RUNoC: Re-inject into the Underground Network to Alleviate Congestion in Large-Scale NoC

Xinghao Zhu[a], Jiyuan Bai[a], Zifeng Zhao[a], Qirong Yu[a],
Xiaofang Zhou[a], Gengsheng Chen[ab]

[a] *School of Microelectronics, Fudan University, Shanghai, China* [b] *Jiashan Fudan Institute, Jiaxing, Zhejiang Province, China*

2025.1.22

Outline

Typical Network-on-Chip (NoC)

Strong Demand for NoC

Network-on-Chip (NoC) gains considerable attention for its remarkable advantages in **scalability** and **high bandwidth**.

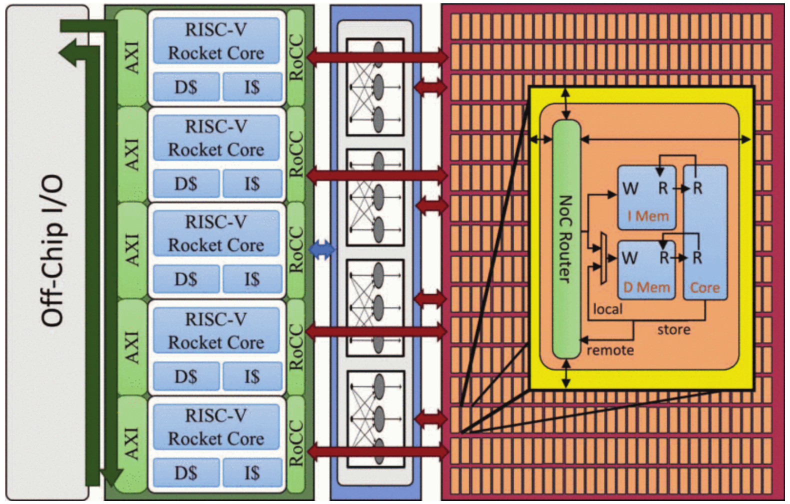


Fig1. Celerity block diagram[1].

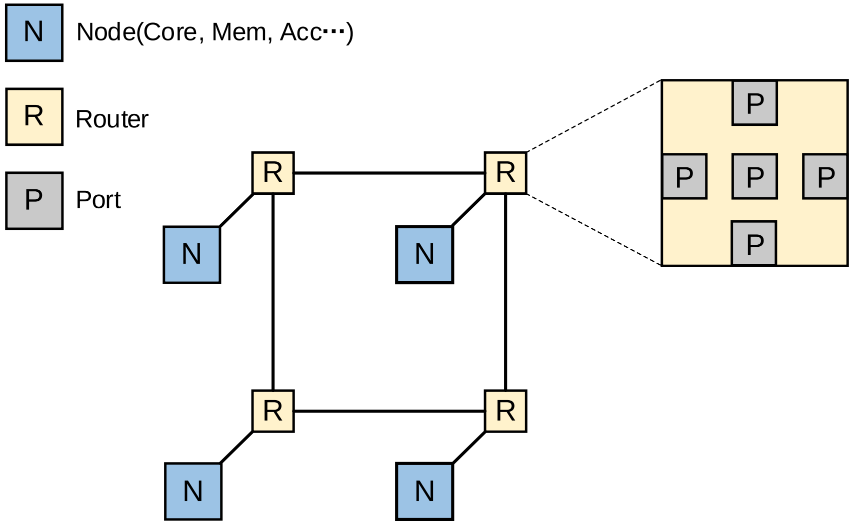


Fig2. Typical NoC architecture.

[1]S. Davidson et al., "The Celerity Open-Source 511-Core RISC-V Tiered Accelerator Fabric: Fast Architectures and Design Methodologies for Fast Chips," in IEEE Micro, vol. 38, no. 2, pp. 30-41.

Typical Network-on-Chip (NoC)

Typical Node Architecture

a) **Router**

Routing the packets to their destination.

b) **Network Interface**

RX: Receiving and analyzing the packets from network.

TX: Transferring the information into flits and send them to network.

