Hierarchical approach for hybrid wireless Network-on-chip in many-core era ☆

Amin Rezaei [a],*, Masoud Daneshtalab [b], Farshad Safaei [c], Danella Zhao [a]

[a] Department of Computer Science, University of Louisiana at Lafayette (ULL), Lafayette, USA
[b] Department of Electronic Systems, Royal Institute of Technology (KTH), Stockholm, Sweden
[c] Department of Computer Science and Engineering, Shahid Beheshti University (SBU), Tehran, Iran

ARTICLE INFO

ABSTRACT

Due to high latency and high power consumption in long hops between operational cores of Network-on-Chips (NoCs), the performance of such architectures has been limited. Billions of transistors available on a single chip present opportunities for new levels of computing capability. In order to fill the gap between computing requirements and efficient communications, a new technology called Wireless NoC has been emerged. Employing wireless communication links between cores, wireless NoC has reasonably increased the performance of NoC. However, wireless transceivers along with associated antenna impose considerable area and power overheads in wireless NoCs. Thus, in this paper, we introduce a hybrid wireless NoC called Hierarchical Wireless-based Architecture (HiWA) to use the wireless resources optimally. In the proposed approach the network is divided into subnets where intra-subnet nodes communicate through wire links while inter-subnet communications are handled almost by single-hop wireless links. Simulation results show that HiWA efficiently reduces power consumption by 39% in comparison with a traditional wireless NoC, called WiNoC, while still achieves 16% lower packet latency than conventional NoC.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Electronic industry is changing in so swift a fashion which is, of course, the result of the permanent demand for innovation and technological advancement. Under these circumstances, manufacturers of electronic devices tend to enrich their products as much as possible; therefore, according to Moore's law, these devices are getting more and more complex. Today electronic devices require small, low-power and very fast chips to work with advanced applications. Many believe that emerging of System-on-Chip (SoC) has been an effective solution for advancement of the industry in the form of Moore's law [1,2]. As the number of cores integrated into a SoC increase the role played by the communication system becomes more and more important [3]. According to [4,5] there are two major limitations in SoC. The first limitation is the latency of long wires to connect Processor Elements (PEs) which by the decrease of technology scale, is changed into a bottleneck. The second limitation is integration of PEs with different standards and various producers on a single chip (i.e. Heterogeneous Computing). Each of these cores has its specific needs and limitations which has caused some problems for designing SoC.

The method of designing based on information which is presented as a solution for the existing challenges in designing SoC [1,4,6] has a key feature according to which, more efficient and at the same time simpler systems are tried to be provided. This

---

key feature includes separating information from communication. Network-on-Chip (NoC) is a very compressed system based on network technology in which PEs are connected to each other according to a communication infrastructure consisted of a switch, router and communication links called Interconnection Network. According to [7], despite the fact that NoC has a lot of advantages because of the presence of long wires in two-sided metal interconnects, it suffers from high latency and high power consumption.

At this juncture, by way of extremely high energy costs, single-core high-frequency processors are less considered and processors manufacturers are moving toward designing multi and many-core chips. Therefore, alternative technologies such as Wireless Network-on-Chip [8,9], 3D Network-on-Chip [10] and Photonic Network-on-Chip [11] were introduced. In this paper, a hybrid wireless network-on-chip architecture called Hierarchical Wireless-based Architecture (HiWA) along with performance evaluation parameters is introduced. HiWA is based on a 2D mesh that is divided into square subnets. Then, in each subnet if necessary one of the Conventional Routers (CRs) is replaced by a Wireless Router (WR) which has wireless connections with WRs in neighbors' subnets. Moreover, A WR placement algorithm as well as a routing algorithm is proposed for HiWA. In addition, reducing number of WRs to obtain a trade-off between latency and power consumption has been targeted. The rest of the paper is organized as follows. Section 2 review backgrounds and related works. The HiWA architecture is proposed in Section 3. Section 4 evaluates HiWA based on performance parameters. Finally, some conclusions are given in Section 5.

2. Backgrounds and related works

Recent progresses in silicon integrated circuit technology have permitted the integration of tiny transceivers antennas on a single chip, which results in introducing Wireless Network-on-Chip (Wireless NoC). In [12] transceivers for 60 GHz inter and intra-chip communications are designed. In [13] on-chip wireless transceivers are utilized to assist the progress of fast pre-bonding wafer testing enabled by direct accesses to components under test within the ICs. A low Terahertz (324 GHz) frequency generator is fulfilled in 90 nm CMOS [14]. Moreover, a signal source operating near 410 GHz that is fabricated using low-leakage transistors in a 6 M 45 nm digital CMOS technology is reported [15]. Based on these techniques, the output power level of the on-chip millimeter-wave generator can be as high as −1.4 dBm in the 32 nm CMOS process, which is large enough for on-chip short distance communication [16]. Following the rule of thumb in RF design, the highest available bandwidth is 10% of the carrier frequency. According to this experimental estimation, up to 16 channels can be available for wireless NoC in the range of 100–500 GHz. With recent developments of millimeter-wave circuits, bandwidths of hundred GHz will be reachable in near future. In addition to the bandwidth, wireless NoC requires low-power on-chip wireless transceivers. Silicon Mach-Zehnder electro-optic modulator at data rates up to 10 Gb/s with low RF power consumption of only 5 pJ/bit [17] is commercially available.

