Microprocessors and Microsystems 35 (2011) 119–129

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

On an efficient NoC multicasting scheme in support of multiple applications running on irregular sub-networks

Xiaohang Wang [a,b], Mei Yang [b], Yingtao Jiang [b], Peng Liu [a,*]

[a] Department of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, Zhejiang 310027, PR China
[b] Department of Electrical and Computer Engineering, University of Nevada, Las Vegas 89154, USA

ARTICLE INFO

Article history:
Available online 17 August 2010

Keywords:
Network-on-chips (NoCs)
Chip multiprocessor (CMP)
Multicast
Routing

ABSTRACT

When a number of applications simultaneously running on a many-core chip multiprocessor (CMP) chip connected through network-on-chip (NoC), significant amount of on-chip traffic is one-to-many (multicast) in nature. As a matter of fact, when multiple applications are mapped onto an NoC architecture with applicable traffic isolation constraints, the corresponding sub-networks of these applications are mapped onto actually tend to be irregular. In the literature, multicasting for irregular topologies is supported through either multiple unicasting or broadcasting, which, unfortunately, results in overly high power consumption and/or long network latency. To address this problem, a simple, yet efficient hardware-based multicasting scheme is proposed in this paper. First, an irregular oriented multicast strategy is proposed. Literally, following this strategy, an irregular oriented multicast routing algorithm can be designed based on any regular mesh based multicast routing algorithm. One such algorithm, namely, Alternative Recursive Partitioning Multicasting (AL + RPM), is proposed based on RPM, which was designed for regular mesh topology originally. The basic idea of AL + RPM is to find the output directions following the basic RPM algorithm and then decide to replicate the packets to the original output directions or the alternative (AL) output directions based on the shape of the sub-network. The experiment results show that the proposed multicast AL + RPM algorithm can consume, on average, 14% and 20% less power than bLBDR (a broadcasting-based routing algorithm) and the multiple unicast scheme, respectively. In addition, AL + RPM has much lower network latency than the above two approaches. To incorporate AL + RPM into a baseline router to support multicasting, the area overhead is fairly modest, less than 5.5%.

1. Introduction

Advance in technology continues to drive the increase of transistor integration capacity. It is estimated that by 2015, there will be 100 billion transistors integrated on a 300 mm^2 die [1]. To exploit this large number transistors and also take into consideration of pressing high power consumption of ever bigger chips, the design paradigm is migrating to many-core architectures [1,2]. Network-on-chip (NoC) [3] has been proposed as the mainstream on-chip network architecture to efficiently interconnect the large number of (16 or more) processing cores integrated on a many-core system. Some most recent, high profile examples include Intel's Teraflop [4] and Tilera [5] chips featuring many-core chip multiprocessors (CMPs) architectures with 2D mesh topologies [13] for on-chip interconnect.

With the development of diverse applications and programming models on CMPs, one-to-many communication and one-to-all communication are becoming more common. For example, in CMPs with cache coherent shared memory systems, the cache coherence protocols exhibit one-to-many communication characteristics to keep the ordering of different requests or to invalidate shared data on different cache nodes [6]. In [7], it has been observed that 5–10% of the network traffic is one-to-many in nature, ranging from scientific workloads to commercial workloads, in communication traces of different cache coherence protocols and operand network. Therefore, efficient support of one-to-many communications in CMPs, particularly hardware multicast support, will benefit a wide range of applications by boosting the network performance with reduced power consumption. Unfortunately, up to date, there is only very limited number of chip router designs that actually support multicasting [6–8].

In addition, the following issues make multicast supporting even more complicated. The first issue is topology irregularity. The large number of cores on a CMP unquestionably offers high parallelism in computation. To better utilize these vastly available computation resources, virtualization of the chip becomes a

* Corresponding author at: Department of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, Zhejiang 310027, PR China.

E-mail addresses: baikeina@yahoo.com.cn (X. Wang), Mei.Yang@unlv.edu (M. Yang), yingtao@egr.unlv.edu (Y. Jiang), liupeng@isee.zju.edu.cn (P. Liu).

0141-9331/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.micpro.2010.08.003

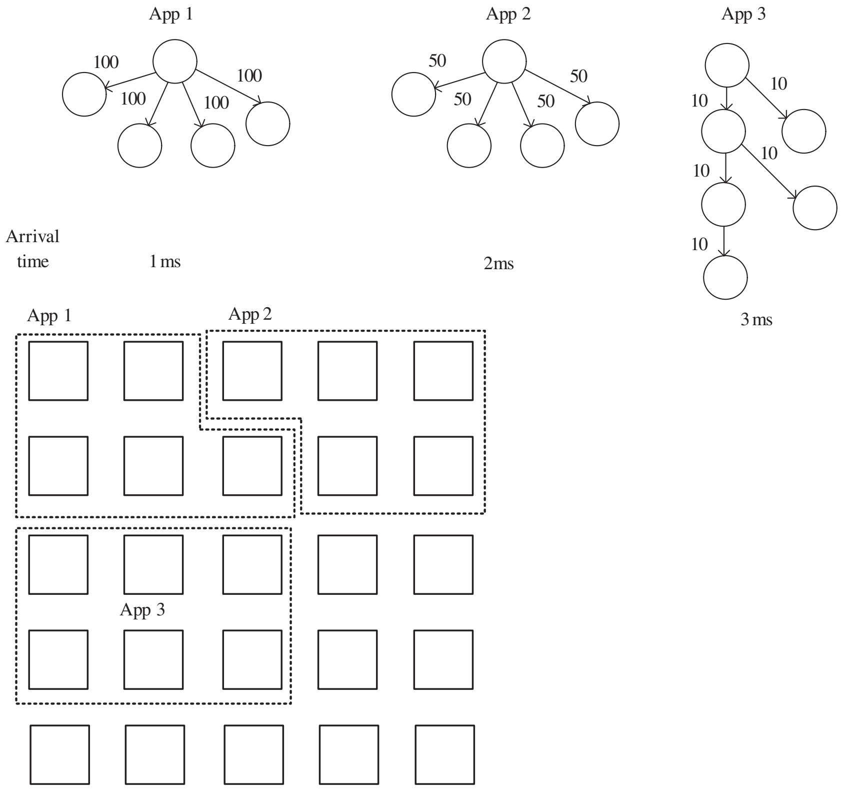


Fig. 1. Sub-network partition and task mapping of multiple applications on a 5 × 5 based mesh NoC.

necessity [9], where resources can be distributed among different virtual machines [3]. Applying virtualization [8] at the NoC level basically allows a single NoC-based CMP to be shared by multiple applications with each mapped to different sub-networks of the chip [10] either statically [11] or dynamically [12]. Fig. 1 shows an example with three applications arriving at 1 ms, 2 ms, and 3 ms. The three applications are allocated to three sub-networks which may not be regular shapes (e.g., 2D mesh, torus). On the other hand, virtualization requires traffic isolation [8]; that is, communication between nodes in a virtualized region is limited to the sub-network only. The irregular sub-network and traffic isolation requirements together negate regular 2D mesh oriented routing algorithms, like *XY* routing, odd–even routing, etc. [13].

The second issue is unpredictability of the application communication behavior. Different types of applications, such as desktop, server, embedded systems, will be executed on general purpose CMPs. It is impossible to pre-characterize the communication patterns among the cores inside a sub-network. As a result, customized NoC routing approaches (like the ones using routing tables [14]) may not be feasible.

Hence, it is important to design an efficient multicast mechanism which supports irregular topologies without the need of a routing table. In this paper, an irregular sub-network oriented multicast strategy is first proposed. Following this strategy, an irregular sub-network oriented multicast routing algorithm, namely, Alternative Recursive Partitioning Multicast (AL + RPM), is developed based on RPM [13], an efficient deterministic multicast routing algorithm proposed for regular mesh topology. To our best knowledge, our approach is the first multicast routing approach, as opposed to the broadcast-based one [8], that targets to irregular sub-networks.

In the rest of the paper, Section 2 reviews the existing work on multicast routing schemes in NoCs. Section 3 presents the preliminaries. Section 4 describes the irregular sub-network oriented multicast routing strategy and algorithm. Section 5 reports the performance evaluation of AL + RPM. Finally, Section 6 concludes the paper.

