-free routing algorithm in which only a limited number of packets are allowed to use wireless channels. The required time for transmitting the selected packets through wireless links is measured with regard to the bandwidth of the wireless channels and traffic load. Simulation results on synthetic traffic patterns and real-world 3-tuple traffic patterns indicated a considerable improvement in latency, throughput, wired and wireless link utilization and packet loss probability.

Keywords Congestion-aware routing algorithm · Wireless network-on-chip · Time interval · LTCA

1 Introduction

Silicon technologies have grown significantly in recent years. Along with the scaling down of silicon technologies, chip designs are moving towards integrating a large number of cores on a single chip (DiTomaso et al. 2011). Multicore platforms employ NoC architecture as the preferable interconnection backbone (Duraisamy et al. 2017). Furthermore, the NoC paradigm is introduced as a regular and scalable infrastructure for interconnections (Murray et al. 2016a). As the number of PEs increases and data exchanges among them grow, the wired links are becoming insufficient for handling the required data rates, and may form a bottleneck

for system performance (Matolak et al. 2012). Moreover, by increasing the NoC scale, these multi-hop wireline paths lead to high latency in a conventional NoC. In addition, as the number of PEs increases, the wired links consume considerable power in multi-hop LD communication. Therefore, to alleviate the long wire delays and high power consumption of CMPs, alternative technologies such as photonic NoC (Vantrease et al. 2008), 3D-NoC (Davis et al. 2005) and WiNoC have been introduced.

Wireless interconnects are a potential solution to save energy and reduce latency by means of unique benefits of wireless channels. Wireless links have significant benefits: high power efficiency for LD communication, reduced complexity compared to a system with wave-guides or wires and compatibility with CMOS wireless technologies (DiTomaso et al. 2015). However, it should be noted that wireless links are used as a complement in WiNoC but they do not completely replace wired links (Rayess et al. 2017). Using both wired and wireless links and, also, the division of the network into smaller subnets can improve system performance. Wireless links play an important role in LD communications

✉ Mohammad Ali Jabraeil Jamali
m_jamali@itrc.ac.ir

Shokoofeh Mikaeeli Mamaghani
Sh.mik1364@gmail.com

¹ Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran

² Department of Computer Engineering, Shabestar Branch, Islamic Azad University, Shabestar, Iran

by reducing the hop distances among LD cores. Consequently, it leads to low latency in WiNoC.

With regard to the limited number of wireless channels on a chip and given the different bandwidths of wired and wireless links, congestion control and routing in WiNoC require more attention. Hence, some methods are needed for efficiently allocating the valuable wireless channel to the packets. Furthermore, WRs are prone to congestion due to the shared use of wireless links by all the PEs (Hu et al. 2015). The congestion probability in WiNoC might be even more than that of conventional NoC (Rezaei et al. 2016). Although low latency is regarded as one of the major objectives in WiNoC, the issue of WRs congestion leads to high latency in WiNoC. Therefore, the congestion-aware architectures or congestion-aware routing algorithms are required to prevent the congestion of WRs; otherwise, the overuse of wireless channels leads to the congestion on WRs and on the paths leading to them in WiNoC.

In recent years, various methods have been proposed to reduce the congestion of WRs. Some of them tried to solve the congestion problem by means of additional hardware resources and by increasing the size of the buffers around WRs (Wang et al. 2012). Some other studies tried to eliminate the congestion of WRs by developing congestion-aware routing algorithms. When congestion-aware routing algorithms are combined with other methods such as the WRs replacement (Wang et al. 2012; Hu et al. 2015; Rezaei et al. 2016), application mapping (Rezaei et al. 2015, 2017) and task migration (Wang et al. 2012; Rezaei et al. 2015, 2016, 2017), they are usually able to reduce congestion and improve latency.

In this paper, a load-balanced congestion-aware routing algorithm is proposed which uses the time interval-based approach for allocating wireless channels to the LD cores. Without increasing hardware resources of the system, the proposed algorithm is able to reduce the congestion of WRs and improve latency. This algorithm selects only a few LD packets to use wireless links with respect to some criteria such as the wired hop distance between the source and destination, the number of the hybrid wired/wireless hops, the network frequency, and the transmission rate of wired and wireless channels. Then, with regard to the bandwidth of wireless links and the size of packets, the required time for passing authorized packets through wireless channels is measured. In this measurement, the traffic load is also considered to balance the network load between the wired and wireless channels. During this time span, all the new packets injected into the network adopt the conventional wired routing. Hence, by considering the network load and transmitting an appropriate number of packets towards WRs, LTCA algorithm balances the utilization of wired/wireless links and reduces the congestion of WRs. The followings are considered as the contributions of this paper:

- Taking into account some network parameters such as the network frequency, wired and wireless hop distance between the source and destination and the bandwidth of the wired and wireless links, the proposed method selects the packets intelligently for transmitting through the WRs.
- By applying time interval and considering network load, LTCA balances the load on the wired and wireless networks and achieves better performance than the other methods.
- The proposed algorithm has the minimum hardware requirement in comparison with other related work.

The rest of the paper is organized as follows: Sect. 2 provides a brief review of the related work and congestion-aware routing algorithms. Section 3 elaborates the load-balanced interval-based congestion-aware routing algorithm to reduce WRs congestion. Section 4 reports the performance evaluation of the proposed algorithm. Finally, Sect. 5 provides the conclusion of the paper.

2 Related work

In line with the development of technology, the number of cores within a chip increases. As a notable instance of multi-core systems, the 256-core processor of the Intel company in which 22 nm technology has been applied can be mentioned (Chen et al. 2015); or the 48-core cloud computer (Howard et al. 2011) and 100-core processors of Tiler company are other instances of multi-core systems. In all these examples, NoC is considered as a regular infrastructure to establish interconnections among different cores. However, the system performance degrades by increasing latency and energy consumption due to the multi-hop communications between LD cores in conventional NoC. WiNoC utilizes high-bandwidth wireless channels on a chip to minimize the long hop distances among LD cores. It results in considerable improvement in latency and energy consumption.

Wireless communications in WiNoC are carried out using antenna and transceivers (Murray et al. 2016b). Different types of antennas are used in WiNoC, such as millimeter-wave antennas and carbon nanotube (CNT) antennas. In Lin et al. (2007), millimeter-wave antennas are introduced as efficient antennas for long-range communications on the chip. They operate at frequencies of tens of GHz. In Lee et al. (2009), the antennas located on a polyamide layer have extended communication up to 1 cm. In addition to the above-mentioned technologies, other antennas have been designed which can function in different frequency channels (Deb et al. 2013) and prevent the interference between channels. At the architecture level, on-off key modulation is a good choice since it has a simple and uncomplicated

circuit design with little area overhead and efficient power consumption (Murray et al. 2016b). In DiTomaso et al. (2011), OOK-based transceivers are used which provide 16 non-overlapping wireless channels with a total bandwidth of 512 Gbps. Whereas millimeter-wave antennas achieve a bandwidth of 10 s of GHz, CNTs are antennas which operate at terahertz frequencies (Kempa et al. 2007) and achieve a bandwidth of around 500 GHz. Researchers in Lee et al. (2009) used sub-THz antennas in a polyamide layer to provide 16 non-overlapping channels at the frequency of 100–500 GHz where each channel includes 20 Gbps data rate with 10–20 mm transmission range. According to the various technologies which were mentioned above, different numbers of wireless channels with different bandwidth can be used on a chip.

In addition to various wireless technologies, different kinds of topologies have been used in WiNoC. One of the most common topologies in WiNoC is a two-dimensional mesh topology. According to the different kinds of architectures presented in various literature (DiTomaso et al. 2015; Hu et al. 2015; Rezaei et al. 2017), wireless channels are shared by a group of PEs in WiNoC. Due to the limited number of wireless channels on a chip and shared use of these channels by all PEs, it is possible for various LD cores to select the same wireless channel for transmitting their packets, which will cause the hotspots effects and lead to the congestion at the wireless node. The hotspot effects could make the load distribution extremely unbalanced (Qiuli et al. 2018) and lead to the congestion of overloaded WRs. This will eventually result in a large amount of packet loss and degrade the network performance (Qiuli et al. 2018). During the congestion, nodes have to drop packets to accommodate the new incoming packets (Bansal et al. 2018). Therefore, different algorithms have been proposed to reduce the congestion of WRs and balance the load distribution between the wired and wireless networks. One of the methods of congestion control in conventional NoC is task mapping (Chou and Marculescu 2008; Carvalho et al. 2010; Fattah et al. 2012, 2013); in a similar vein, it has attracted lots of attention in WiNoC. However, the methods proposed in conventional NoC do not cover the congestion of WRs. Therefore, in Rezaei et al. (2017) a congestion-aware platform based on OOK transmitters is introduced for WRs. A dynamic task mapping approach and the strategy of task migration are used in this paper to reduce the congestion of WRs. Task migration refers to the migration of a task being executed in a core to another core and executing the remaining part of the task in that core (Rezaei et al. 2015). The main purpose of task migration is to improve network performance.