Beyond traditional wired interconnect solutions, different emerging approaches including 3D Network-on-Chip and Photonic Network-on-Chip were proposed [10,11]. In addition, the design of a wireless NoC based on CMOS UWB technology is introduced [18]. The antennas used in [18] achieve a transmission range of 1mm; therefore, for a typically die area of 20 mm×20 mm, this architecture requires multi-hop communication. Additionally, a broad survey regarding different wireless NoC architectures and their design doctrines is given in [19]. Furthermore, using miniaturized on-chip antennas as an enabling technology, a hybrid wireless NoC (WiNoC) is designed [7]. Complex design steps used in WiNoC require high hardware overhead to implement in micro-architecture level. Graphene or Carbon Nano-Tube (CNT) based on-chip antennas are anticipated to support high bandwidth wireless communication channels [20,21]. Since integration of CNT antennas with standard CMOS technology faces serious challenges, utilizing mm-wave CMOS transceivers running in the sub-THz frequency ranges seems to be a more feasible solution. Thus, in [22] mm-wave on-chip wireless antennas for both inter and intra-chip communications are modeled and evaluated.

In this paper, a hierarchical wireless network-on-chip architecture is introduced. Although several aspects of the proposed architecture are addressed in the paper, HiWA is a flexible platform that different placement and routing algorithms can be applied to it without changing the architectural structure. With optimal placement of WRs a trade-off between latency and power consumption parameters is obtained. The trade-off criteria are easily changeable according to the application where HiWA will be used.

3. Proposed architecture

3.1. Topology

The backbone of the proposed Hierarchical Wireless-based Architecture, HiWA, is based on 2D mesh NoC. The topology is shaped in such a way that first 2D mesh is divided into square subnets; then, in each subnet if necessary one of Conventional Routers (CRs) is replaced by a Wireless Router (WR) which has wireless links with WRs in neighbors' subnets. WRs are capable of performing both wired and wireless communication.

3.2. Addressing

Addressing method in HiWA is formed of X_{subnet}, Y_{subnet}, X_{local}, and Y_{local} in which the first two fields indicate the location of subnet and the next two fields determine the place of local node (router) in the subnet. Separating local and subnet address fields results in a simple design of hierarchical systems. Besides, it decreases hardware complexity of routers.
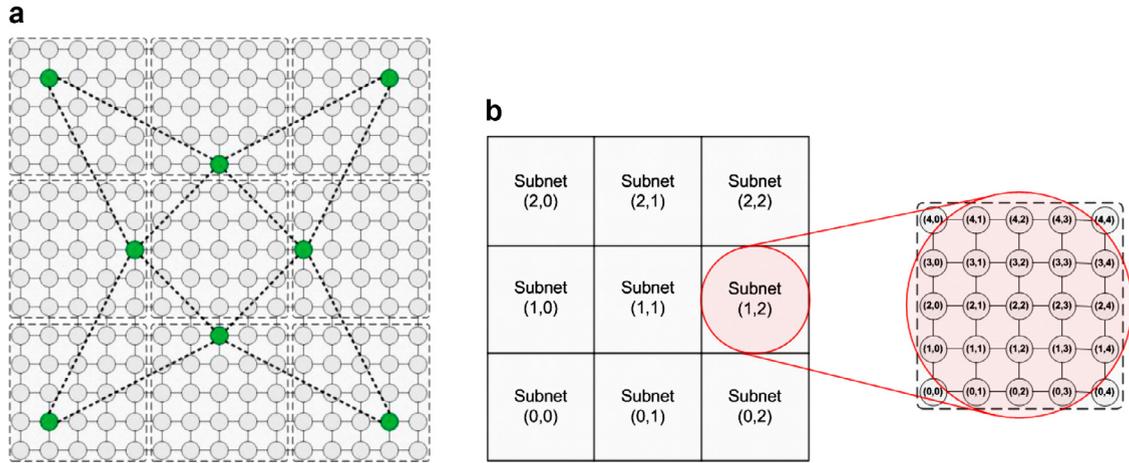
a

b

Fig. 1. (a) Demonstration of a 225-node HiWA (Light-colored nodes are CRs and dark-colored nodes are WRs) (b) 225-node HiWA topology and its addressing method.
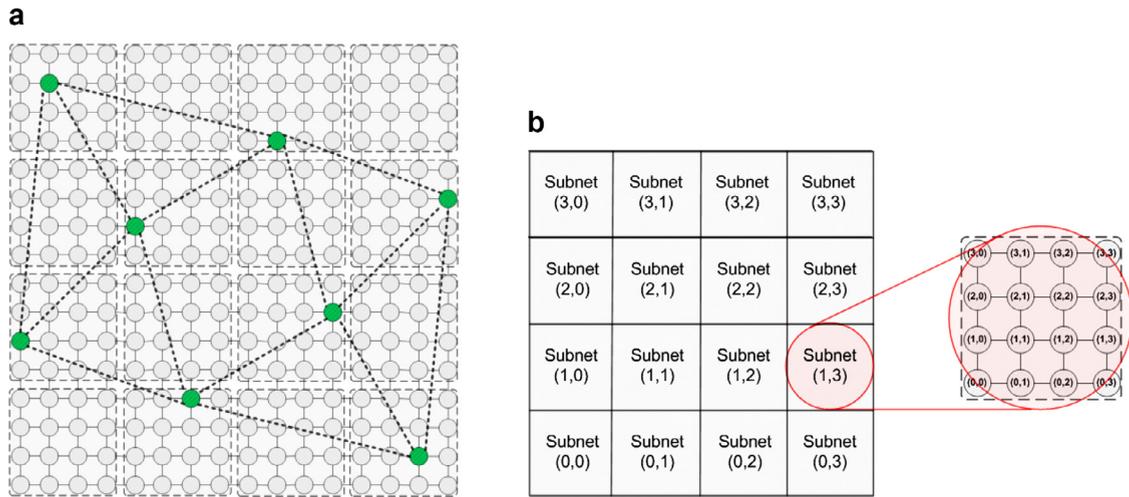
a

b

Fig. 2. (a) Demonstration of a 256-node HiWA (Light-colored nodes are CRs and dark-colored nodes are WRs) (b) 256-node HiWA topology and its addressing method.