2. Related work

Multicast communication has been extensively studied in computer networks and interconnection networks [13]. However, due to the power and area constraints pertaining to NoCs, supporting multicast in NoCs has a different set of requirements. Particularly, an efficient multicasting approach for NoCs should result in low network latency and low power and area consumptions. A simple multicasting approach is to send a multicast packet as multiple unicast packets. However, such a scheme suffers from very large network latency and high power consumption [7]. Below reviews existing multicasting approaches proposed for mesh-based NoCs.

The multicast problem of regular mesh topology has been studied as in [13]. Generally, there are two types of multicast routing strategies, namely, path-based [15] and tree-based [6,7,16–18] multicast. Path-based multicast routing is to deliver the packet to each destination sequentially following one path [15]. Path-based multicast is attractive for its simplicity in hardware design. However, if the destination nodes are widely spread, path-based multicast may suffer higher latency compared to tree-based multicast. It is shown in [7] that path-based multicast may increase network latency by 48% compared to a unicast router.

Tree-based multicast routing [6,7,16–18] is to deliver the packet along a common path as far as possible and replicate packets (branch) for a unique set of destination nodes when necessary. Several tree-based multicast approaches have been proposed for NoCs with regular mesh topology. The virtual circuit tree-based multicast (VCTM) [7] avoids sending redundant packets as multiple unicast. However, it uses a lookup table based multicasting router which has high power and area overhead. The approaches in [16,19] extend the unicast XY routing to support multicasting, where a packet will always be sent to the *X* direction first and replicated if there are destinations in the *Y* direction. The approach in [19] is referred as multicast *XY* in later text. In [17,18], tree-based adaptive multicast routing approaches are proposed. The region partition multicast (RPM) [6] selects the replication points for multicast packets based on the distribution of destinations in the network partition. Each node partitions the whole network into at most eight regions according to its position. Replication decisions are made by checking the regions that the destination nodes fall in. The simulation results in [6] show that RPM improves the average packet latency by 50% and saves the router and link power by 25% compared with VCTM.

However, the aforementioned approaches [6,7,16–19] cannot support multicasting for irregular sub-networks. The bLBDR routing [8] proposed for collective communication in irregular sub-networks supports multicasting by broadcasting in sub-networks. In bLBDR, connectivity bits are used to define different sub-networks. However, the broadcast nature of this scheme makes the network congested and results in higher power consumption.

In this paper, we focus on designing hardware-based multicast routing scheme for irregular sub-networks and propose the alternative RPM routing scheme based on RPM [6]. Compared with the broadcast-based bLBDR, AL + RPM inherits the efficiency of tree-based multicast routing while provides the flexibility to support irregular sub-networks.

3. Preliminaries

3.1. Architecture and power models

The target NoC architecture is a tile based NoC, which is composed of $N \times N$ tiles interconnected by a 2-D mesh network. Each tile (node exchangeably), indexed by its coordinate (x, y) or its ID $xN + y$, where $0 \leqslant x \leqslant N-1$ and $0 \leqslant y \leqslant N-1$, has one processing core and one router. Each router (shown in Fig. 2) connects to its local processing core and four neighbour tiles through bidirectional channels. A 5×5 crossbar switch is used as the switching fabric of the router. The arbitration unit arbitrates the connection requests sent from the input ports so that each output receives data from at most one input port. At each input port, buffers are used to support virtual channels (VCs). The VC allocation unit controls the virtual channel allocation. The routing unit decides the output directions.

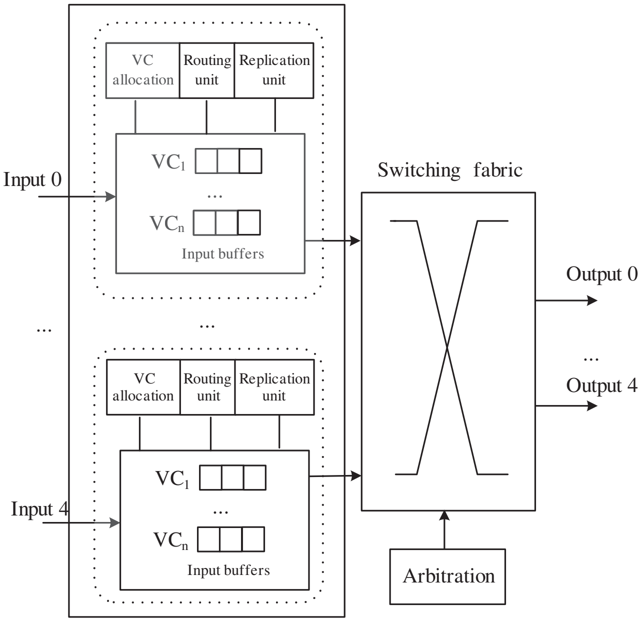


Fig. 2. Router architecture.

Assume wormhole switching is used here. To support multicast, the replication unit is used to make copy of flits of a multicast packet according to the decision of the routing unit. The replication is done inside the crossbar where a packet is forwarded to multiple output channels. Asynchronous replication [20] scheme is chosen as the replication approach. In asynchronous replication, multiple replicated flits are allowed to be forwarded independently. If one replicated flit is blocked, other replicated flits can be forwarded asynchronously.

The power model used in [21] is followed in this study. The bit power (E_{bit}) is defined as the power consumed when one bit of data is transported through a router, and it can be calculated as,

$$E_{bit} = E_{Sbit} + E_{Bbit} + E_{Wbit} \tag{1}$$

where E_{Sbit}, E_{Bbit}, and E_{Wbit} represent the power consumed by the switch, the buffer, and the interconnection wires inside the switching fabric, respectively. As explained in [21], E_{Bbit} and E_{Wbit} are negligibly small compared to E_{Sbit}. Hence, the average power consumption for a unicast communication which sends *BW* bits from source tile *s* to destination tile *t* can be represented as,

$$E^{s,t}_{Unicast} = \eta_{hops} \times E_{Sbit} + (\eta_{hops} - 1) \times E_{Lbit} \tag{2}$$

where η_{hops} is the number of routers traversed from tile *s* to tile *t*, E_{Sbit} is the power consumed by the switch, and E_{Lbit} is the power consumed on each link.

The average power consumption for a multicast communication which sends 1 bit from the source tile *s* to the set of destination tiles $\overline{D}$ can be represented as,

$$E^{s,\overline{D}}_{Multicast} = \eta_l \times E_{Lbit} + \eta_R \times E_{Sbit} \tag{3}$$

where η_R is the total number of routers and η_l is the total number of links that are on the multicast path from tile *s* to all tiles in *T*, respectively.

The objective of this work is that, given a sub-network of a 2D mesh NoC, design a hardware multicast support scheme, to (1) support irregular sub-networks, (2) avoid using routing tables, and (3) have low network latency and power consumption.

3.2. Assumptions and definitions

The following assumptions are made throughout the paper.

Assumption 1. The shape of the sub-network mapped with an application is near convex [12]. More specifically, we only consider such sub-networks that, there exists at least one minimal path (measured in hop counts) completely inside the sub-network for each pair of nodes inside the sub-network. Fig. 3a illustrates the

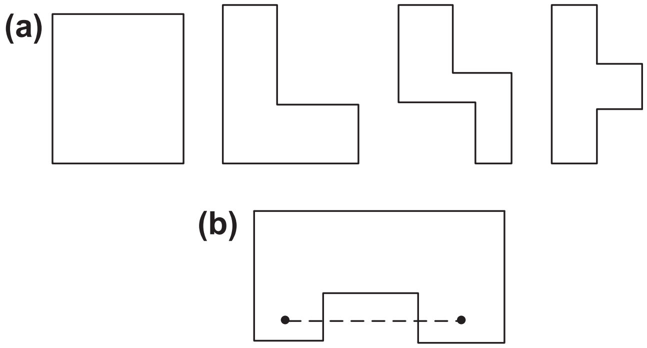


Fig. 3. (a) Sub-networks allowed. (b) Sub-networks not allowed.

sub-network shapes that are considered. Fig. 3b shows an example of the sub-network shape wherein the minimal path between the two dotted nodes is not completely inside the sub-network.