In Wettin et al. (2014), authors used a token-based scheme to communicate the state of the input buffers of a WR to the wireline switches. Packets are transmitted through the wired links if the WR buffers are full. A blocking scheme

is proposed in Abadal et al. (2018) to avoid packets to enter a congested area. It uses a block signal that forces all the packets to go through the wired links when the WRs suffer from congestion. In DiTomaso et al. (2015), a reconfiguration algorithm was proposed to adapt to traffic patterns. This algorithm uses the hardware counters for collecting historical statistics from WRs. It gives more bandwidth to nodes with the most traffic. In Wang et al. (2012), the congestion information is globally and locally exchanged among subnets. Simulated annealing (SA) method is used in Wang et al. (2012) for the replacement of WRs. The main rationale behind SA has been adopted from annealing in solids (Hu et al. 2015). Sometimes, SA method selects bad solutions in order to avoid the local optimum problem. Furthermore, the authors of this method focus on parallel buffers to reduce the congestion of the paths leading to WRs. Finally, it can be concluded that increasing system resources significantly improves WiNoC performance in these papers.

The location and the number of optimal WRs are regarded as the important issues in WiNoC; therefore, the idea of WRs placement is used in other studies, such as Hu et al. (2015) and Bahrami et al. (2016). Researchers in Hu et al. (2015) used SA method to find the appropriate place for WRs in order to reduce latency in WiNoC. At the outset of executing the algorithm, WRs are located at the center of each subnet. In each step, a WR is selected and its location is exchanged with one of its neighboring routers. In case this replacement reduces the cost function, it will be accepted by the algorithm. Cost function refers to the average hop distance between all the source and destination pairs. However, if the replacement of WR increases the cost function, the probability of its acceptance will be so low. This bad solution is selected just for avoiding local optimal. In the next step, the Δ parameter is used to balance the transmitted packets to WRs. In this regard, only if the wired hop distance between the source and destination is greater than the total wireless hop distance and the Δ parameter, the hybrid wired/wireless routing will be applied. Otherwise, packet transmission will be carried out through conventional wired links. The Δ parameter depends on the network size and is used to balance the utilization of wired and wireless links. This parameter is experimentally obtained through numerous simulations on uniform random traffic pattern and 3-tuple traffic patterns. Therefore, a new value may be required to calculate for this parameter by changing traffic patterns, the bandwidth of wireless channels and network size.

The above-mentioned methods try to reduce the congestion of WRs and enhance the system performance by increasing hardware resources of the system, finding the location of WRs and task migration. Nevertheless, a dynamic routing algorithm is more important than task migration and application mapping for reducing the congestion of WRs (Rezaei et al. 2016). Thus, in this paper, a load-balanced congestion-aware routing algorithm

for WRs is proposed which can balance the utilization of wired and wireless links and reduce latency. The proposed algorithm does not use any application mapping, task migration or WR replacement algorithms. Therefore, the complexity of the proposed algorithm is lower than the other algorithms. Moreover, this method can be used for different traffic patterns and it can improve system performance in WiNoC without increasing hardware resources of the system such as a buffer size.

3 Load-balanced time-based congestion-aware routing algorithm (LTCA)

WiNoC consists of wired and wireless networks. A packet can be transmitted through wired links, wireless links or hybrid wired/wireless links in WiNoC (Hu et al. 2015). Figure 1 depicts a 10 × 10 WiNoC with 100 PEs. Each PE is connected to a conventional router. Since mesh topology is regarded as the most commonly used topology in NoC, in

this study, LTCA is evaluated in a two-dimensional mesh. As shown in Fig. 1, WiNoC includes 5 × 5-subnets and a WR is located at the center of each subnet. The routers and subnets are numbered in a row. Given the limitation of the number of wireless channels on a chip and for enhancing WiNoC scalability and reducing the occupied area by WRs, only one WR is used at the center of each subnet in this paper, and the WR located in each subnet is shared by all the PEs of that subnet.

In this algorithm, while making a decision about routing through wireless channels, different factors are taken into consideration so that resources can be used more effectively. Hop count in the hybrid wired/wireless routing should be shorter than the hop count in the conventional wired routing; however, this condition is not by itself sufficient for selecting a wireless channel in routing. Although the average hop distance is significantly reduced by transmitting more packets via wireless links, it may lead to the congestion of WRs and high latency. Moreover, the issue of competition in accessing wireless channel occurs, especially in LD traffic patterns. Several other factors such as network frequency, the internal latency of routers, the capacity of buffers, the width of wired channels and the dominant traffic type of the network (local or LD) can impact on the latency parameter in the hybrid wired/wireless routing. Taking these issues into consideration can result in a balanced utilization of wired and wireless links. At the outset, it is demonstrated that selecting the LD traffic which is true in Eq. (4) can lead to latency reduction. The size of each subnet is assumed to be $n \times n$. The symbols used in the equations are briefly described in Table 1.

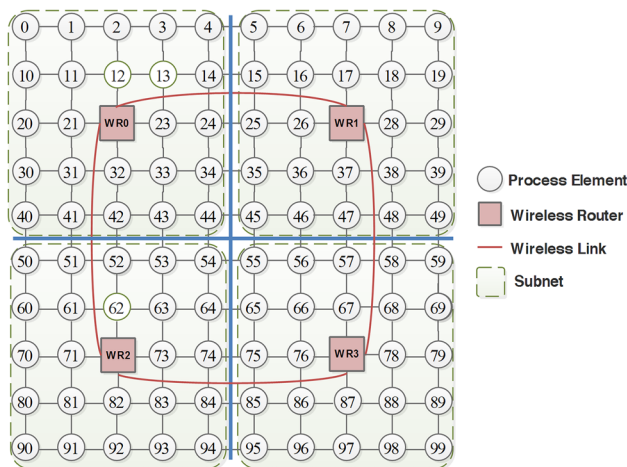


Fig. 1 A 10 × 10 WiNoC

$$DR_{wired} = width \times frequency \tag{1}$$

$$NumWL = HP_{wired}(wr_s \cdot wr_d) / n \tag{2}$$

$$HP_{wireless(s,d)} = HP_{wired}(s, wr_s) + NumWL + HP_{wired}(wr_d, d) \tag{3}$$

$$HP_{wired(s,d)} > HP_{wireless(s,d)} + (DR_{wired} / DR_{wireless})(NumWL) - NumWL. \tag{4}$$

Table 1 Symbols used in the equations of this study

Symbol	Description
S	Source node
D	Destination node
wr_s	WR located in source subnet
wr_d	WR located in destination subnet
HP_{wired}	Hop distance between the source and destination using conventional wired links
$HP_{wireless}$	Hop distance between the source and destination using hybrid wired/wireless links
NumWL	Number of wireless links on the path between source and destination
width	Width of the wired link
Frequency	Network frequency
DR_{wired}	Data rate of wired channel
$DR_{wireless}$	Data rate of the wireless channel

Equation (4) briefly indicates that a wired hop is not always equal to a wireless hop. Therefore, latency might be different in each of them. Equation (4) determines an equivalent of a wireless hop based on the wired hops. In this equation, wr_r refers to the WR which is located in the R th subnet. $NumWL$ stands for the number of required wireless links in the route between the source and destination. It is measured according to the wired hop count between two adjacent WRs (n) and the length of the wired route between wr_s and wr_d . Also, DR_{wired} refers to data transmission rate of wired links and $DR_{wireless}$ denotes data transmission rate of a wireless link. $DR_{wired}/DR_{wireless}$ indicates the ratio of the transmission speed of a wired channel to a wireless channel. Finally, since $NumWL$ has been once included in the $HP_{wireless}$ formula in Eq. (3), it is subtracted from the Eq. (4).