Fig. 1a shows a 225-node HiWA which is divided into 9 subnets of 5×5 nodes. 8 WRs are used and subnet (1,1) does not have any WR. According to an experimental estimation which is obtained through simulation results, for NoCs of 225 and 256 nodes, using 8 WRs is the best option to establish a trade-off between latency and power consumption parameters. Fig. 2a shows a 256-node HiWA which is divided into 16 subnets of 4 × 4 nodes. Again 8 WRs are used and half of the subnets don't have any WR. Figs. 1b and 2b represent addressing method of the HiWA. These architectures are easily scalable to 900 and 1024 nodes by quadruplicate each one and use 24 WRs.

3.3. Wireless router placement algorithm

The purpose of WR placement is to allocate optimal minimum number of WRs across the network. Once the network is initialized, an optimization by means of Ant Colony Optimization (ACO) heuristic [23] is performed. ACO is a population based solution for combinatorial optimization problems that is inspired by the foraging behavior of ants and their inherent ability to find the shortest path from a food source to their nest. ACO has attracted a lot of attention in the fields of discrete problems due to its population-based search capability as well as simplicity and robustness. ACO used heuristic technique to produce an appropriate initial solution and determine an efficient search direction depending on the experience.

In reality, ants search for food sources in a random way. Shortly after an ant discovers a food source, they carry some food back to their colony. Fig. 3 illustrates the real ant colony behavior. When they move along the paths, they lay a chemical substance called pheromone as they travel. In turn, shorter paths will have the higher rate of pheromone trails. Each ant makes decisions by using pheromone trails as a communication mechanism. Strength of pheromone trail deposited on the ground depends on the quality of the solution (food source) found. Pheromone trails accumulate with multiple ants in shorter paths, which cause
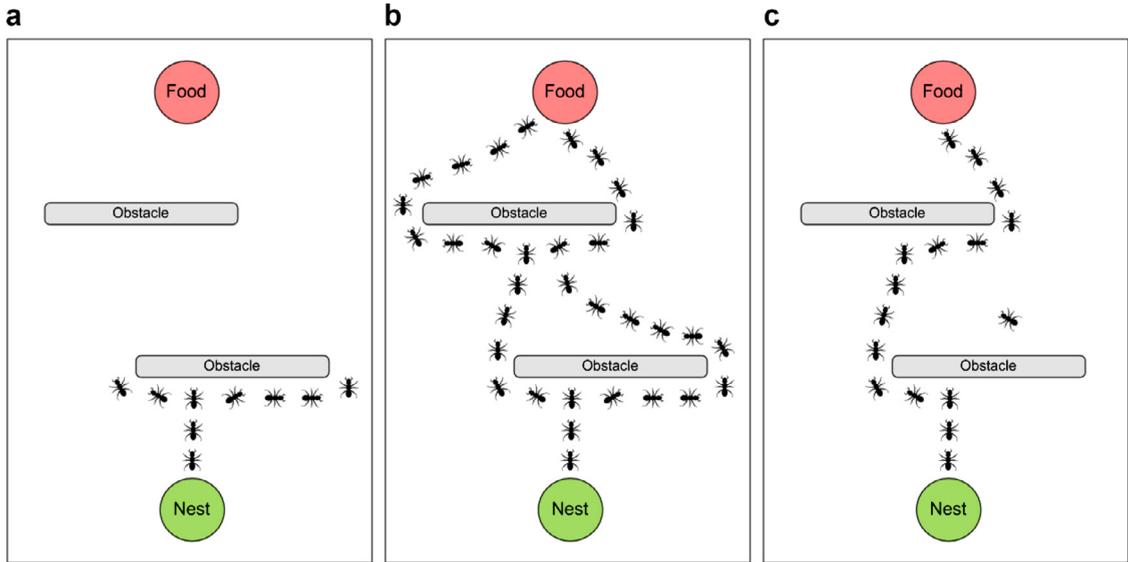
a   **b**   **c**

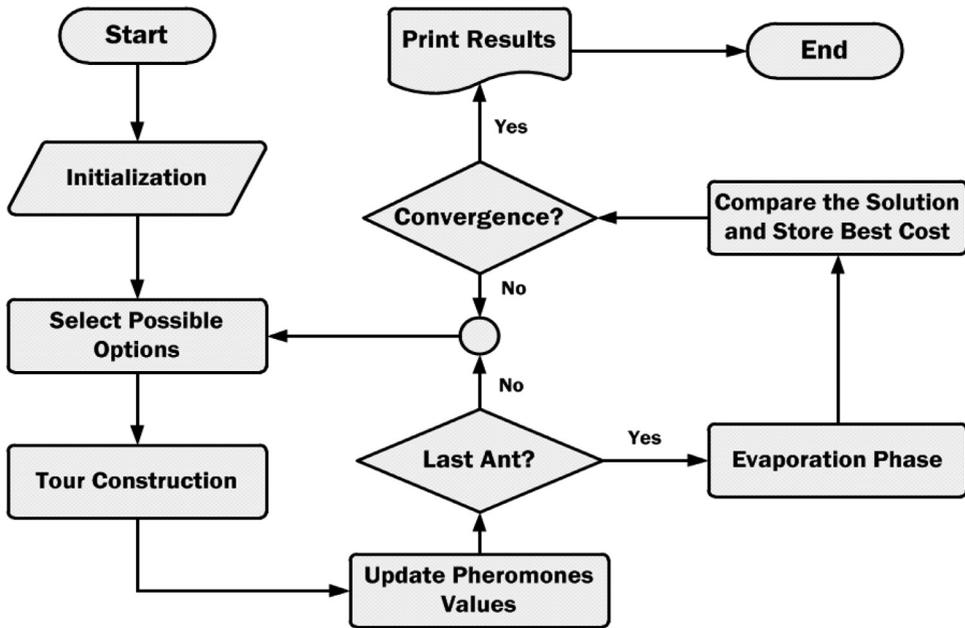Fig. 3. Illustration of the real ant colony behavior.