Assumption 2. The applications can be statically or dynamically mapped to the NoC-based CMPs. For dynamic mapping [22], a global manager processor (GM) is responsible for resource management.

Assumption 3. In each packet, the destination addresses are encoded in bit string [13].

To represent the shape of a sub-network, connectivity bits [8] are used at each router.

Definition 1 (*Connectivity bits*). Each router has four connectivity bits, C_N, C_E, C_S, and C_W, each defines the connectivity at the specific output direction. Suppose a tile has coordinate (x, y), C_N is 1 if tile (x, y) and its north neighbour tile $(x, y - 1)$ are in the same sub-network. Similarly, C_x is 1 if the tile and its neighbour tile on the x direction are in the same sub-network. For example, suppose the run time application mapping algorithm in [12] is used to map applications 1–3 (shown in Fig. 4a). The four connectivity bits of tile 6 is shown in Fig. 4b.

A tile may be shared by several sub-networks. For example, a cache memory may be shared by several applications. Fig. 4c shows that tile 2 is shared by two overlapping sub-networks. The connectivity bits in Definition 1 cannot describe overlapping sub-networks. As in [8], to support up to M overlapped sub-networks, each connectivity bit is extended to M bits indexed by the sub-network ID (Fig. 4c). For example, in Fig. 4d, C_N is extended into $C_{N[1]}, \ldots, C_{N[4]}$ if four sub-networks need be supported.

Definition 2 (*Extended connectivity bits*). Each router located at tile with coordinate (x, y) has $4 \times M$ connectivity bits, $\{C_{N[1]}, \ldots, C_{N[M]}\}$, $\{C_{W[1]}, \ldots, C_{W[M]}\}$, $\{C_{E[1]}, \ldots, C_{E[M]}\}$, and $\{C_{S[1]}, \ldots, C_{S[M]}\}$. Suppose a tile has coordinate (x, y), $C_{x[q]}$ $(q = 1, \ldots, M) = 1$ if tile (x, y) and its neighbour tile on the x direction are in the same sub-network with ID q. Extended connectivity (EC) bits EC_N, EC_E, EC_S, and EC_W are defined as follows. Given the sub-network ID q, $EC_N = C_{N[q]}$, $EC_W = C_{W[q]}$, $EC_E = C_{E[q]}$, $EC_S = C_{S[q]}$.

Fig. 4d shows that a MUX can be used to find the value of an extended connectivity bit from the connectivity bits given a sub-network ID. Fig. 4e and f shows the values of extended connectivity bits of tile 2 for sub-network ID 1 and 2, respectively. The connectivity bit registers can be set statically or by GM [22] with dynamic mapping.

In addition, we adopted the network partition concept from RPM [6].

Definition 3 [6] (*Region*). At each tile with coordinate (x, y), the network is partitioned into eight regions, $R_0, R_1, \ldots, R_8$. R_0 is the set of tiles with coordinates (x_0, y_0), where $x_0 > x$ and $y_0 < y$. R_1 is set of tiles with coordinates (x, y_1), where $y_1 < y$. R_2 is the set of tiles with coordinate (x_2, y_2) where $x_2 < x$ and $y_2 < y$. R_3 is the set of tiles with coordinates (x_3, y) where $x_3 < x$. R_4 is the set of tiles with coordinates (x_4, y_4) where $x_4 < x$ and $y_4 > y$. R_5 is set of tiles with coordinates (x, y_5) where $y_5 < y$. R_6 is the set of tiles with

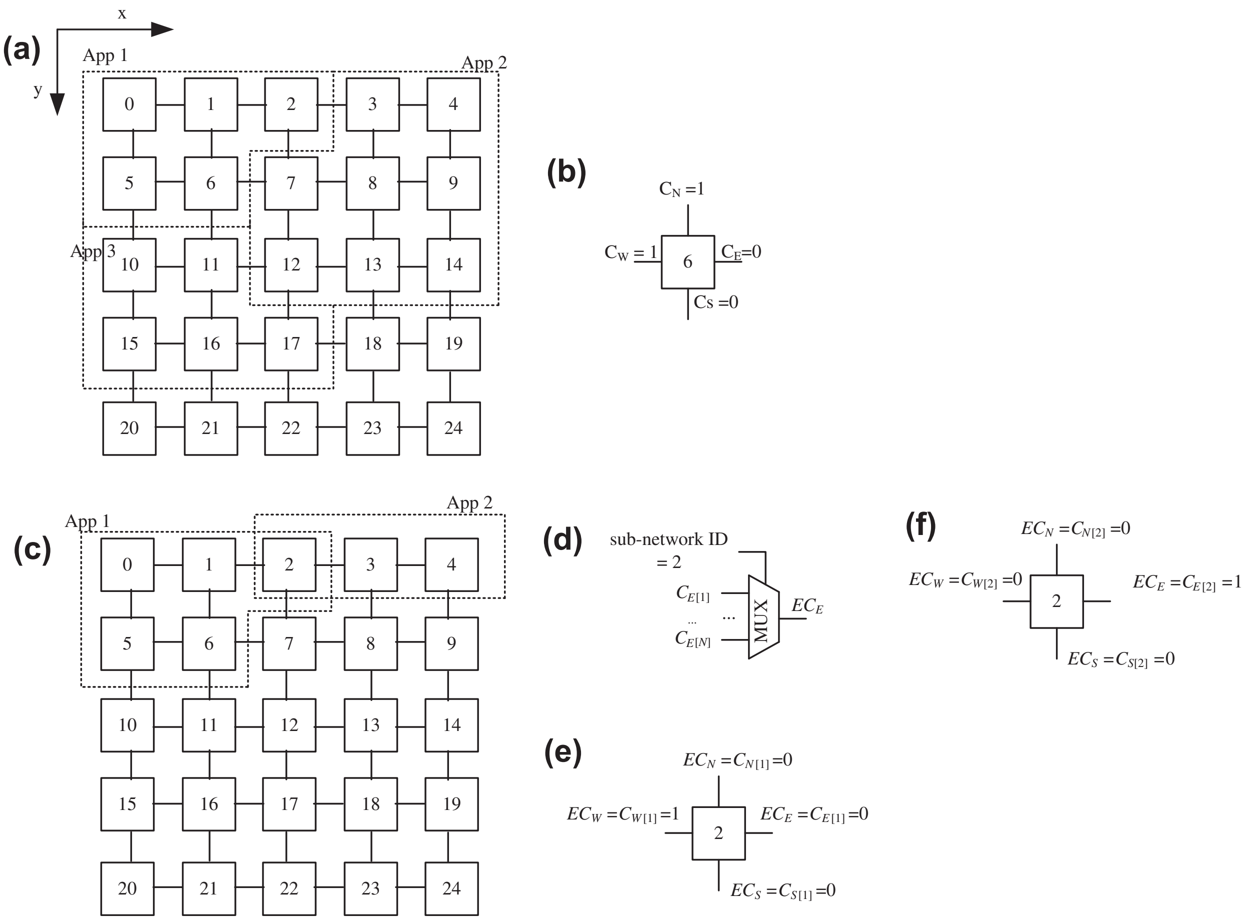


Fig. 4. (a) Sub-networks with three applications mapped on. (b) Connectivity bits of node 6. (c) Two overlapped sub-networks sharing node 2. (d) Extended connectivity bit generated using a MUX. (e) Extended connectivity bits of node 2 for sub-network 1. (f) Extended connectivity bits of node 2 for sub-network 2.

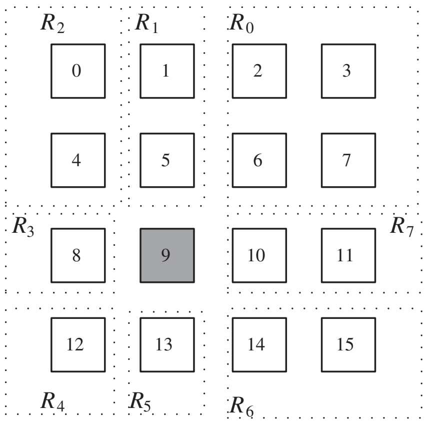


Fig. 5. Eight regions at the shaded node.

coordinates (x_6, y_6) where $x_6 > x$ and $y_6 > y$. R_7 is set of tiles with coordinates (x_7, y), where $y_7 > y$. Some nodes at the boundary may have regions without nodes. For example, R_0, R_1, and R_2 of node 0 in Fig. 5 are empty sets.