Figure 2 depicts the routing process between PE73 and PE87. This example indicates that using the condition $HP_{wired} > HP_{wireless}$ in the selection of a wireless channel is not by itself adequate to reduce latency. Network frequency is 1 GHz, the width of wired channels is 64 bits and data rate for each wireless channel is 20 Gb/s. Furthermore, it should be noted that the WR is shared by all the PEs in a subnet (Hu et al. 2015). The frequency division multiple access (FDMA) technique is adopted in each subnet for dividing the wireless channel. With regard to the existence of two wireless neighbors around the WR2, the transmission rate of each wireless link in this example is, on average, equal to 10 Gb/s. According to all the above-mentioned assumptions, Eq. (4) is obtained in the following way:

$$DR_{wired} = width \times frequency = 64 \text{ Gb/s}$$

$$HP_{wired(s,d)} = 5$$

$$HP_{wireless(s,d)} = 3$$

$$5 \not> 3 + \left(\frac{64}{10}\right) (1) - 1.$$

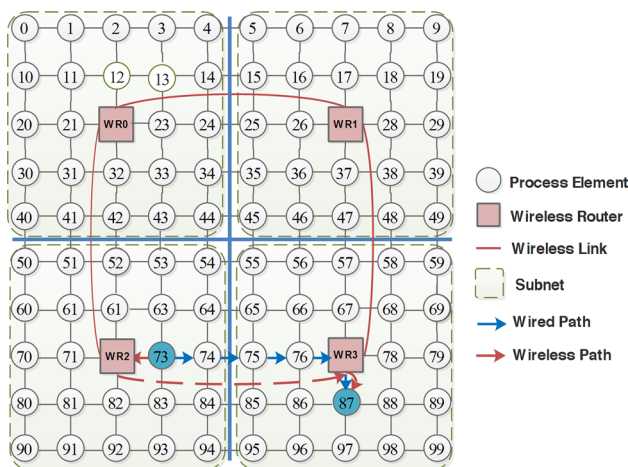


Fig. 2 Routing from source 73 to destination 87

In this example, $HP_{wired(s,d)} > HP_{wireless(s,d)}$; however, the hybrid wired/wireless routing for this source–destination pair might not improve latency parameter because it does not satisfy Eq. (4). It should be noted that the transmission speed of a wired hop in this example is more than the transmission speed of a wireless hop. This instance shows that one wired hop is not always equal to one wireless hop. Furthermore, the two factors of the high frequency of wired links and the division of wireless channel result in higher data rate for wired links than the wireless links during routing. With regard to the above-mentioned arguments, Eq. (4) is considered to transmit traffic through wireless channels in this paper. It takes advantage of the high data rate of the wired links in shorter distances and employs wireless links for longer distances.

In the proposed method, at first, the number of packets which are true in condition 4 is determined in each subnet. Then, the permission for using the wireless channels is given to them. The routing decision is made at the source and the permission will not be refunded until the packet reaches the destination. Next, the required time for transmitting these packets through the wireless channels is calculated according to Eq. (5).

$$\Delta t = \frac{numPacket \times Packet_{size}}{WL_{capacity}} \times \frac{1}{L_f}. \tag{5}$$

In this equation, $WL_{capacity}$ denotes the wireless channel capacity in each subnet; $Packet_{size}$ indicates the size of a packet; numPacket refers to the number of authorized packets in a subnet and Δt refers to the required time for transferring of authorized packets from the wireless channel and is measured according to the bandwidth of wireless links and the packet size. The number of authorized packets in each subnet is variable. Therefore, Δt has different values in different subnets. In this equation, load factor, ($0 < L_f \leq 1$), is defined to balance the utilization of wired and wireless links in WiNoC. It depends on the network load and increases by increasing the network load. The lower the network load, the smaller the value of L_f ; consequently, the larger the Δt value. Therefore, the utilization of wireless links is low when the network load is low. By increasing the network load, L_f becomes larger; consequently, Δt becomes smaller and the utilization of wireless links increases gradually. The maximum value of L_f is related to saturation load, where the wired links are saturated completely and maximum utilization of wireless links can improve network performance. As a result, the utilization of wired and wireless links is more balanced by using the L_f parameter.

In the proposed LTCA, permission to a packet is given at the beginning of the routing process at the source. After the packet has received the required permission to transmit through the wireless links, its related core transmits it towards the WR of its subnet. Then, this core waits for

Δt seconds. In other words, the time interval between the transmissions of two authorized packets from a core is equal to Δt seconds. At this time, if a new packet is injected into the network from that core, the core will transmit it through the wired channels. As a result, the transmission of packets will never stop. Algorithm (1) shows the pseudo-code related to the proposed LTCA.

In this algorithm, Δt refers to the time interval between two consecutive transmissions of the authorized packets from one core to wireless channels. Δt is calculated independently for each subnet. *Current-Time* refers to the current time of the system and *NT* denotes the next permitted time for accessing to wireless channels. In case a packet has been recently injected into the network and has not yet received any permissions, it will receive permission 1 for using wireless channel if condition 4 is true about that packet. In this algorithm, access to the wireless channel is indicated by 1 and the lack of access to the wireless channel is indicated by 0. The core whose packet has received permission 1 waits for the Δt second; after this time, it gives the new permission for its other packets to go through the wireless channel. Hence, if a new packet is injected by this core throughout this time interval, it will be routed in a wired way even if condition 4 is true about it.

The packets injected by different cores can use wireless channels if condition 4 is true about them. In fact, Δt

Table 2 Simulation parameters

Parameters	Setting
Frequency	1 GHz
Number of cores on chip	100
Switching technique	Wormhole
Routing	XY
Number of virtual channels (VCs)	2
Number of wireless routers	4
Flit width	64 bits
Packet size	256 bits
Number of buffers available per VC	1

indicates the required time for passing all the authorized packets of a subnet through the wireless channel related to that subnet. Also, since the number of authorized packets may be different in various subnets, hence, the Δt value in the various subnets will be different. Furthermore, dimension-order (XY) routing algorithm has been adopted in each of the wired and wireless networks. In this method, a packet is routed first in the x-dimension, then in the y-dimension to reach the destination (Dally and Towles 2004). Since the XY routing algorithm is deadlock-free, routing in each wired and wireless network in the proposed LTCA will be deadlock-free. This issue has been proven and demonstrated more exhaustively in Hu et al. (2015).

Algorithm 1: LTCA routing algorithm

Input: traffic information, $numPacket$, L_f

Output: routing path

NT: next time for giving permission

currentTime: current time of the system

```

1: if (a packet hasn't received a permission yet) then
2:   if (the condition 4 is true) && (currentTime > NT) then
3:     permission(s, d) = 1
4:      $\Delta t = \frac{numPacket * Packet_{size}}{WL_{capacity}} * \frac{1}{L_f}$ 
5:     NT = currentTime +  $\Delta t$ 
6:   else
7:     permission(s, d) = 0
8:   end if
9: end if
10: if ((s & d are in the same subnet) or (permission(s, d) == 0)) then
11:   Route packet using only wired links
12: else
13:   Route packet using wired/wireless links
14: end if

```

4 Performance evaluation

The proposed LTCA is compared with baseline NoC and with the algorithm proposed in Hu et al. (2015) in a 10×10 mesh-based WiNoC. The simulations of the proposed LTCA and the other algorithms are carried out by using HNOCS (Ben-Itzhak et al. 2012) package in the omnet++ simulator. HNOCS is a NoC simulator which supports heterogeneity feature with the different link capacities. In this paper, HNOCS simulator is developed for implementing WRs and wireless channels. Table 2 gives a summary of simulation settings.

WiNoC is divided into 5×5 subnets, and a WR is located at the center of each subnet. The width of the wired links is 64 bits and is equal to the size of a flit. The buffer size of the WRs is equal to the buffer size of conventional routers. Both synthetic and application traffic patterns are taken into consideration in evaluating the performance of the proposed method. The synthetic traffic pattern includes uniform random traffic pattern and application traffic includes 3-tuple traffic patterns.

The spatial and temporal features of real-world applications in Soteriou et al. (2006) are considered with respect to three parameters: (1) burstiness, (2) injection distribution, and (3) hop distance (Hu et al. 2015). The burstiness indicates how often and how much bursts are injected into the network; it is modeled by the Hurst parameter, $0.5 < H \leq 1.0$ (Hu et al. 2015). The closer H value to 1.0, the higher the level of burstiness. In this paper, two levels of burstiness are modeled: moderate burstiness ($H=0.65$) and high burstiness ($H=0.9$). Injection distribution refers to the way packets have been distributed among the processing nodes. Two modes, namely hot-spot and evened-out are considered here. In the hot-spot mode, 10% of nodes receive 68% of the entire network traffic. While, in the evened-out mode, 20% of nodes receive the same amount of traffic. The third parameter is the hop distance which indicates the hop distance from the source to the destination. Accordingly, two traffic types, local traffic and LD traffic are defined as

Table 3 3-tuple traffic categories (Hu et al. 2015)

	Burstiness	Injection	Hop distance
3tc0	Moderate	Hot-spot	Local
3tc1	High	Hot-spot	Local
3tc2	Moderate	Eveded-out	Local
3tc3	High	Eveded-out	Local
3tc4	Moderate	Hot-spot	Long distance
3tc5	High	Hot-spot	Long distance
3tc6	Moderate	Eveded-out	Long distance
3tc7	High	Eveded-out	Long distance

follows: in the local traffic, 20% of the entire traffic will have hops greater than (4) However, in LD traffic type, 80% of the entire traffic will go through a route which is greater than 8 hops. The permutation of these three parameters is given in Table 3.