Fig. 4. WR placement flowchart using ACO.

a higher density than longer paths, thus increasing its attractiveness. All pheromone trails are eventually reduced by an evaporation rate. On the other side, a process of evaporation presents the exploration and prevents stalling in a local minimum. In addition, the pheromone values are updated at the end of each iteration. Fig. 4 shows the WR placement flowchart.

  The most vital subject of using ACO algorithm in WR placement is handling of pheromone trails in a proper way. In this case we can choose the nodes as the pheromone trail loader. After deciding an initial pheromone value for each node, the value of pheromone trail will renew with the placement method going on. For the WR placement algorithm, the pheromone trail is updated based on the following formula:

$$\gamma_{klmn,j} = \begin{cases} \gamma_{min} & if \quad \gamma_{klmn,\,j} < \gamma_{min} \\ (1-\alpha)\gamma_{klmn,j} + \Delta\gamma_{klmn,\,j} & if \quad \gamma_{min} \leq \gamma_{klmn,\,j} \leq \gamma_{max} \\ \gamma_{max} & if \quad \gamma_{klmn,\,j} > \gamma_{max} \end{cases} \tag{1}$$

In formula (1), $\gamma_{klmn,j}$ presents the pheromone value of assigning WR_j into $node_{klmn}$ and $0 \leq \alpha \leq 1$ shows evaporation coefficient in ACO algorithm. The pheromone will reduce as the placement goes on. Choosing small amount of α (i.e. low evaporation) leads to slow adaptation, while choosing large amount of α (i.e. high evaporation) results in fast adaptation. Although ACO is capable of adaptation because of the pheromone evaporation, the time required for adaptation to the current setting depends on both the size of the network and the alternation degree. When the alternation is uncompromising, then it may take longer to exclude inutile pheromone trails; therefore, in this case choosing a high evaporation rate is more appropriate. More precisely, a high evaporation rate will ignore the high intensity of pheromone trails that are concentrated to the optimum of the previous setting that is caused by stagnation behavior. On the contrary, a high pheromone evaporation rate may demolish the information that can be utilized on further settings, since any unacceptable solution in the current setting may be acceptable in the next setting [24].

If $node_{klmn}$ is selected by WR_j in the current placement, the pheromone trail of $node_{klmn}$ will be updated. In addition $\Delta\gamma_{klmn,j}$ is computed according to the following formula:

$$\Delta\gamma_{klmn,\,j} = \frac{C_{user}}{C_{current}} \tag{2}$$

In formula (2) $C_{current}$ is the cost value of current placement and C_{user} is a user-defined parameter. Since other nodes except $node_{klmn}$ are not selected in this placement, the pheromone trail is decreased only for them. (i.e. $\Delta\gamma_{klmn,j}$ is considered as zero for these nodes in this placement). Updating the pheromone trail of each node like above, will make the WR placement a local search algorithm. In order to avoid this situation, a Max-Min Ant System (MMAS) [23] is applied. In this case, two values will be given for pheromone trail as maximum and minimum. In the formula (1), γ_{min} and γ_{max} present the minimum and maximum values of pheromone trial in the stage of placement.

Note that the aim is to have at most one WR in each subnet, if $node_{klmn}$ is selected by the WR_j, other possible nodes with the same first two indices as $node_{klmn}$ cannot be selected by other WRs (As it is mentioned in Section 3.2 the first two indices are subnet indices.) Generally it is not possible to find the optimal placement if the platform is considered to be used for dynamic mapping as for each new application flow, WRs might need new placement. However, this approach is more realistic for application specific many-core platform which supposed to load (map) specific applications on the system. So, ACO will be executed for a set of applications supposed to be mapped statically on the system such that it can find near optimal placement for WRs.

One of the critical parameters of ACO is convergence time. Convergence time of an ACO algorithm is defined as the average iteration time that the algorithm spends in converging to the optimal solution. In order to explain how powerful an ACO algorithm is, time complexity of the algorithm obtained from convergence time analysis is taken into account. Convergence time of ACO algorithm is divided into two categories. First, the convergence in value [23] which states that in an infinite iteration time, at least one ant could reach the optimal solution. Second, the convergence in solution [25] which states that in an infinite iteration time, the ACO algorithm can reach the optimal solution with a probability of "one". Moreover, a procedure for estimating the ACO time complexity is introduced in [26]. Based on the analysis of [26], the condition where the ACO algorithm converges in linear time complexity is also given. If we choose the ACO algorithm parameters based on these conditions, WR placement algorithm of HiWA will also converge in linear time.

3.4. Routing algorithm

An organized decision is required to select an appropriate path in HiWA, because when one node sends a packet to another node, it is possible to be transmitted using only wired paths or only wireless paths or a combination of wired and wireless paths. This can be seen as a hybrid network that has been characterized by adding express paths (wireless links) to a 2D mesh NoC; therefore, whether the packet will take or not take the express paths is an important decision to make. One of the benefits of partitioning is that intra-subnet communications are handled through wire paths while inter-subnets communications are a function of hop counts and congestion. Algorithm 1 represents pseudo-code of the routing algorithm.

Since each WR is shared by several nodes, there is a possibility of congestion. In order to balance utilization of wired and wireless networks, a balance parameter called δ is added to the algorithm.

$$\delta = C \times u \tag{3}$$

The values of δs depend on the network size and utilization of wireless network. As it is shown in Eq. (1), δ consists of two major parameters, one static parameter (C) that is defined as a ratio of WRs to CRs and a dynamic parameter (u) that exponentially increases by wireless link utilization. Generally, both increasing in network size and link utilization will result in increasing the values of δs. In each router, there is a table that keeps and updates values of δs based on different situations. By adding δs to the wireless paths, lower priority for traffic balancing is given to them; therefore, congestion will be reduced. Dynamic allocation of wireless paths, ultimate to decrease congestion; Moreover, in a light traffic, more packets can utilize available wireless paths. More discussion about congestion reduction in HiWA is introduced in [27].