Fig. 5 shows a partition of the shaded tile.

4. Irregular sub-network oriented multicast routing

4.1. Motivation example and irregular sub-network oriented multicasting strategy

Before the proposed algorithms are described in detail, an example is given to explain the motivation. Fig. 6 shows an irregular sub-network composed of five nodes. A multicast packet is sent from the source node to two destination nodes. The dashed line represents the path if RPM [6] is used. However, since the sub-network is irregular, the dashed path cannot reach the destinations, i.e., at node 4, the packet cannot go West as the link to West is not available in this sub-network.

Alternatively, if the packet can go North at node 4 as indicated by the solid arrow in Fig. 6, the packet can arrive at both destinations following RPM from node 2.

This example shows that, if the output port found by the multicast routing algorithm is not available in the sub-network, the packet can take an alternative output port which is also on the minimal path to the destination. Each router can check the connectivity bits (defined in Section 3.2) to see whether an output port is available. The irregular sub-network oriented multicast strategy is thus derived below.

- Find the output directions to all the destinations in the destination set using a multicast routing algorithm designed for regular mesh topology.
- For each output direction, check the corresponding connectivity bit. If it is set, then the packet will be replicated and sent to the output direction; otherwise, use an alternative output direction.

Note that, following the strategy, an irregular sub-network oriented multicast routing algorithm can be developed based on any regular mesh oriented multicast routing algorithm. Due to the superiority of RPM over other algorithms (as reviewed in Section 2), we develop Alternative RPM as described below.

4.2. Hardware-based multicast routing algorithm for irregular sub-networks

To support multicast in hardware, a multicast routing logic module (MRLM) is designed. The MRLM is composed of two sub-modules.

(1) *Destination inclusion (DI) sub-module*. For a multicast packet, this sub-module checks the regions that the destination nodes belong to. To perform the destination inclusion function, bit masks are used based on the bit string method [13]. Each input port has the following three types of bit vectors.
- Input destination bit vector D. To encode all nodes in the network, an N-bit vector will be used. The ith bit of the vector is 1 if the ith node is inside the destination set.
- Bit mask vector for each region. There are eight bit masks, $BM\_R_0, \ldots, BM\_R_7$, each with N bits. The mth bit of BM_R_i is 1 if node m is in region R_i of the current node. BM_R_i's ($i = 0\ldots7$) are set offline.
- Destination bit vector for each region. Eight bit vectors $IN\_R_0, \ldots, IN\_R_7$, each with N bits, are used to represent the destinations within each region. For example, IN_R_i is obtained by ANDing D with BM_R_i ($i = 0\ldots7$).

Fig. 7 shows an example with the source node 4 and two destinations 0 and 2. The three bit vectors at node 4 are shown on the right.

(2) *Multicast routing (MR) sub-module*. This sub-module determines the output directions of the multicast packet.

The multicast routing sub-module is designed based on the irregular sub-network oriented multicasting routing algorithm AL + RPM. The basic idea of AL + RPM is to find the output directions using RPM and then decide whether to replicate the packet to the original output directions or the alternative (AL) output directions which are orthogonal to the original ones based on the corresponding connectivity bits. In the following, the details of AL + RPM will be described.

Note that, if destinations are in regions 1, 3, 5, 7, i.e., the X+, $X-$, Y+, $Y-$ regions, there is no alternative output direction. The reason is that, according to Assumption 1, the sub-network must be near

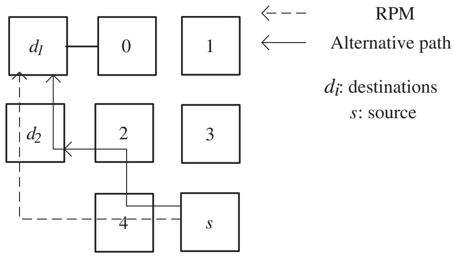
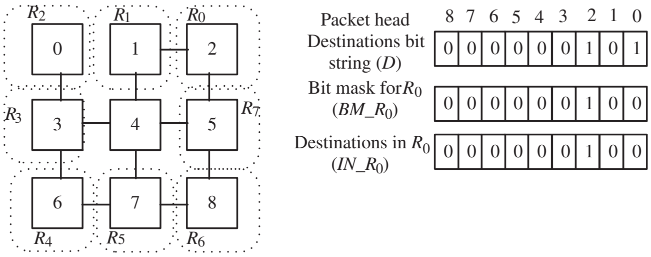


Fig. 6. Path generated by RPM and alterative path in a irregular sub-network.



Packet head	8	7	6	5	4	3	2	1	0
Destinations bit string (D)	0	0	0	0	0	0	1	0	1
Bit mask forR_0 (BM_R_0)	0	0	0	0	0	0	1	0	0
Destinations in R_0 (IN_R_0)	0	0	0	0	0	0	1	0	0

Fig. 7. Bit vectors at node 4 assuming the destinations are 0 and 2.

convex which ensures that there exists at least a minimal path inside a sub-network for any pair of nodes. It is clear that for each destination in regions 1, 3, 5, 7, there is only one minimal path to that destination. Hence, there is no alternative output direction for destinations in those regions. Thus, only the alterative output directions for regions 0, 2, 4, 6 are found.

Fig. 8 lists the AL + RPM algorithm. As in RPM, replication is made as late as possible to reduce the number of replicated packets. As shown in Fig. 8, if there are destinations in both R_0 and R_2, instead of replicating packets to West and East, the packet will be forwarded to North first and replicated later. Similar treatment is applied to destinations belonging to other similar region combinations. By this way, the resulted total number of hop counts for a multicast communication is minimized.

Fig. 9 shows an example of finding the routing path using AL + RPM for a multicast communication from s to destinations d_1, d_2, and d_3. In step 1, node s replicates two packets to nodes 3 and 6 as its extended connectivity bits EC_E and EC_N are 1. In step 2, instead of replicating packets to East and West, node 3 forwards the packet to node 1. Node 6 forwards the packet to node 7 as its EC_E is 0 and the alternative output for destination d_3 is South. In step 3, node 1 replicates two packets to nodes d_1 and d_2. Node 7 also forwards the packet to d_3. In the last step, nodes d_1, d_2 and d_3 consume the packets as they are the destinations.

The above AL + RPM is not deadlock free. In order to avoid deadlocks, virtual channels are used. As stated in [13] and [6], two virtual networks can be used to avoid deadlocks for mesh-based networks. For AL + RPM, two virtual networks are used, VN_0 and VN_1. VN_0 does not allow packets to turn to NORTH while VN_1 does not allow packets to turn to SOUTH. The virtual network to be used is decided for each packet at the source router and cannot be changed at the intermediate routers.

4.3. Hardware cost

To estimate the area overhead of AL + RPM, the routers for 8 × 8 network are synthesized using the Synopses Physical Compiler with TSMC 90 nm library. For 8 × 8 network, 64 bits are needed for representing the destinations. Up to eight overlapped sub-networks are supported, thus, eight connectivity bits are used for each direction. Fig. 10a shows the structure of the routing unit which includes three sub-modules, the DI sub-module, MR sub-module (AL + RPM), and the EC bits generation sub-module. Fig. 10b shows the circuit of the MR sub-module.