In this section, 8 different 3-tuple traffic patterns, as well as random uniform traffic pattern, are evaluated in a 10×10 WiNoC. In this study, latency is defined as end-to-end latency and throughput is defined based on the number of delivered packets to the destination at the time unit. Furthermore, the utilization of the wired/wireless links and packet loss probability are used as other criteria for evaluating WiNoC performance.

Another significant issue in this paper is related to detecting and finding the balance point of the wired and wireless link utilization in proposed LTCA. It should be determined

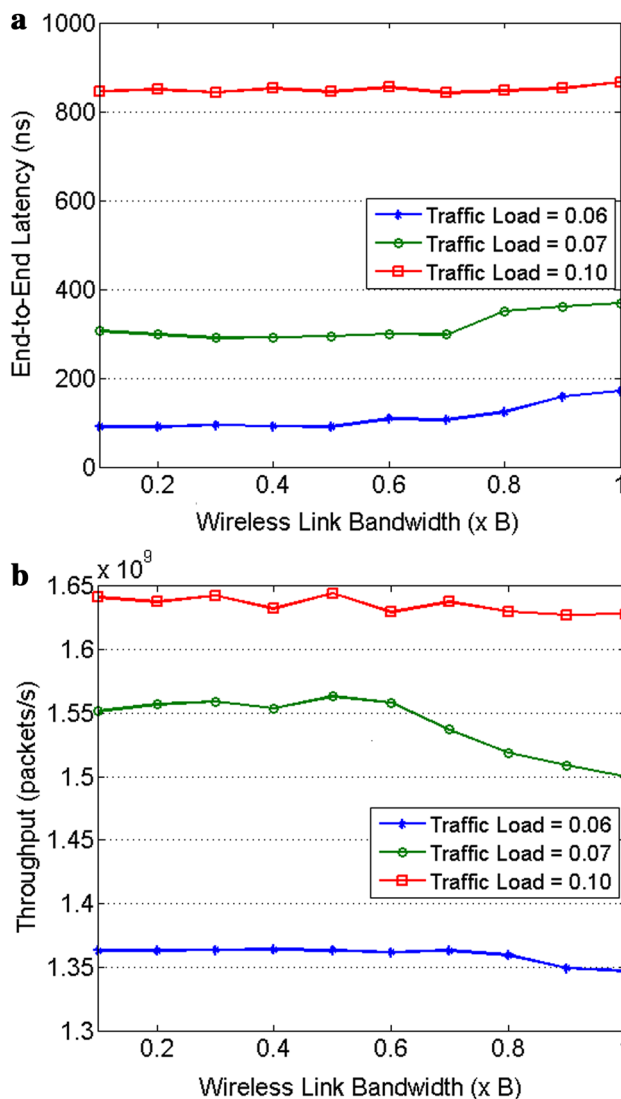


Fig. 3 Variations of **a** latency and **b** throughput in uniform random traffic pattern

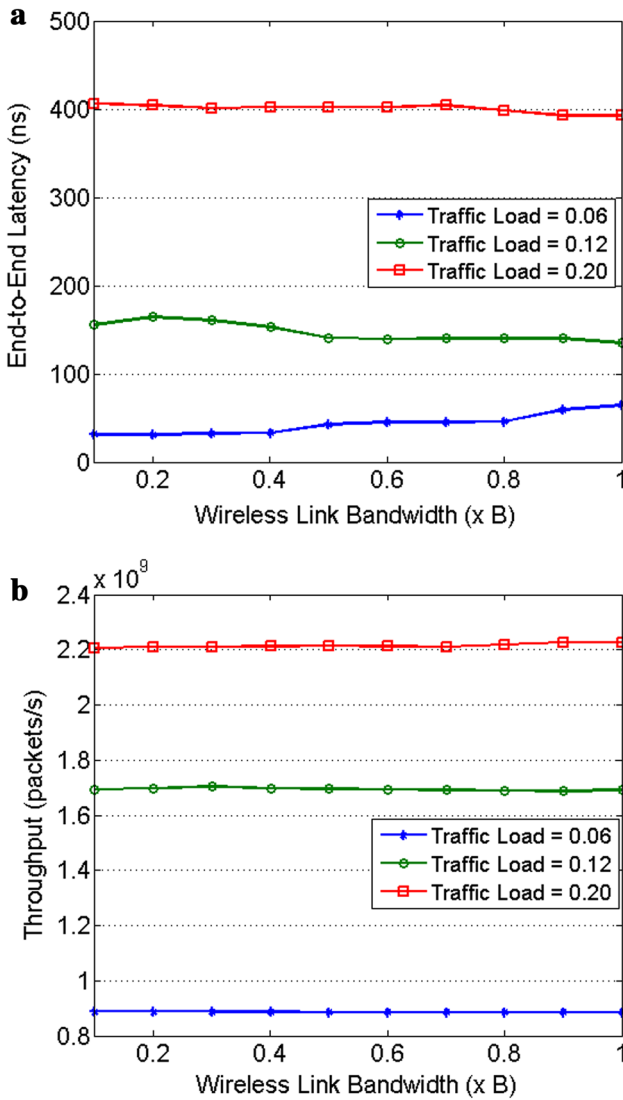


Fig. 4 Variations of a latency and b throughput in 3tc2 traffic pattern

how much load for wired and wireless links will result in minimum latency and maximum throughput in WiNoC. In this paper, that amount of load is referred to as the balance point of the wired/wireless channels. For finding this point, WiNoC is analyzed in three different situations: low load, moderate load and high load. Figures 3, 4 and 5 depict average latency and throughput variations by putting the load at different capacities of wireless links with regard to instances of various traffic patterns. Each traffic pattern is evaluated under low, moderate and high load. The horizontal axis represents the wireless channel bandwidth used by LTCA. Parameter B refers to the total bandwidth of the wireless channel in each subnet. The results are classified into three groups as follows:

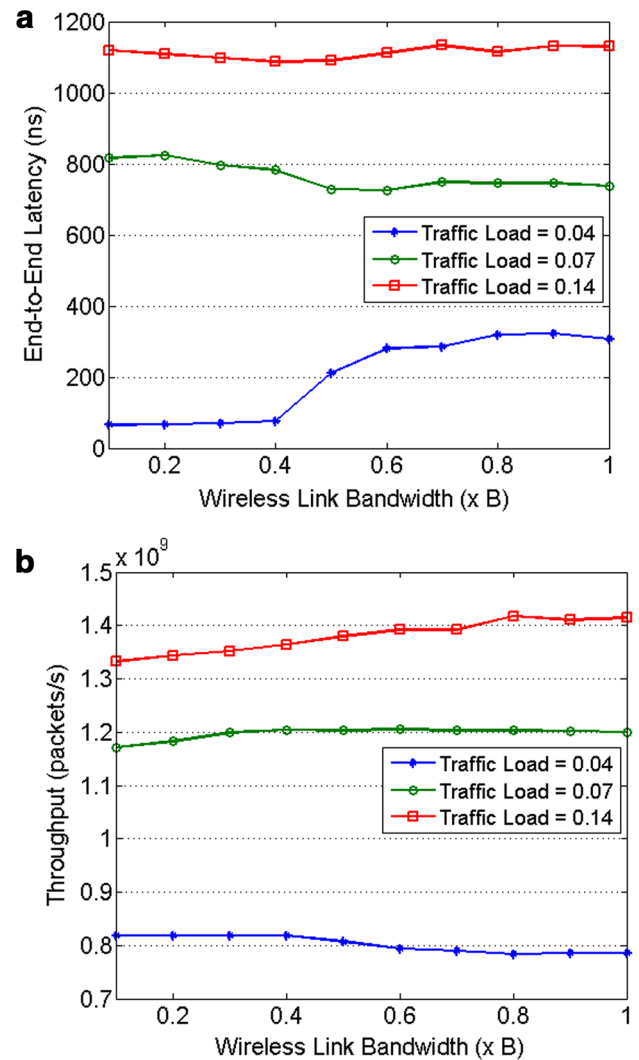


Fig. 5 Variations of a latency and b throughput in 3tc7 traffic pattern

- When the network load is low, the wired links are quiet. Therefore, the more utilization of wireless links (compared to wired links) leads to unbalanced utilization of wired/wireless links and high latency. Hence, latency reduction is achieved by less utilization of wireless links in low load situation.
- When the network load is moderate, the best performance with regard to latency and throughput is achieved when 50% of wireless link capacity is used. Simulation results indicated that the greatest amount of reduction regarding latency parameter in the proposed LTCA is achieved when the traffic load is put at approximately 50% of the capacity of wireless channels. In other words, when about 50% load is injected on wireless links, the utilization of the wired and wireless channels will be more balanced.
- When the network load is high, the wired links are saturated. In this situation, the more utilization of wireless

links can improve throughput in WiNoC (especially in LD traffic patterns).