In the case that XY routing is used in lower levels, the complexity of HiWA routing algorithm is $O(\sqrt{n})$. The worst case scenario is when a packet should pass through the diameter of the network. In this case, it requires $2 \times (\sqrt{n} - 1)$ hops to route, in which n is the number of cores.
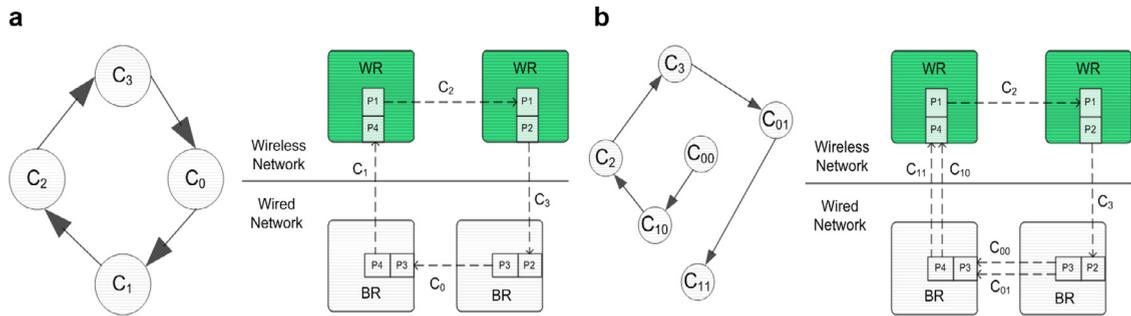
Fig. 5. A deadlock prone situation in HiWA (a) Without using virtual channels (b) By using virtual channels.

Table 1
Simulation parameters.

Parameter	Characteristic
Technology	65 nm
Clock frequency	1 GHz
Number of cores	225, 256
Switching	Wormhole
Routing	XY
Wireless communication	mm-Wave Antennas
MAC	FDMA
Number of virtual channels	2
Flit length	64 bits

3.5. Deadlock avoidance

One of the principal subjects should be targeted in networks using wormhole switching is deadlock avoidance. Although using XY routing in each wired and wireless network guarantees deadlock free packet transmission, when packets transmit through both wired and wireless paths, there is a possibility of graph dependency and deadlock as it is shown in Fig. 5.

In order to overcome the problem, virtual channels are taken into account. In each input port of the routers two sets of virtual channels are used. One of them is for traffic transmission using nearest WR while the other one is for traffic transmission using wired links or traffic transmission of the WR to the destination node. However, any deadlock-free routing can be applied in wired and wireless networks based on Algorithm 1.

Note that there are two famous deadlock solutions in the literature: deadlock prevention (i.e. design system so that deadlock is impossible) and deadlock avoidance (i.e. steer around deadlock with smart scheduling). In deadlock avoidance, the system dynamically considers every request and decides whether it is safe to grant it at this point, thus it allows more concurrency. On the other hand, in deadlock prevention, the goal is to ensure that at least one of the necessary conditions for deadlock can never hold. In HiWA, as well as the possibility of graph dependency that is described in Fig. 5, only a finite amount of processes at a time can use a resource and no preemption is allowed; therefore, all the conditions for a deadlock prone situation are met and applying any deadlock prevention approach is not possible.

4. Simulation results

In this section, 225-node HiWA (Fig. 1) and 256-node HiWA (Fig. 2) are compared with WiNoC [7] as a traditional wireless NoC, and basic 2D mesh NoC to evaluate the performance and feasibility of the proposed architecture. Comparisons are based on average hop count, normalized latency and normalized power consumption parameters.

4.1. Simulator and simulation parameters

Simulations are conducted using an open-source simulator called XMulator [28] a listener-based integrated simulation platform for interconnection networks. XMulator is a discrete event simulator that extensively uses XML format for defining topologies, parameters, and outputs that increases flexibility to deal with different types of inputs and different forms of results. Moreover, for calculating the power consumption, Orion 2.0 [29] library is added to the simulator. A notable feature of Orion is presenting hierarchical models. The wireless transceiver is designed with TSMC 65-nm standard CMOS process to obtain its power and delay characteristics and are fed in the simulator. Table 1 shows a summary of simulation of parameters. In

Algorithm 1: HiWA routing algorithm.

Input: Buffer usage and incoming packet of local wireless routers
Output: Routing path
s: source node
d: destination node
WR_s: source nearest wireless router
WR_d: destination nearest wireless router
H_B: travelling distance between s and d without using wireless links
H_w: travelling distance between s and d using both wireless and wired links

if s and d are in the same subnet **then**
 | Route packets from s to d using wired path
else
 | **if** $H_w + \delta < H_B$ **then**
 | | Route packets in three steps:
 | | a. Route packets from s to WR_s using wired links
 | | b. Route packets from WR_s to WR_d using wireless links
 | | c. Route packets from WR_d to d using wired links
 | **else**
 | | Route packets from s to d using wired path
 | **end**
end

---

simulations, six trace-driven traffic patterns taken from Splash II [30] are used. Note that these traffic patterns are repeated to cover all the nodes of the system.

4.2. Optimal number of wireless routers

An experimental estimation which is obtained through simulation results under uniform traffic pattern, for two sizes of HiWA is presented in Fig. 6.

In this experimental estimation, HiWA is configured with two different mesh sizes, 255 and 256 along with different number of WRs compared with conventional NoC. According to this estimation, that is product of reduction in latency and power consumption, using 8 WRs is the optimal point to establish a trade-off between latency and power consumption parameters. Considering only the latency reduction for evaluation, it is obvious that the more the number of WRs, the less the latency. On the other hand, the results show that using more than 10 WRs will significantly increase power consumption of HiWA.