Fig. 11 shows the power, area cost, and delay comparison of the baseline router and the router implementing AL + RPM. The baseline router supports the multicast *XY* routing algorithm [19] which uses the three types of bit vectors for four directions. As shown in the table, the area overhead of AL + RPM over the baseline router is 5.5%. It is estimated that the percentage of the overhead tends to be stable when the network size grows.

```
//Alternative RPM for irregular sub-networks
//Temporary bit vectors: N_DestSet, E_DestSet, W_DestSet S_DestSet are initially set to be zero.
if ($IN\_R_0$ ) begin    // if $IN\_R_0$ is non-zero, i.e., there is destination in $R_0$
   if ($EC_N$ == 1) N_DestSet= $IN\_R_0$  OR N_DestSet
   else E_DestSet= $IN\_R_0$  OR E_DestSet
end
if ($IN\_R_1$) N_DestSet= $IN\_R_1$  OR N_DestSet

if ($IN\_R_2$) begin
   if ( !$IN\_R_3$ &&( $IN\_R_1$|| $IN\_R_0$)) begin // if there are destinations in $R_2$ and $R_1$ or $R_2$ and $R_0$, go north first, replicate later
      if ($EC_N$ == 1) N_DestSet= $IN\_R_2$  OR N_DestSet
      else W_DestSet= $IN\_R_2$  OR W_DestSet
   end
   else begin
      if ($EC_W$ == 1) W_DestSet= $IN\_R_2$  OR W_DestSet
      else N_DestSet= $IN\_R_2$  OR N_DestSet
    end
end

if ($IN\_R_3$) W_DestSet= $IN\_R_3$ OR W_DestSet

if ($IN\_R_4$) begin
   if ( ! $IN\_R_5$ && $IN\_R_3$) begin // if there are destinations in $R_4$ and $R_3$, go west first, replicate later
      if ($EC_W$ == 1) W_DestSet= $IN\_R_4$ OR W_DestSet
     else S_DestSet= $IN\_R_4$  OR S_DestSet
  end
   else begin
      if( $EC_S$ == 1) S_DestSet= $IN\_R_4$  OR S_DestSet
      else W_DestSet= $IN\_R_4$ OR W_DestSet
   end
end

if ($IN\_R_5$) S_DestSet= $IN\_R_5$  OR S_DestSet

if ($IN\_R_6$) begin
   if( ! $IN\_R_7$ && ($IN\_R_4$ || $IN\_R_5$) ) begin  // if there are destinations in $R_6$ and $R_4$ or $R_6$ and $R_5$, go south first, replicate later
      if ($EC_S$ == 1) S_DestSet= $IN\_R_6$  OR S_DestSet
     else E_DestSet= $IN\_R_6$  OR E_DestSet
   end
   else begin
      if ($EC_E$ == 1) E_DestSet= $IN\_R_6$  OR E_DestSet
      else S_DestSet= $IN\_R_6$  OR S_DestSet
   end
end

if ($IN\_R_7$) E_DestSet= $IN\_R_7$  OR E_DestSet
```

Fig. 8. Pseudocode of AL + RPM multicast routing algorithm.

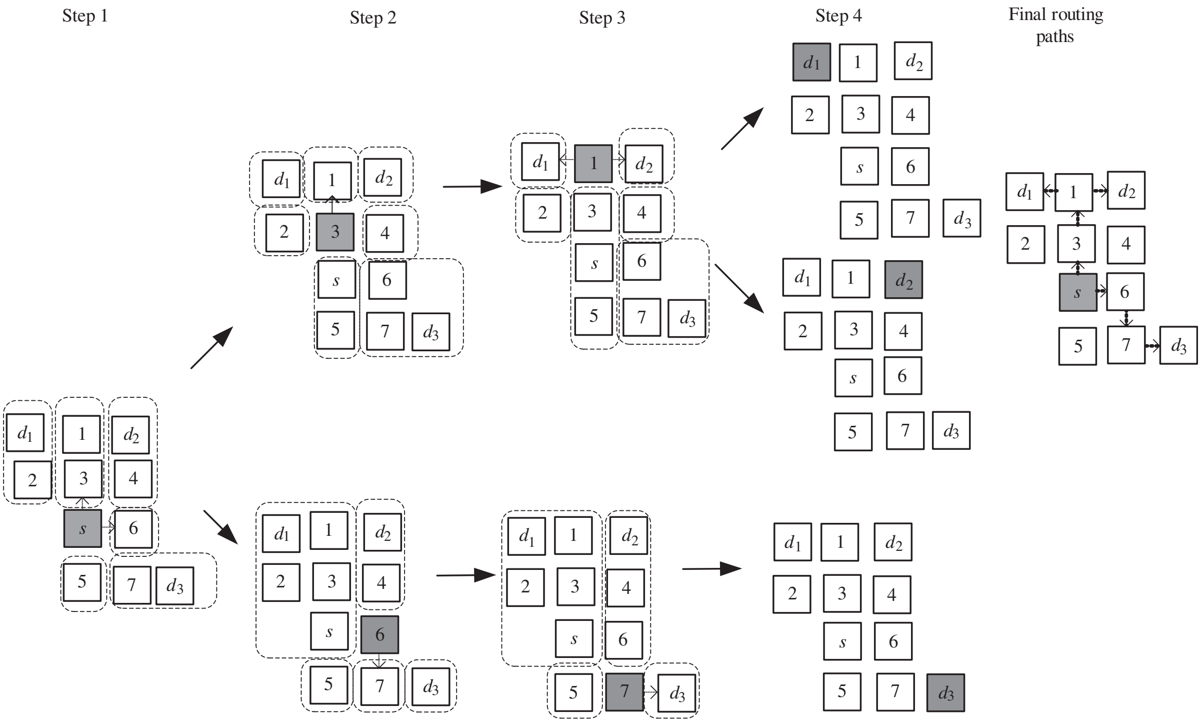


Fig. 9. An example showing the routing steps of AL + RPM.

5. Performance evaluation

5.1. Experiment settings

To evaluate the performance of the AL + RPM multicast routing algorithm, AL + RPM is simulated under traces from real applications and random traffic. The performance of AL + RPM in terms of power consumption (as defined in Section 3.1) and network latency is compared against bLBDR and multiple unicast. These multicast algorithms are implemented on the cycle accurate simulator Noxim [25]. The power parameters are based on the synthesis results using Synopses Physical compiler with TSMC 90 nm library.

The RSIM [23] full system simulator running SPLASH benchmark set [24] is used to obtain the traffic traces of real applications. For random traffic, the unicast traffic has uniform distribution of the destinations, and the multicast traffic is generated randomly with the average destination set size 8.

Note that the multiple unicast (shown as multiple UC in all figures) routing is modified here to support irregular sub-networks. The basic algorithm is based on *XY* routing. If the default output direction is not available, the output direction on the other dimension is chosen.

5.2. Real applications

The communication traces of benchmarks brnes, moldyn, radix, raytrace, tomcatv, and ocean from SPLASH are extracted for our evaluation using RSIM. Based on the traffic analysis of the trace files, five synthetic traffics are generated which have the same injection rate, destination addresses and multicast to unicast ratio as the original trace files of the five applications. These five applications are mapped to the sub-networks as in Fig. 12. Each tile implements a packet generator and a packet receiver.

Fig. 13 shows the power consumption and packet latency results of AL + RPM, and bLBDR under multi-application benchmark. The power consumption is normalized over the minimum power consumption of the two algorithms. The average packet latency is obtained by averaging the network latency of all received packets. The power consumption of bLBDR is about 30% higher than that of AL + RPM.

5.3. Random benchmarks with uniform traffic

Five random benchmarks are generated and mapped on 8 × 8 mesh-based CMP as in Fig. 14. The average number of destinations is set to 8. Figs. 15 and 16 show the results of AL + RPM, bLBDR, and multicast UC with the traffic ratio (MUR) of multicast (MC) to unicast (UC) traffic ranging in {0.05:1, 0.2:1, 0.25:1, 0.3:1}. MUR ranging from 0.2:1 to 0.3:1 reflects real application traffic scale. MUR of 0.05:1 is used to evaluate the efficiency of AL + RPM with multiple unicast.

As shown Fig. 15a, when MUR is small, AL + RPM and bLBDR both have lower power consumption than multiple unicast. For larger MUR, AL + RPM outperforms broadcast-based bLBDR and multiple unicast significantly (Fig. 15b–d). As the injection rate increases, the power consumption of Multiple unicast and bLBDR increases much faster than that of AL + RPM. Multiple unicast has the largest power consumption. For instance, when MUR is 0.3:1 (Fig. 15d), AL + RPM saves 29% power consumption than that of Multiple unicast. On average (averaging over Fig. 15), AL + RPM saves 20% power consumption than that of Multiple unicast. The reason is simply due to the fact that AL + RPM saves a large number of replicated packets compared with multicast unicast, which produces the number of replicated packets as much as the number of destinations.