To fit Δt with network load, L_f parameter is used in this paper ($0 < L_f \leq 1$). This parameter increases gradually by increasing network load. When the network load is low, the L_f value is small. By increasing traffic load, L_f increases gradually. The maximum amount of L_f is equal to 1 when the network is saturated. At this point, the maximum utilization of wireless channels is taken in order to achieve high throughput. Therefore, L_f adjusts Δt according to network load; consequently, it balances the utilization of wired and wireless links in different traffic loads. In this study, L_f is calculated simply by considering saturation point, $SatPoint$, in conventional NoC and current traffic load, $TrLoad$, (Eq. 6). For example, if $Satpoint = 0.2$ and $TrLoad = 0.1$, L_f is equal to 0.5 (half of wireless link capacity). If $Satpoint = 0.2$ and $TrLoad = 0.2$, L_f is equal to 1 (maximum use of wireless link capacity in LTCA).

$$L_f = TrLoad / Satpoint. \tag{6}$$

Figure 6 is related to the saturation throughput of LTCA and other algorithms. The improvement regarding throughput in uniform random traffic pattern is negligible, i.e. about 2%. Figure 7 illustrates end-to-end latency in the uniform random traffic pattern. The maximum improvement of latency in the uniform random traffic pattern in the LTCA algorithm is equal to 12.8%. While injecting a packet into the network according to the uniform random traffic pattern, the destination of the packet is selected randomly. Also, the packets injected from one core at different cycles select different destinations. Regarding such patterns in which the destinations of packets are not pre-determined, Δt value will be measured approximately. For example, assuming that the size of each subnet is $n \times n$, n^2 packets will be produced in

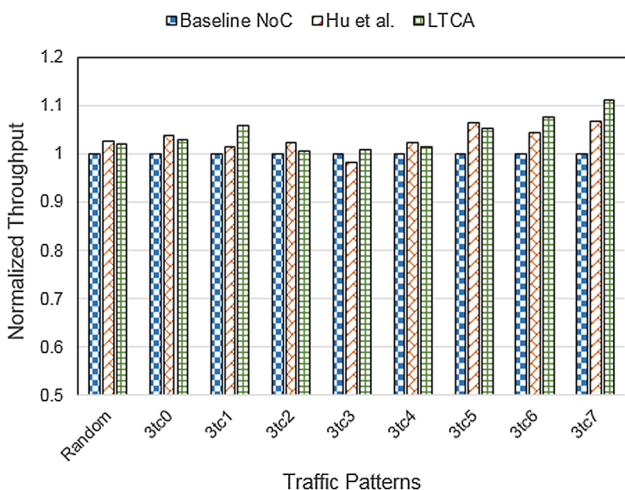


Fig. 6 Comparison of throughput in LTCA and other methods

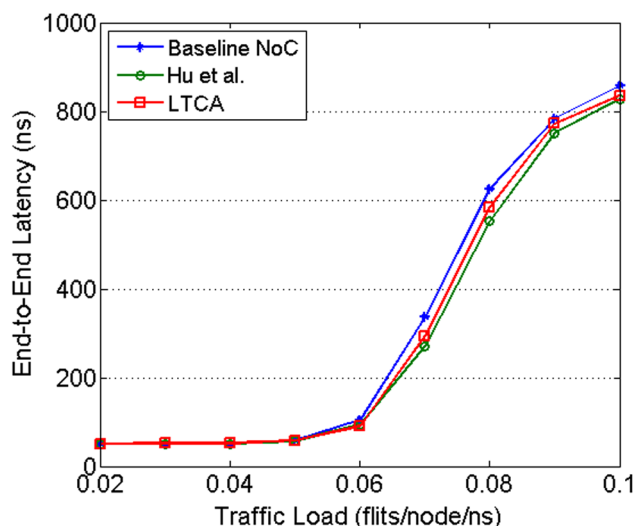


Fig. 7 Comparison of latency in uniform random traffic pattern

each subnet during a cycle. Also, assuming that packets have been uniformly distributed, $\frac{3}{4}$ of the injected packets in each subnet will be transmitted to the adjacent subnets. Thus, the Δt indicates the required time for transmitting $\frac{3n^2}{4}$ packets through a wireless channel in a subnet which is measured via Eq. (7):

$$\Delta t = \frac{3n^2}{4} \times \frac{Packet_{size}}{WL_{capacity}} \times \frac{1}{L_f}. \tag{7}$$

Figures 8, 9, 10 and 11 indicate end-to-end latency in the local traffic patterns. The maximum improvement in latency in LTCA algorithm in local traffic patterns is related to 3tc3 traffic pattern with 62% in comparison to baseline NoC. The