These architectures are easily scalable to 900 and 1024 nodes by quadruplicate each one and use 24 wireless routers. It means for each size of the HiWA we need to obtain the optimal number of the WRs. For example, by using 8 WRs for 900-node HiWA (instead of using 24 ones), although the overall latency will increase, the system power consumption will decrease. Our goal is to establish a trade-off between latency and power consumption parameters.

4.3. Hop count

Comparison of the average hop counts between HiWA, WiNoC, and conventional NoC is shown in Fig. 7. The results state that both HiWA and WiNoC reduce average hop counts 42% and 51% respectively in comparison with conventional NoC. Since the wireless links act as shortcut paths, they decrease travel distance of packets from source to destination effectively.

4.4. Latency

Fig. 8 shows the performance gains of using wireless links. The results indicate that both HiWA and WiNoC reduce average packet latency 16% and 19% respectively in comparison with conventional NoC. This reduction is because of using WRs. Note that saving in hop counts is not directly convertible to latency reduction. In the current technologies bandwidth of wireless links is smaller than bandwidth of wired links. Consequently congestion occurs in WRs and overhead of packet blocking will be added to latency. According to ITRS [31] in 16nm CMOS technology gain frequency and power gain will reach in 600 GHz and 1 THz respectively; therefore, wireless NoCs will reach much more reduction in latency compared to conventional NoCs.

4.5. Power consumption

Fig. 9 shows power consumption comparison between different architectures. On average, 14% and 39% reduction is observable in total power consumption in HiWA versus conventional NoC and WiNoC respectively. Since the power consumption in a NoC originates from the operation of the PEs and the interconnection components between those PEs, it is proportional to the
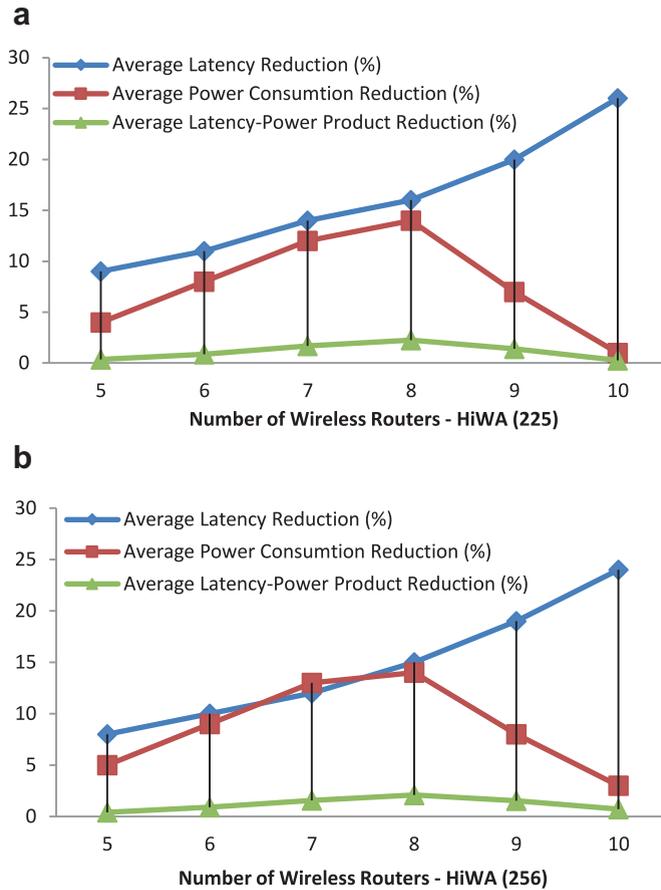
a

b

Fig. 6. Optimized number of wireless nodes under uniform traffic pattern (a) 225-node HiWA (b) 256-node HiWA.
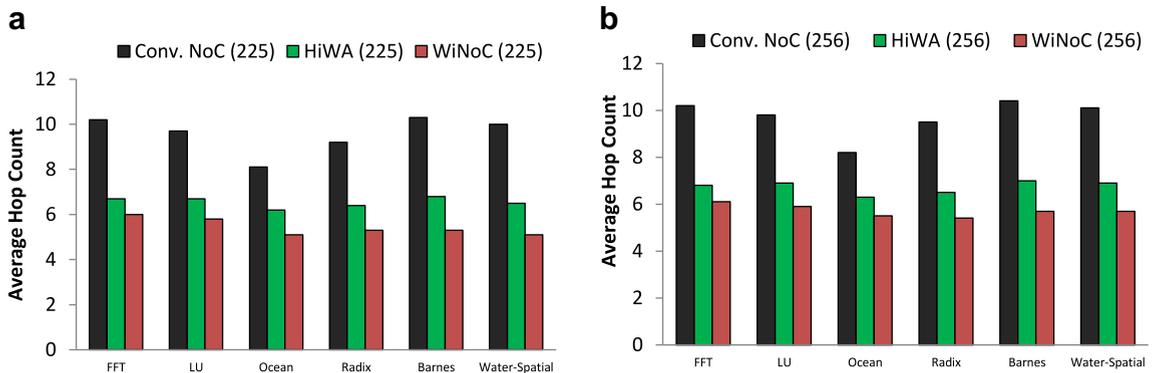
a

b

Fig. 7. Hop count comparison (a) 225-node HiWA, conventional NoC, and WiNoC (b) 256-node HiWA, conventional NoC, and WiNoC.