AL + RPM also achieve lower power consumption than bLBDR. For example, when MUR is 0.3:1, AL + RPM saves 18% power consumption than that of bLBDR. On average, AL + RPM saves 14% power consumption than that of bLBDR. The reason is that compared with the broadcast-based bLBDR, AL + RPM saves the

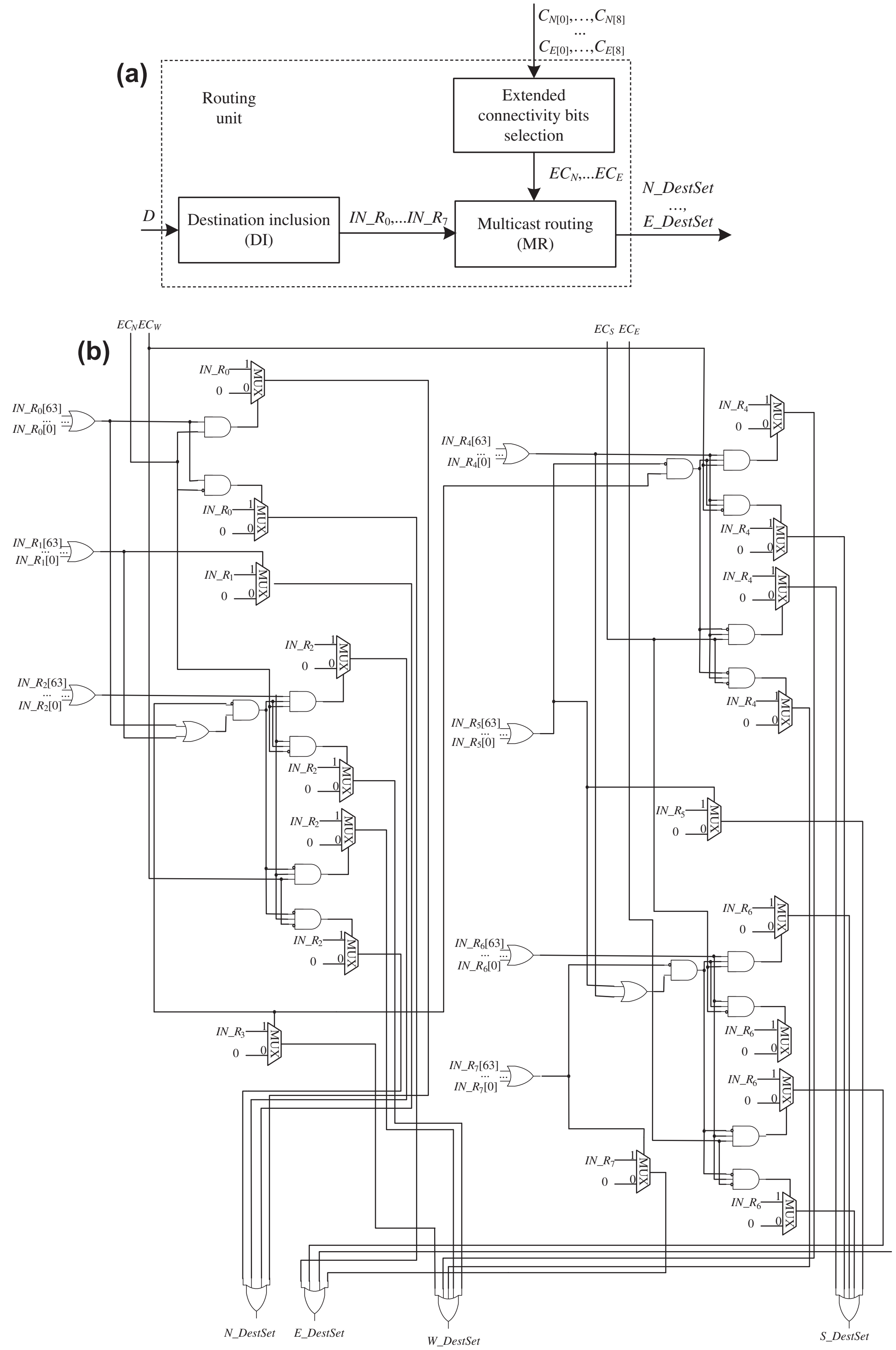


Fig. 10. (a) Structure of the routing unit. (b) The circuit of MR sub-module, assuming the network size is 8 × 8.

number of replicated packets significantly, which lowers the power consumption.

In terms of average packet latency, when MUR is low (Fig. 16a), the three multicast algorithms do not show much difference when

	AL+RPM	Baseline
Power (mW)	8.84	8.68
Area (um^2)	215447	204246
Delay (nm)	2.43	2.40

Fig. 11. Power, area cost and delay comparison of baseline router and the router implementing AL + RPM.

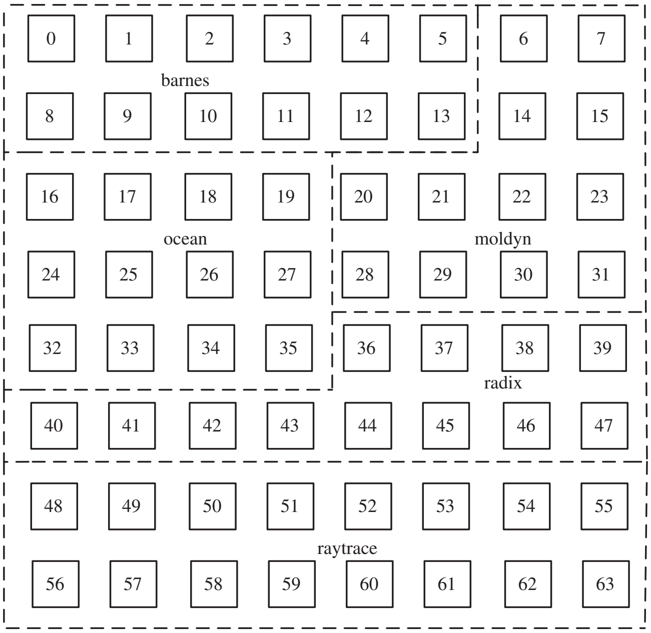


Fig. 12. The multi-application benchmark mapping on an 8 × 8 mesh-based CMP.

injection rate is low. However, when the injection rate increases, the average packet latency of multiple unicast increases dramatically. When MUR is 0.2:1 and 0.25:1 (Fig. 16b and c), the latency of AL + RPM increases much slower than that of bLBDR and Multiple unicast. The reason is that less packets are replicated using AL + RPM, thus, the network is less congested than using bLBDR and Multiple unicast. The difference becomes more distinct when MUR is larger. When MUR is 0.3:1 and injection rate is high (e.g., near 0.15), the latency of AL + RPM is only 50% or less than that of bLBDR.

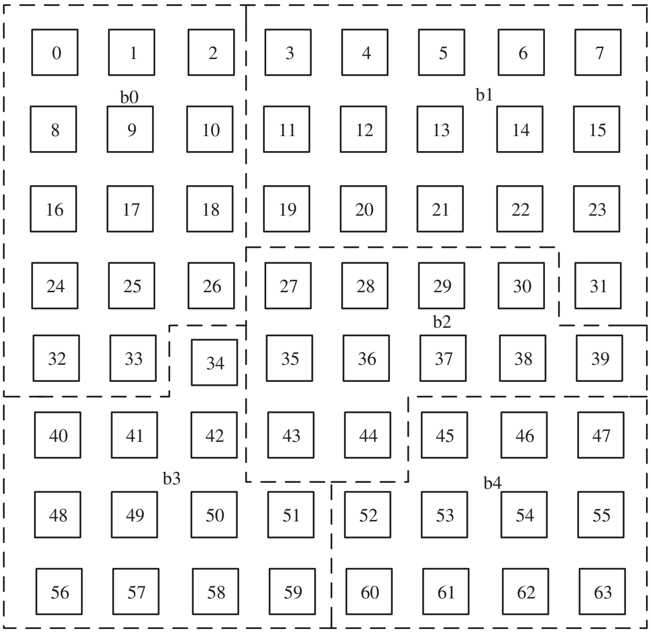


Fig. 14. The random benchmark mapping on 8 × 8 mesh-based CMP.