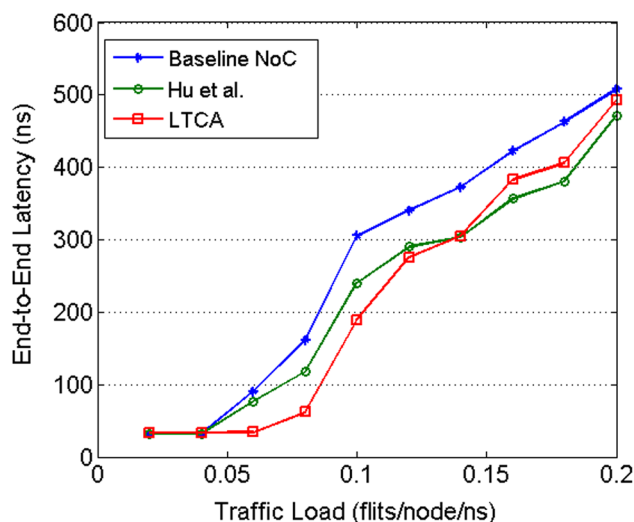


Fig. 8 Comparison of latency in 3tc0 traffic pattern

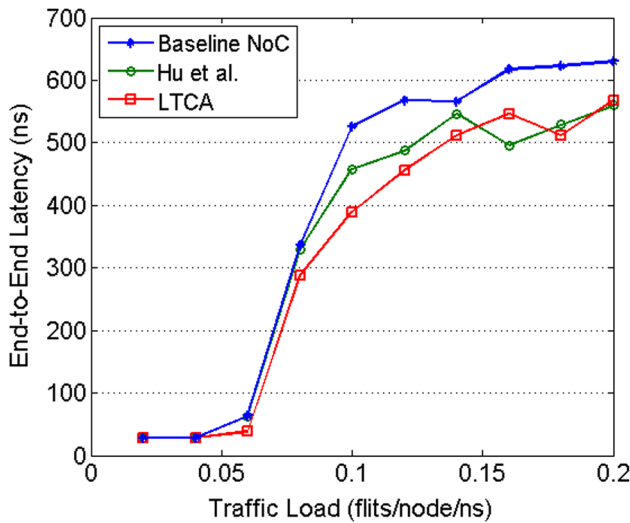


Fig. 9 Comparison of latency in 3tc1 traffic pattern

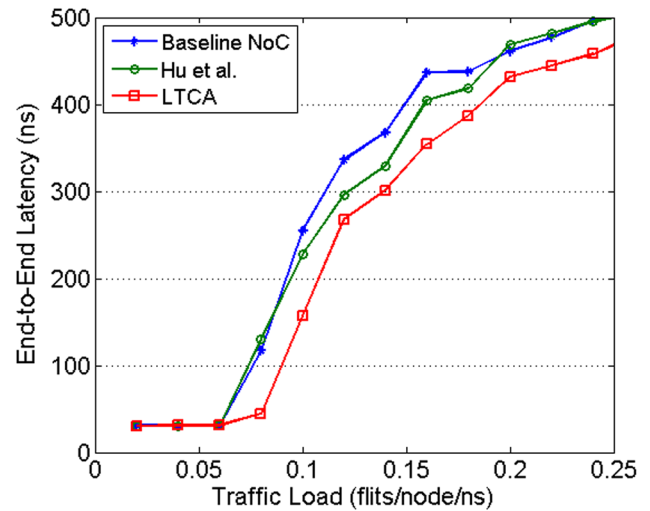


Fig. 11 Comparison of latency in 3tc3 traffic pattern

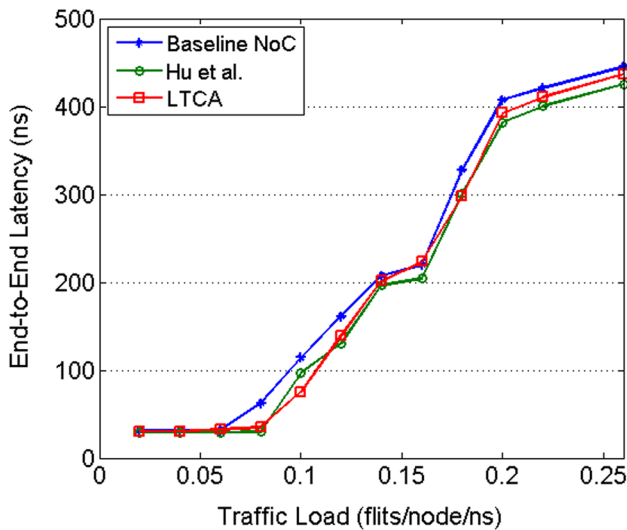


Fig. 10 Comparison of latency in 3tc2 traffic pattern

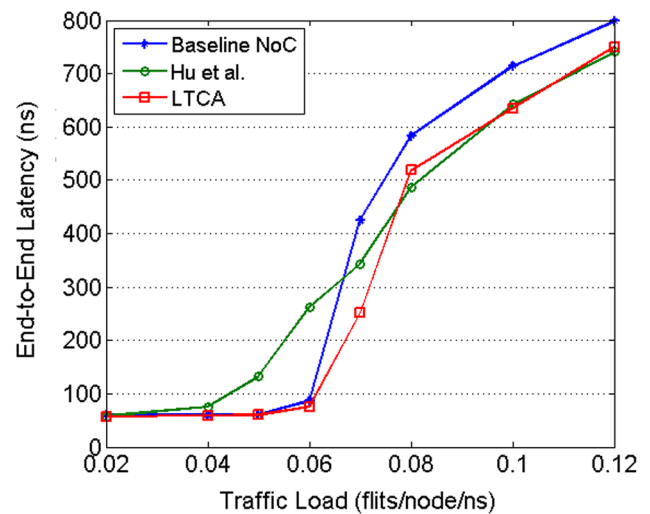


Fig. 12 Comparison of latency in 3tc4 traffic pattern

maximum throughput enhancement in the local traffic patterns is achieved up to 5.8% at saturation load and is related to 3tc1 traffic pattern. In the majority of local traffic patterns, the proposed LTCA demonstrated more improvement than the method proposed in Hu et al. (2015) with regard to latency parameter. Nevertheless, in some cases, method (Hu et al. 2015) demonstrates a better performance than LTCA close the saturation. Such a result indicates that, in local traffic patterns, the idea of WRs replacement can significantly enhance WiNoC performance.

Although Δ in method (Hu et al. 2015) is experimentally measured through several simulations, its value seems to be appropriate in the majority of traffic load in local

traffic patterns. However, this is not true in LD traffic patterns (3tc4–3tc7). As shown in Figs. 12, 13, 14 and 15, the reduction of latency in method (Hu et al. 2015) is mainly related to the close to saturation points and beyond them. Even in some cases, average latency in pre-saturation points is worse than that of baseline NoC. In the majority of LD traffic patterns, the proposed LTCA demonstrates notably more improvement than its wired counterparts and method (Hu et al. 2015) in terms of latency and throughput. LTCA has more adaptability than the algorithm proposed in Hu et al. (2015) since the Δt value is different in different traffic patterns and is measured according to the specific features of each traffic pattern. Moreover, LTCA considers the network load to determine the accurate value of Δt . In contrast, the Δ parameter which is defined in method (Hu et al. 2015) for

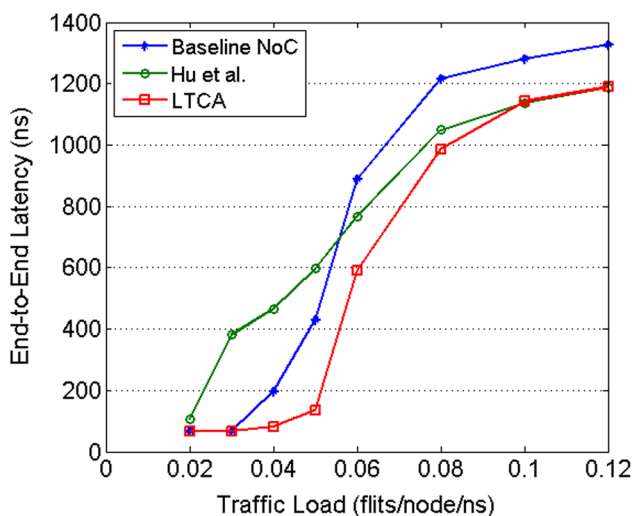


Fig. 13 Comparison of latency in 3tc5 traffic pattern

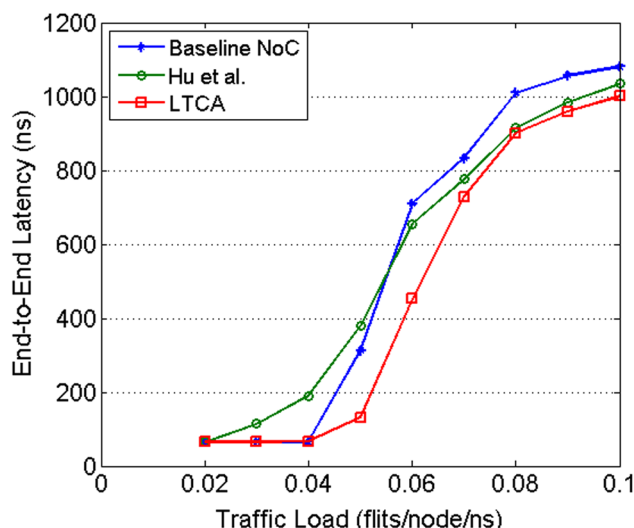


Fig. 15 Comparison of latency in 3tc7 traffic pattern

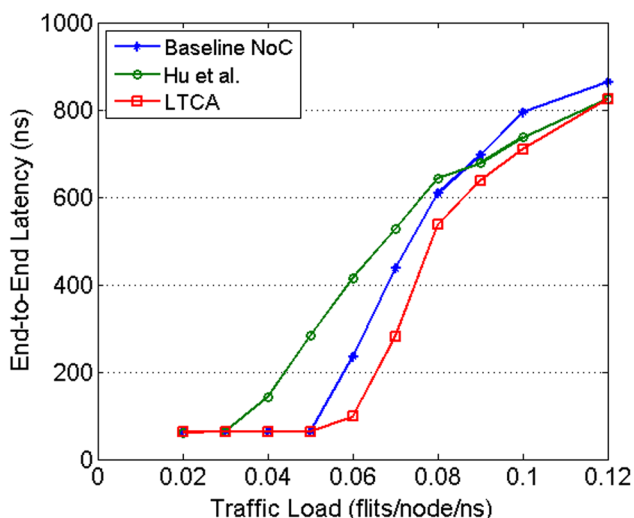


Fig. 14 Comparison of latency in 3tc6 traffic pattern

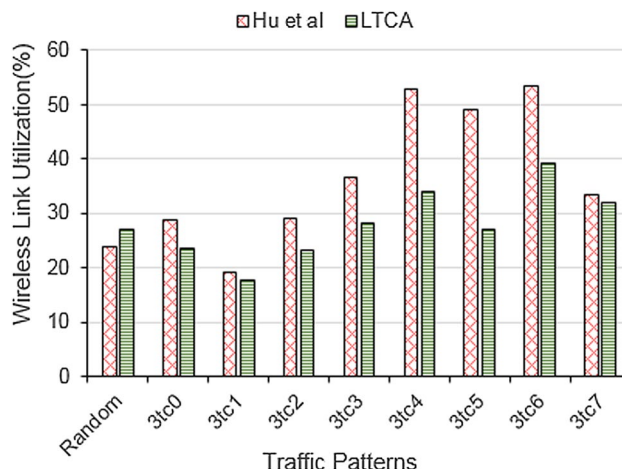


Fig. 16 Wireless link utilization

preventing congestion of WRs, is constant in all the local and LD traffic patterns. In general, it can be argued that the idea of WRs replacement in local traffic patterns is highly suitable. However, in LD traffic patterns, preventing the congestion of WRs is the more significant issue.