switching activity arising from packets moving across the network. In addition, router buffers have the greatest impact on power consumption. This improvement is caused by transmission speed up (reducing average hop counts) and reduction of average power consumption in buffers. On the other hand, as WRs consumes more power than CRs, increasing the number of WRs (as it's in WiNoC) severely increases total power consumption of the system. Thus, we need to obtain a trade-off between latency and power consumption parameters based on each system requirements.
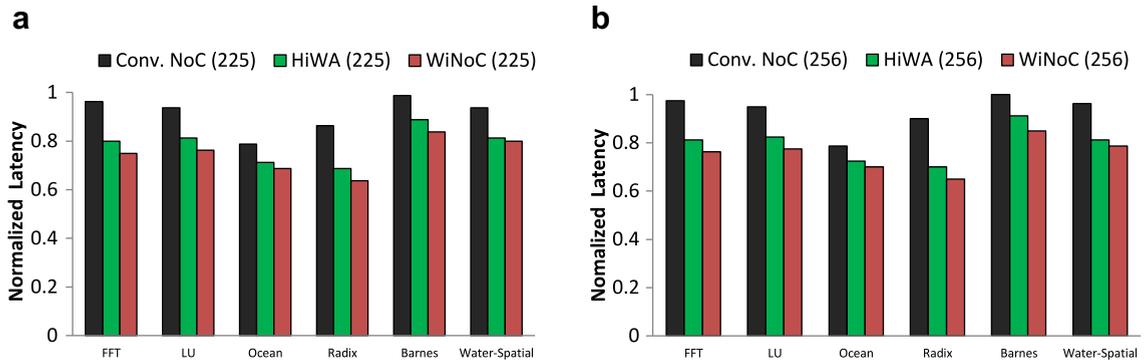
Fig. 8. Latency comparison (a) 225-node HiWA, conventional NoC, and WiNoC (b) 256-node HiWA, conventional NoC, and WiNoC.
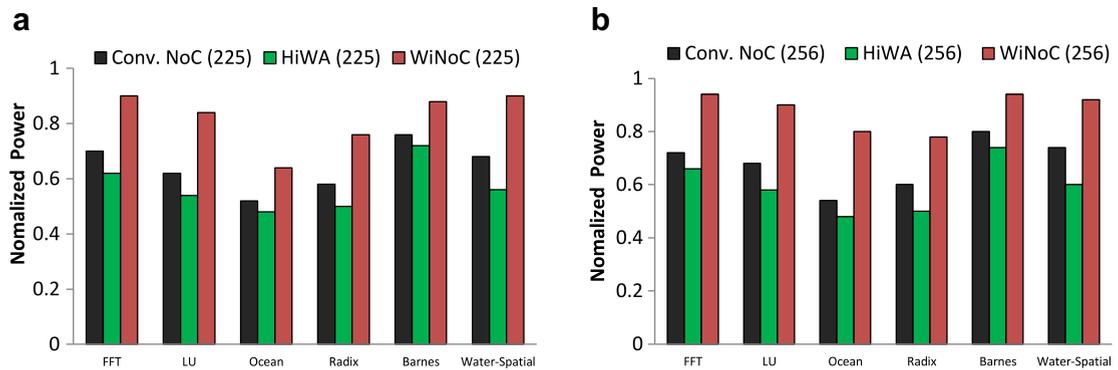

Fig. 9. Power comparison (a) 225-node HiWA, conventional NoC, and WiNoC (b) 256-node HiWA, conventional NoC, and WiNoC.

5. Conclusion

In this paper, a flexible hybrid wireless network-on-chip architecture called Hierarchical Wireless-based Architecture, HiWA, with wired and wireless communication was proposed. Although several aspects of the proposed architecture were addressed in the paper, HiWA is a flexible platform that different placement and routing algorithms with low hardware overhead can be applied to it without changing the architectural structure. In addition, according to simulation results HiWA efficiently reduces power consumption by 39% in comparison with WiNoC, while still achieves 16% lower packet latency than conventional NoC. The trade-off criteria are easily changeable according to the application where HiWA will be used.

References

[1] Benini L, Micheli GD. Networks on chips: a new SoC paradigm. IEEE Comput 2002;35(1):70–8.
[2] Horowitz M, Dally WJ. How scaling will change processor architecture.. Proceedings of the IEEE international solid-state circuits conference (ISSCC); 2004. Vol 1 p. 132–3.
[3] Palesi M, Daneshtalab M. Routing algorithms in networks-on-chip. Springer; 2014.
[4] Fu F, Sun S, Song J, Wang J, Yu M. A NoC performance evaluation platform supporting designs at multiple levels of abstraction. Proceedings of the IEEE conference on industrial electronics and applications; 2009. p. 425–9.
[5] Pande PP, Grecu C, Jones M, Ivanov A, Saleh R. Performance evaluation and design trade-offs for network-on-chip interconnect architectures. IEEE Trans Comput 2005;54(8):1025–40.
[6] Benini L, Bertozzi D. Network-on-chip architectures and design methods. In: Proceedings of computers and digital techniques; 2005. p. 261–72.
[7] Pande PP, Ganguly A, Chang K, Teuscher C. Hybrid wireless network on chip: a new paradigm in multi-core design. In: Proceedings of international workshop on network on chip architectures (NoCArc); 2009. p. 71–6.
[8] Chang MF, Cong J, Kaplan A, Naik M, Reinman G, Socher E, Tam SW. CMP network-on-chip overlaid with multi-band RF-interconnect. In: Proceedings of IEEE international symposium on high performance computer architecture (HPCA); 2008. p. 191–202.
[9] Rezaei A, Safaei F, Daneshtalab M, Tenhunen H. HiWA: a hierarchical wireless network-on-chip architecture. In: Proceedings of IEEE international high performance computing & simulation (HPCS); 2014. p. 499–505.
[10] Pavlidis VF, Friedman EG. 3-D topologies for networks-on-chip. IEEE Trans Very Large Scale Integr 2007;15(10):1081–90.
[11] Shacham A, Bergman K, Member S, Carloni LP. Photonic networks-on-chip for future generations of chip multiprocessors. IEEE Trans Comput 2008;57(9):1246–60.
[12] Laha S, Kaya S, Matolak DW, Rayess W, DiTomaso D, Kodi A. A new frontier in ultralow power wireless links: network-on-chip and chip-to-chip interconnects. IEEE Trans Comput-Aided Des Integr Circ Syst 2015;34(2):186–98.
[13] Chandran U, Zhao D. Cost-optimal design of wireless pre-bonding test framework. IEEE Int Syst Chip Conf (SOCC) 2014:324–9.
[14] Huang D, LaRocca T, Chang M-C, Samoska L, Campbell AFung,R, Andrews M. Terahertz CMOS frequency generator using linear superposition technique. IEEE J Solid-State Circ 2008;43(12):2730–8.