The experimental results confirm that AL + RPM achieve significant improvement than multiple unicast and broadcast-based approach in both power consumption and network performance. When MUR is high and traffic is heavy, AL + RPM is even superior.

6. Conclusion

In this paper, an irregular sub-network oriented multicast routing strategy was proposed. The basic idea of this routing strategy is that, if the output channel found by regular topology oriented multicast routing is not available, choose an alternative output channel which also leads to the minimal path to the destination. As a matter of fact, following this strategy, an irregular topology oriented multicast routing algorithm can be designed based on any regular mesh based multicast routing algorithm. One such algorithm, AL + RPM is proposed to support multicasting for irregular subnetworks based on RPM which only supports multicasting for regular mesh topology. Experimental results of the algorithm under traces from real benchmarks and synthetic benchmarks confirm that AL + RPM significantly improve the power consumption and network performance compared with multiple unicast and broadcast-based bLBDR, the AL + RPM multicast routing approach supporting irregular sub-networks. The area overhead of AL + RPM has shown to be quite modest.

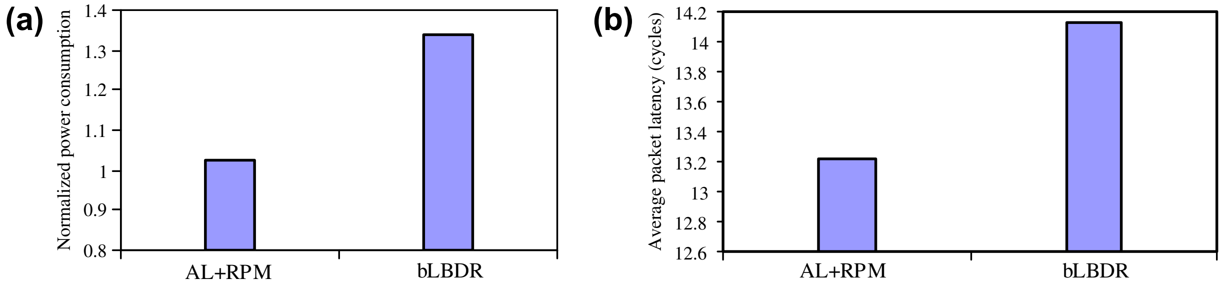


Fig. 13. Normalized power consumption (a) and latency, and (b) of AL + RPM and bLBDR.

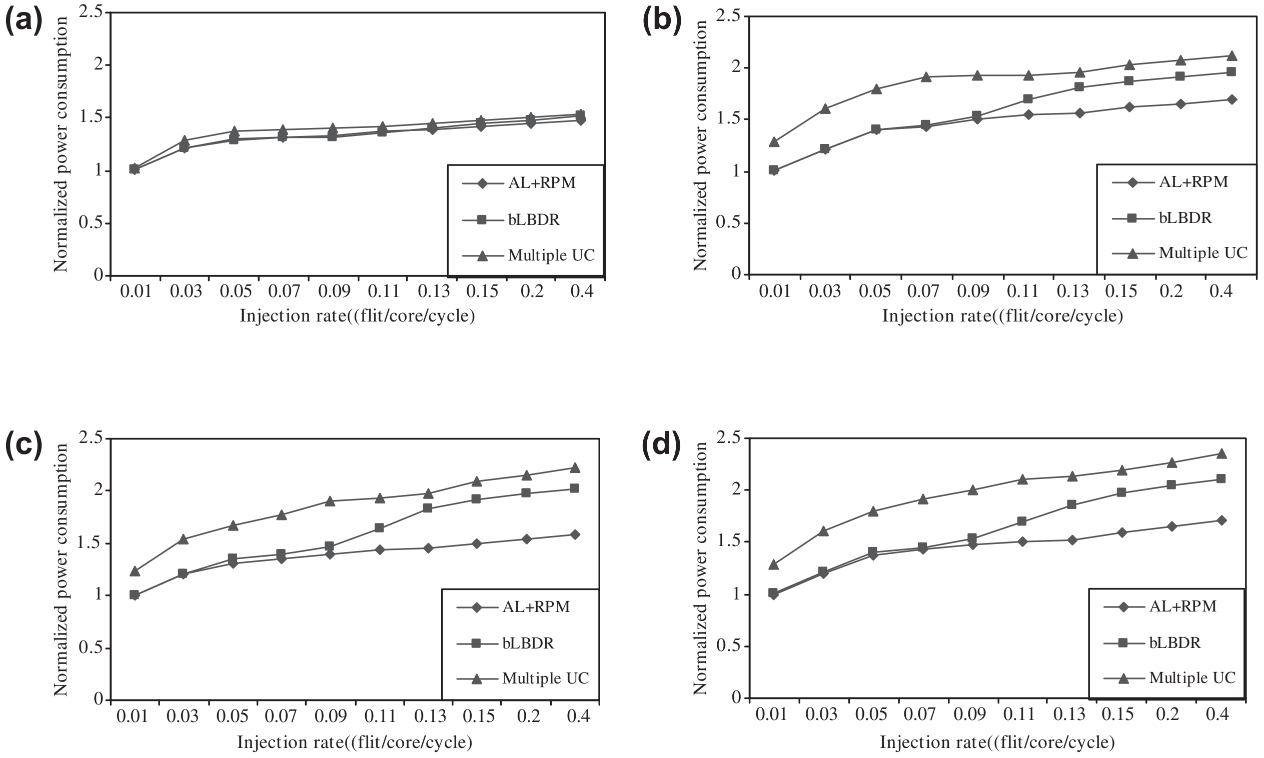


Fig. 15. Normalized power consumption results with MUR set to (a) 0.05:1, (b) 0.2:1, (c) 0.25:1 and (d) 0.3:1.