In LD traffic patterns, maximum latency reduction in comparison with the baseline NoC is 68% which occurs in the 3tc5 traffic pattern. Maximum throughput enhancement is about 11% in comparison with conventional NoC which is related to the 3tc7 traffic pattern. The improvement of the proposed method is significant in LD traffic patterns where the use of wireless links is more important.

Figure 16 illustrates the comparison of the proposed LTCA and algorithm (Hu et al. 2015) with regard to the utilization of wireless links. Although wireless link utilization in

all the traffic patterns in method (Hu et al. 2015) is more than that of the proposed LTCA, as shown in Fig. 17, the overall link utilization, i.e. both wired and wireless links, is better improved in LTCA than the method (Hu et al. 2015). Furthermore, excessive use of wireless links in comparison with wired links in method (Hu et al. 2015) leads to high packet loss probability. Figure 18.a depicts packet loss probability at the saturation point. Figure 18.b indicates the packet loss probability at 2/3 saturation point. Both of the Fig. 18.a and 18.b indicate that the proposed LTCA routing algorithm has a better improvement than baseline NoC and method (Hu et al. 2015) with respect to packet loss probability.

Figures 19 and 20 depict other simulations that are carried out on samples of local and LD patterns in order to further investigation and comparison of different algorithms

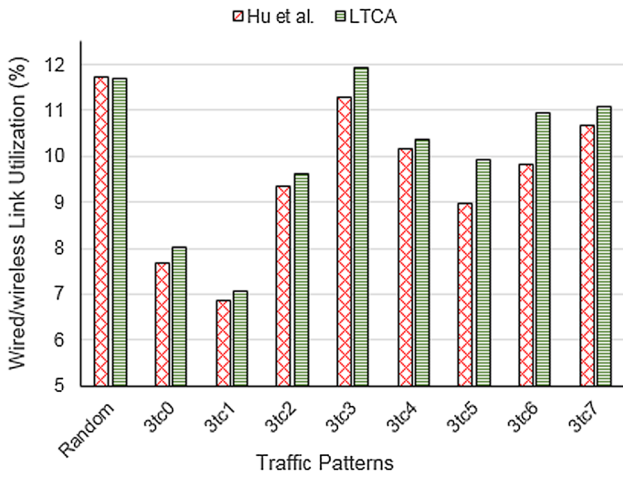


Fig. 17 Wired/wireless link utilization

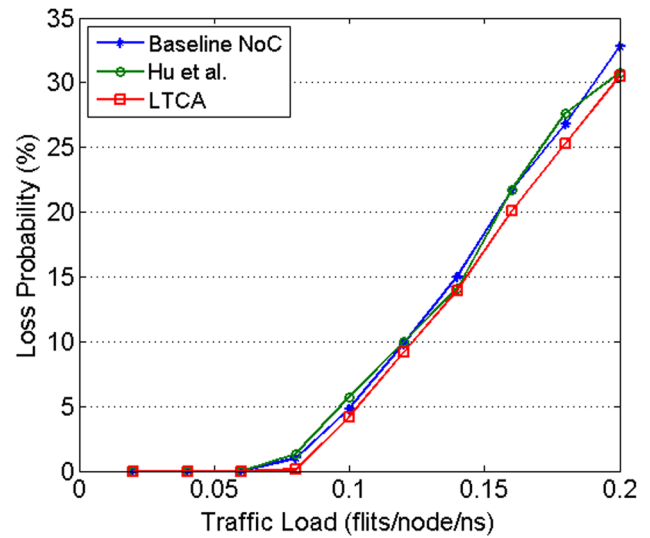


Fig. 19 Comparison of packet loss probability in 3tc3 traffic pattern

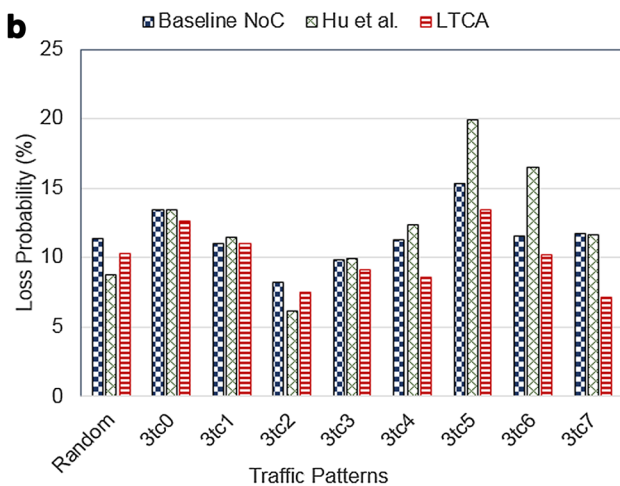
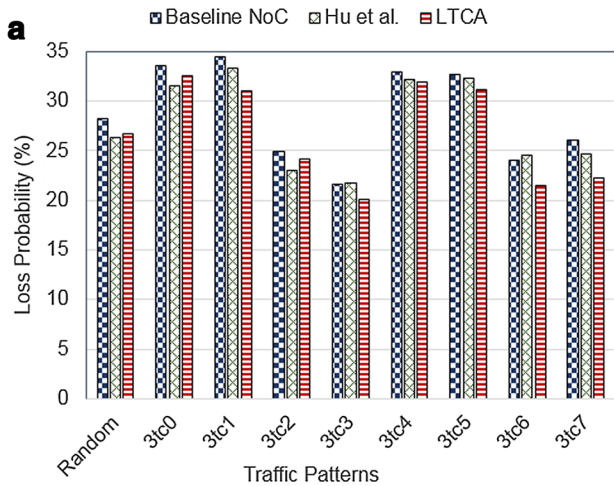


Fig. 18 Packet loss probability at **a** saturation load **b** 2/3 saturation load

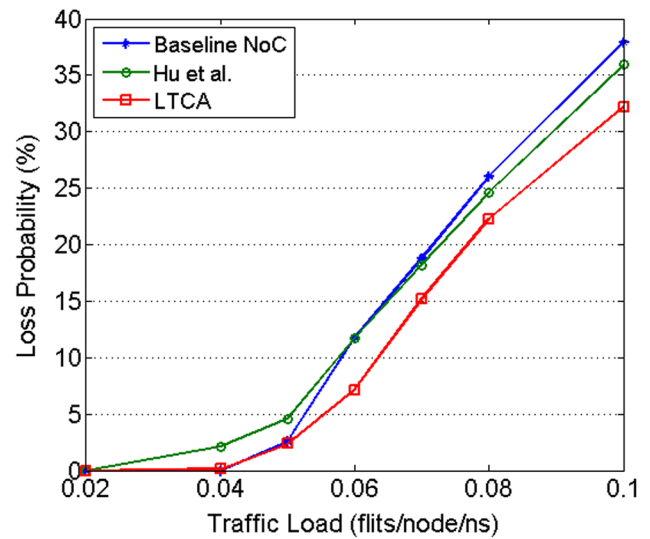


Fig. 20 Comparison of packet loss probability in 3tc7 traffic pattern

with regard to packet loss probability. 3tc3 and 3tc7 traffic patterns are randomly selected from among local and LD traffic patterns for further investigation. Based on the overall obtained results, it seems that Δ parameter in method (Hu et al. 2015) is unable to sort out the congestion problem in WRs, especially in LD traffic patterns and pre-saturation load. Consequently, it leads to the loss of packets. As a result, although wireless channels are regarded as valuable resources in WiNoC and their maximum utilization is significant, balanced use of wired and wireless links is regarded as a more critical issue which should be taken into consideration for reducing latency.

5 Conclusion

WiNoC consists of wired and wireless channels. It takes advantage of the high data rate of wired links for shorter distances while using wireless links for longer distances. Long multi-hops wired links are replaced with a single wireless hop in WiNoC. However, WiNoC has only a limited number of wireless channels which are shared by all the cores. When a lot of LD traffic wants to go through wireless channels, congestion occurs at WRs which leads to system inefficiency and high latency.