[15] Seok E, Cao C, Shim D, Arenas DJ, Tanner DB, Hung C, O KK. A 410GHz CMOS push-push oscillator with an on-chip patch antenna. IEEE Int Solid-State Circ Conf (ISSCC) 2008:472–629.
[16] Lee SB, Tam SW, Pefkianakis I, Lu S, Chang MF, Guo C, Reinman G, Peng C, Naik M, Zhang L, Cong J. A scalable micro wireless interconnect structure for CMPs. In: Proceedings of the international conference on mobile computing and networking (MobiCom); 2009. p. 217–28.
[17] Green W, Rooks M, Sekaric L, Vlasov Y. Ultra-compact, low RF power, 10 Gb/s silicon Mach–Zehnder modulato. Opt Express 2007;15(25):17106–13.
[18] Zhao D, Wang Y. SD-MAC: design and synthesis of a hardware-efficient collision-free QoS-aware MAC protocol for wireless network-on-chip. IEEE Trans Comput 2008;57(9):1230–45.
[19] Deb S, Ganguly A, Pande PP, Belzer B, Heo D. Wireless NoC as interconnection backbone for multicore chips: promises and challenges. IEEE J Emerg Select Top Circ Syst 2012;2(2):228–39.
[20] Abadal S, Alarcón E, Cabellos-Aparicio A, Lemme M, Nemirovsky M. Graphene-enabled wireless communication for massive multicore architectures. IEEE Commun Mag 2013;51(11):137–43.
[21] Ganguly A, Chang K, Deb S, Pande PP, Belzer B, Teuscher C. Scalable hybrid wireless network-on-chip architectures for multicore systems. IEEE Trans Comput 2011;60(10):1485–502.
[22] Lin; J, Wu H-T, Su Y, Gao L, Sugavanam A, Brewer JE, K O K. Communication using antennas fabricated in silicon integrated circuits. IEEE J Solid-State Circ 2007;42(8):1678–87.
[23] Dorigo M, Stutzle T. Ant colony optimization. MIT Press; 2004.
[24] Mavrovouniotis M, Yang S. Adapting the pheromone evaporation rate in dynamic routing problems. Springer Berlin Heidelberg; 2013.
[25] Dorigo M, Blum C. Ant colony optimization theory: a survey. Elsevier J Theor Comput Sci 2005;344(2-3):243–78.
[26] Huang H, Wu C-G, Hao Z-F. A pheromone-rate-based analysis on the convergence time of ACO algorithm. IEEE Trans Syst, Man, Cybern, Part B: Cybern 2009;39(4):910–23.
[27] Rezaei A, Daneshtalab M, Zhao D, Safaei F, Wang X, Ebrahimi M. Dynamic application mapping algorithm for wireless network-on-chip. In: Proceedings of IEEE euromicro conference on parallel, distributed and network-based computing (PDP); 2015. p. 421–4.
[28] Nayebi A, Meraji S, Shamaei A, and Sarbazi-Azad H. "XMulator: A listener-based integrated simulation platform for interconnection networks," In Asia International Conference on Modeling & Simulation (AMS), pp. 128-132, 2007.
[29] Kahng A, Li B, Peh L, Samadi K. Orion 2.0: A power-area simulator for interconnection networks. IEEE Trans Very Large Scale Integr (VLSI) Syst 2012;20(1):191–6.
[30] Woo S.C., Ohara M, Torrie E, Singh J.P., and Gupta A. "The SPLASH-2 programs: characterization and methodological considerations," In International Symposium on Computer Architecture (ISCA), pp. 24-36, 1995.
[31] ITRS. International Technology Roadmap for Semiconductors, 2007 edition.

Amin Rezaei is currently pursuing his Ph.D. studies at University of Louisiana at Lafayette, the USA. He received his B.Sc. degree in Computer Engineering from University of Isfahan, Iran, in 2011 and M.Sc. degree in Computer Engineering from Shahid Beheshti University, Iran, in 2014. His main research interests include Parallel and Heterogeneous Computing, and Multi & Many Core System-on-Chips.

Masoud Daneshtalab is currently a European Marie Curie fellow in Department of Electronic and Embedded Systems at KTH Royal Institute of Technology, Sweden. He was lecturer and project manager at University of Turku in Finland from 2012–2014. He is a member of IEEE and has published one book, four book chapters, and over 150 refereed international journals and conference papers.

Farshad Safaei received his B.Sc., M.Sc., and Ph.D. degrees in Computer Engineering from Iran University of Science and Technology in 1994, 1997 and 2007, respectively. He is currently an assistant professor in the Department of Computer Science and Engineering, Shahid Beheshti University, Iran. His research interests include Performance Modeling & Evaluation, Interconnection Networks, Complex Networks, and High Performance Computer Systems.

Danella Zhao is currently a Lockheed Martin endowed associate professor with the Center for Advanced Computer Studies, University of Louisiana at Lafayette. She gained her M.Sc. and Ph.D. from the CSE Department, State University of New York at Buffalo in 2001 and 2004 respectively. She received the NSF Career Development Award in 2009 and JAPS Fellowship Award in 2006.