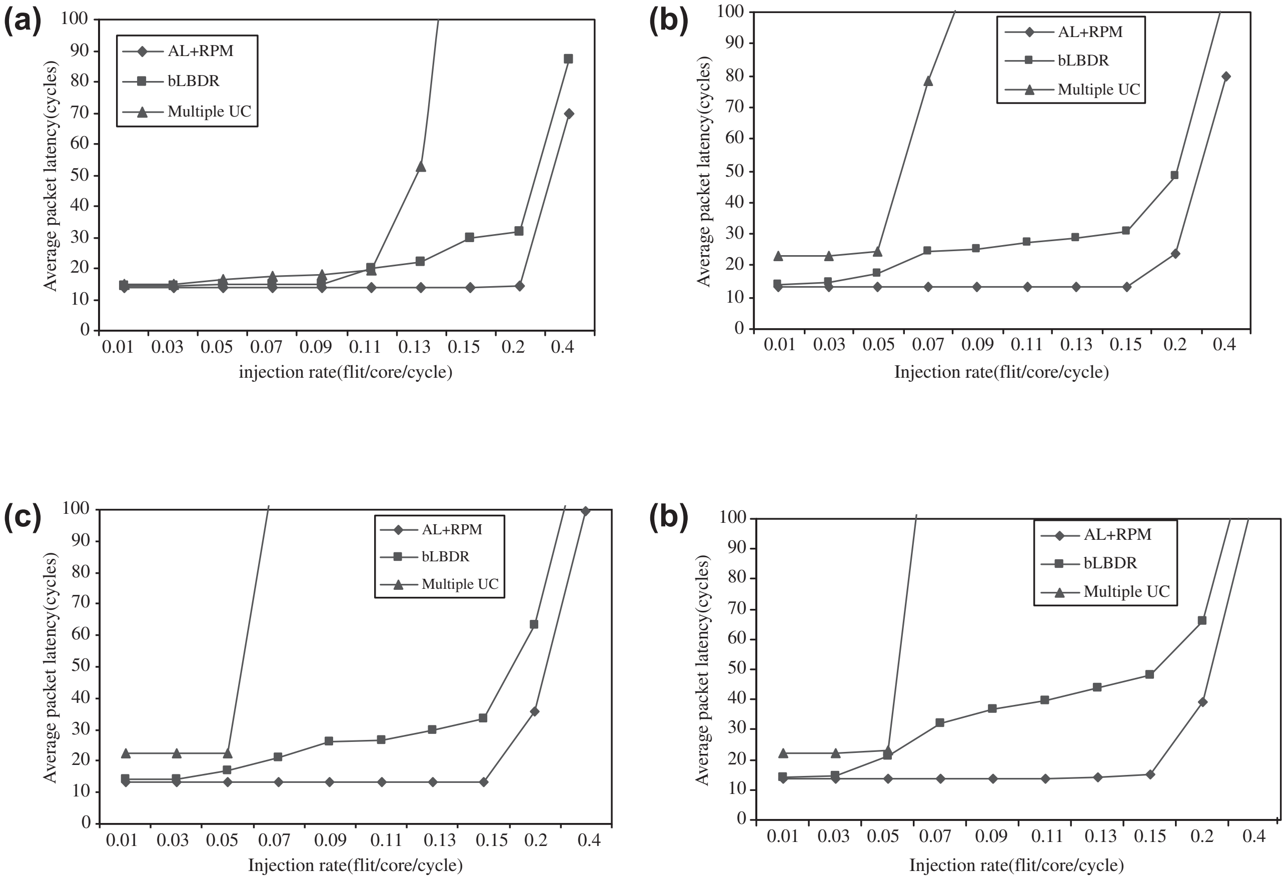


Fig. 16. Latency results with MUR set to (a) 0.05:1, (b) 0.2:1, (c) 0.25:1 and (d) 0.3:1.

Acknowledgement

This work has been supported by NSF under grant no. ECCS-0702168 and National Natural Science Foundation of China under grant no. 60873112.

References

[1] S. Borkar, Thousand core chips: a technology perspective, in: Proc. 44th Design Automation Conf., ACM, 2007, pp. 746–749.
[2] J.L. Manferdelli, N.K. Govindaraju, C. Crall, Challenges and opportunities in many-core computing, Proc IEEE 96 (2008) 808.
[3] J. Held, J. Bautista, S. Koehl, From a few cores to many: a tera-scale computing research review, Intel Research White Paper, 2006.
[4] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, S. Borkar, A 5-GHz mesh interconnect for a teraflops processor, IEEE Micro 27 (2007) 51–61.
[5] D. Wentzlaff, P. Griffin, H. Hoffmann, L. Bao, B. Edwards, C. Ramey, M. Mattina, C.C. Miao, J.F. Brown, A. Agarwal, On-chip interconnection architecture of the tile processor, IEEE Micro 27 (2007) 15–31.
[6] L. Wang, Y. Jin, H. Kim, E.J. Kim, Recursive partitioning multicast: a bandwidth-efficient routing for on-chip, in: Proc. 3rd ACM/IEEE Int'l Symp. Networks-on-Chip, 2009.
[7] N.E. Jerger, L.S. Peh, M. Lipasti, Virtual circuit tree multicasting: a case for on-chip hardware multicast support, in: Proc. 35th Int'l Symp. Computer Architecture, 2008, pp. 229–240.
[8] S. Rodrigo, J. Flich, J. Duato, Efficient unicast and multicast support for CMPs, in: Proc. 41st IEEE/ACM Int'l Symp. Microarchitecture, 2008, pp. 364–375.
[9] A. Gavrilovska, S. Kumar, H. Raj, K. Schwan, V. Gupta, R. Nathuji, R. Niranjan, A. Ranadive, P. Saraiya, High-performance hypervisor architectures: virtualization in hpc systems, in: 1st Workshop on System-level Virtualization for High Performance Computing, 2007.
[10] S. Murali, Designing Reliable and Efficient Networks on Chips, Springer Verlag, 2009.
[11] M.B. Taylor, J. Kim, J. Miller, D. Wentzlaff, F. Ghodrat, B. Greenwald, H. Hoffman, P. Johnson, J.W. Lee, W. Lee, The Raw microprocessor: a computational fabric for software circuits and general-purpose programs, IEEE Micro 111 (2002) 25–35.
[12] C.L. Chou, R. Marculescu, User-aware dynamic task allocation in networks-on-chip, in: Proc. Conf. Design, Automation and Test in Europe, ACM, 2008, pp. 1232–1237.
[13] J. Duato, S. Yalamanchili, L. Ni, Interconnection Networks an Engineering Approach, Morgan Kaufmann, 2002.
[14] M. Palesi, R. Holsmark, S. Kumar, V. Catania, Application specific routing algorithms for networks on chip, IEEE Trans. Parallel Distribut. Syst. 20 (2009) 316–330.
[15] Z. Lu, B. Yin, A. Jantsch, Connection-oriented multicasting in wormhole-switched networks on chip, in: Proc. Emerging VLSI Technologies and Architectures, 2006.
[16] I.V. Senin, L. Mhamdi, K. Goossens, Efficient multicast support in buffered crossbars using networks on chip, in: Proc. Global Telecommunications Conference, 2009.
[17] P. Abad, V. Puente, J.A. Gregorio, MRR: enabling fully adaptive multicast routing for CMP interconnection networks, in: Proc. 15th Int'l Conf High-Performance Computer Architecture, 2009, pp. 355–366.
[18] F.A. Samman, T. Hollstein, M. Glesner, Planar adaptive router microarchitecture for tree-based multicast network-on-chip, in: Proc. Int'l Workshop on Network on Chip Architectures, 2008, pp. 6.
[19] C. Xiang, Design and verification of Network on chip components, Dept. Information Science and Electronic Engineering Zhejiang University, Hangzhou, 2008 (in Chinese).
[20] V. Varavithya, P. Mohapatra, Asynchronous tree-based multicasting in wormhole-switched MINs, IEEE Trans Parallel Distribut. Syst. 10 (1999) 1159–1178.
[21] J. Hu, R. Marculescu, Energy-and performance-aware mapping for regular NoC architectures, IEEE Trans. Comput. Aid. Des. Int. Circ. Syst. 24 (2005) 551–562.
[22] C.L. Chou, U.Y. Ogras, R. Marculescu, Energy-and performance-aware incremental mapping for networks on chip with multiple voltage levels, IEEE Trans. Comput. Aid. Des. Int. Circ. Syst. 27 (2008) 1866–1879.
[23] R. Fernandez, J.M. Garc a, RSIM x86: a cost-effective performance simulator, in: Proc. High Performance Computing & Simulation Conf., 2005, pp. 774–779.
[24] J.P. Singh, W.D. Weber, A. Gupta, SPLASH: stanford parallel applications for shared-memory. ACM SIGARCH Computer Architecture News, 1992.
[25] Noxim. <http://sourceforge.net/projects/noxim>.

Xiaohang Wang received the B.Eng. degree in communication and electronic engineering from Zhejiang University, China, in 2006. He is currently pursing the Ph.D. degree in communication and electronic engineering at Zhejiang University, China. His research interests include compiler, parallel programming models, core-based digital SoC and NoC design and test.

Dr. Mei Yang received her Ph.D. in Computer Science from the University of Texas at Dallas in Aug. 2003. She has been an assistant professor in the Department of Electrical and Computer Engineering, University of Nevada, Las Vegas since Aug. 2004. Her research interests include computer architectures, networking, and embedded systems.

Dr. Yingtao Jiang received his Ph.D. in Computer Science from the University of Texas at Dallas in Aug. 2001. He joined the Department of Electrical and Computer Engineering, University of Nevada, Las Vegas in Aug. 2001. He has been an associate professor since Aug. 2007. His research interests include algorithms, computer architectures, VLSI, networking, nano technologies, etc.

Peng Liu received the B. Eng. and M. Eng. degrees in optical engineering from Zhejiang University, in 1992, and 1996, respectively, and the Ph.D. degree in communication and electronic engineering from Zhejiang University, China, in 1999. He has been an Associate Professor with the Information Science and Electronic Engineering Department, Zhejiang University, Hangzhou, China, since 2002. His research focuses embedded processor, multiprocessor systems-on-chip architectures, on-chip interconnection networks, real-time operating system, compiler, and circuits for communications.