In this paper, a load-balanced congestion-aware routing algorithm was proposed which balances the network load between wired and wireless networks and reduces the congestion of WRs by controlling access to the wireless channels. The proposed algorithm considers some criteria such as network frequency, the width of wired channels and the transmission rate of wireless channels for selecting authorized packets during the hybrid wired/wireless routing. Then, it calculates the required time for transmitting the selected packets through wireless channels with regard to traffic load; meanwhile, it adopts the conventional wired routing for transmitting other packets. When the determined time span is finished, the licensing process for the new packets resumes again. Regardless of WR replacement techniques and without using extra hardware, the proposed method is able to reduce the congestion in WRs. It also distributes the traffic load between the wired and wireless network in a balanced way to achieve low latency by applying load factor in hybrid wired/wireless routing. The results of evaluating the proposed LTCA algorithm indicated it was able to significantly improve system efficiency parameters such as latency, throughput, wired/wireless link utilization and the packet loss probability.

As for directions for further research, congestion information can be exchanged among different subnets by means of unique features of wireless interconnects. Also, the appropriate size of the time intervals should be investigated and determined. Furthermore, methods should be applied for selecting the proper number of packets according to the respective interval for reducing latency and balancing the utilization of wired/wireless links.

References

- Abadal S, Torrellas J, Alarcón E, Cabellos-Aparicio A (2018) OrthoNoC: a broadcast-oriented dual-plane wireless network-on-chip architecture. *IEEE Trans Parallel Distrib Syst* 29:628–641. <https://doi.org/10.1109/TPDS.2017.2764901>
- Bahrami B, Jamali MAJ, Saeidi S (2016) Proposing an optimal structure for the architecture of wireless networks on chip. *Telecommun Syst* 62:199–214. <https://doi.org/10.1007/s11235-015-0075-9>
- Bansal A, Gupta A, Sharma DK, Gambhir V (2018) IICAR-inheritance inspired context aware routing protocol for opportunistic networks. *J Ambient Intell Humaniz Comput*. <https://doi.org/10.1007/s12652-018-0815-2>
- Ben-Itzhak Y, Zahavi E, Cidon I, Kolodny A (2012) HNOCS: modular open-source simulator for heterogeneous NoCs. In: Proceedings of 2012 international conference on embedded computer systems architecture, modelling and simulations IC-SAMOS. pp 51–57. <https://doi.org/10.1109/SAMOS.2012.6404157>
- Carvalho ELdS, Calazans NLV, Moraes FG (2010) Dynamic task mapping for MPSoCs. *IEEE Des Test Comput* 27:26–35
- Chen G, Anders MA, Kaul H et al (2015) A 340 mV-to-0.9 V 20.2 Tb/s source-synchronous hybrid packet/circuit-switched 16 × 16 network-on-chip in 22 nm tri-gate CMOS. *IEEE J Solid State Circuits* 50:59–67
- Chou C-L, Marculescu R (2008) Contention-aware application mapping for network-on-chip communication architectures. In: ICCD 2008, IEEE international conference on computer design, pp 164–169
- Dally WJ, Towles BP (2004) Principles and practices of interconnection networks. Morgan Kaufmann Publishers Inc., San Francisco
- Davis WR, Wilson J, Mick S et al (2005) Demystifying 3D ICs: the pros and cons of going vertical. *IEEE Des Test Comput* 22:498–510. <https://doi.org/10.1109/MDT.2005.136>
- Deb S, Chang K, Yu X et al (2013) Design of an energy-efficient CMOS-compatible NoC architecture with millimeter-wave wireless interconnects. *IEEE Trans Comput* 62:2382–2396
- DiTomaso D, Kodi A, Kaya S, Matolak D (2011) IWISSE: Inter-router wireless scalable express channels for network-on-chips (NoCs) architecture. In: Proceeding of symposium on high perform interconnects hot interconnects, pp 11–18. <https://doi.org/10.1109/HOTI.2011.12>
- DiTomaso D, Kodi A, Matolak D et al (2015) A-WiNoC: adaptive wireless network-on-chip architecture for chip multiprocessors. *IEEE Trans Parallel Distrib Syst* 26:3289–3302. <https://doi.org/10.1109/TPDS.2014.2383384>
- Duraisamy K, Xue Y, Bogdan P, Pande PP (2017) Multicast-aware high-performance wireless network-on-chip architectures. *IEEE Trans Very Large Scale Integr Syst* 25:1126–1139
- Fattah M, Ramirez M, Daneshdalan M et al (2012) CoNA: dynamic application mapping for congestion reduction in many-core systems. In: Computer design (ICCD), 2012 IEEE 30th international conference, pp 364–370
- Fattah M, Daneshdalan M, Liljeberg P, Plosila J (2013) Smart hill climbing for agile dynamic mapping in many-core systems. In: Proceedings of the 50th annual design automation conference, ACM, New York, NY, USA, p 39:1–39:6
- Howard J, Dighe S, Vangal SR et al (2011) A 48-core IA-32 processor in 45 nm CMOS using on-die message-passing and DVFS for performance and power scaling. *IEEE J Solid State Circuits* 46:173–183
- Hu W-H, Wang C, Bagherzadeh N (2015) Design and analysis of a mesh-based wireless network-on-chip. *J Supercomput* 71:2830–2846. <https://doi.org/10.1007/s11227-014-1341-4>
- Kempa K, Rybczynski J, Huang Z et al (2007) Carbon nanotubes as optical antennae. *Adv Mater* 19:421–426. <https://doi.org/10.1002/adma.200601187>
- Lee S-B, Zhang L, Cong J et al (2009) A scalable micro wireless interconnect structure for CMPs. In: Proceedings of 15th annual international conference on mobile computing and networking—MobiCom'09 217. <https://doi.org/10.1145/1614320.1614345>
- Lin JJ, Wu HT, Su Y et al (2007) Communication using antennas fabricated in silicon integrated circuits. *IEEE J Solid State Circuits* 42:1678–1687

- Matolak DW, Kodi A, Kaya S et al (2012) Wireless networks-on-chips: architecture, wireless channel, and devices. *IEEE Wirel Commun* 19:58–65. <https://doi.org/10.1109/MWC.2012.6339473>
- Murray J, Wettin P, Pande PP, Shirazi B (2016a) Sustainable wireless network-on-chip architectures. Morgan Kauffmann, Cambridge, pp 1–9. <https://doi.org/10.1016/B978-0-12-803625-9.00008-X>
- Murray J, Wettin P, Pande PP, Shirazi B (2016b) Wireless small-world NoCs. In: wireless network-on-chip architectures, pp 37–45. <https://doi.org/10.1016/B978-0-12-803625-9.00011-X>
- Qiuli C, Wei X, Fei D, Ming H (2018) A reliable routing protocol against hotspots and burst for UASN-based fog systems. *J Ambient Intell Humaniz Comput*. <https://doi.org/10.1007/s12652-018-0810-7>
- Rayess W, Matolak DW, Kaya S, Kodi AK (2017) Antennas and channel characteristics for wireless networks on chips. *Wirel Pers Commun*. <https://doi.org/10.1007/s11277-017-4144-0>
- Rezaei A, Daneshtalab M, Zhao D et al (2015) Dynamic application mapping algorithm for wireless network-on-chip. In: Proceeding of 23rd Euromicro international conference on parallel, distributed and network-based processing (PDP), 2015, pp 421–424. <https://doi.org/10.1109/PDP.2015.14>
- Rezaei A, Daneshtalab M, Palesi M, Zhao D (2016) Efficient congestion-aware scheme for wireless on-chip networks. In: 2016 24th Euromicro international conference on arallel, distributed, and network-based processing, pp 742–749. <https://doi.org/10.1109/PDP.2016.88>
- Rezaei A, Daneshtalab M, Zhao D (2017) CAP-W: congestion-aware platform for wireless-based network-on-chip in many-core era. *Microprocess Microsyst* 52:23–33. <https://doi.org/10.1016/j.micpro.2017.05.014>
- Soteriou V, Eisley N, Wang H et al (2006) Polaris: a system-level roadmap for on-chip interconnection networks. In: IEEE international conference on computer design, ICCD, 2006, pp 134–141. <https://doi.org/10.1109/ICCD.2006.4380806>
- Vantrease D, Schreiber R, Monchiero M et al (2008) Corona: system implications of emerging nanophotonic technology. In: Proceedings of international symposium on computer architecture, pp 153–164. <https://doi.org/10.1109/ISCA.2008.35>
- Wang C, Hu WH, Bagherzadeh N (2012) A load-balanced congestion-aware wireless network-on-chip design for multi-core platforms. *Microprocess Microsyst* 36:555–570. <https://doi.org/10.1016/j.micpro.2011.10.002>
- Wettin P, Kim R, Murray J et al (2014) Design space exploration for wireless NoCs incorporating irregular network routing. *IEEE Trans Comput Des Integr Circuits Syst* 33:1732–1745. <https://doi.org/10.1109/TCAD.2014.2351577>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.