c) **Processor**

CPU/Mem/Computing Unit…

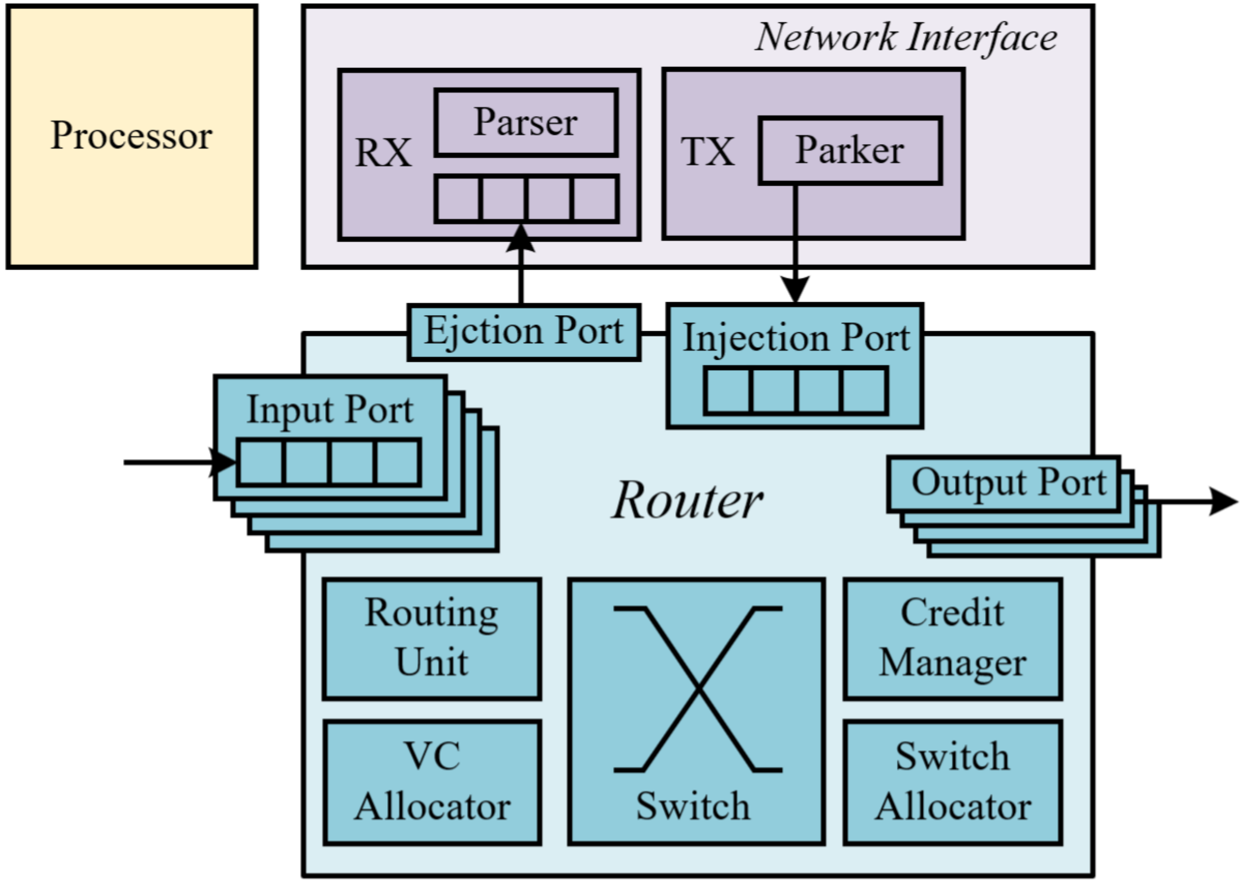


Fig. Typical node architecture.

Long Path Congestion in NoC

Long Path Congestion

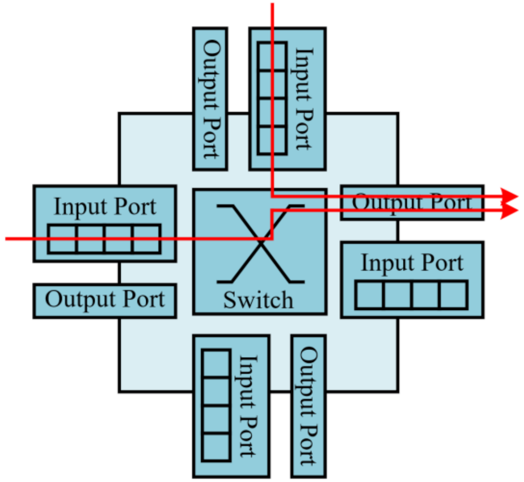


Fig1. Node congestion.

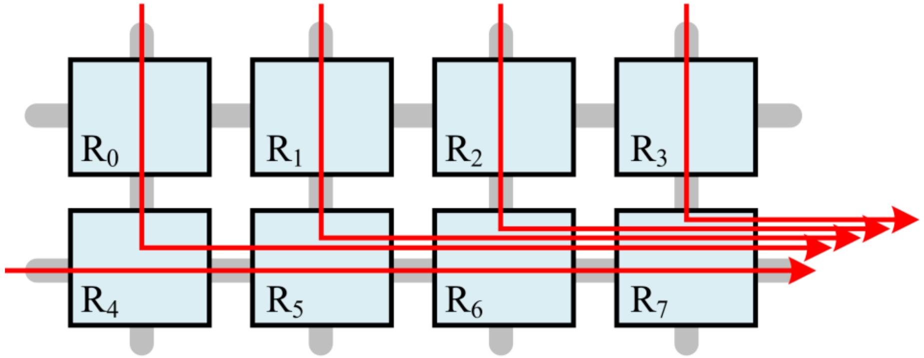


Fig2. Prolonged path congestion.

Existed Solution

a) Carefully designed routing algorithm – Self-relief mechanism

Often require extra paths and relay on global information, which results in route latency and high resource usage

Existed Solution

b) Two-Level Network (TLN) – Pseudo-3D topology

Symmetric TLN (S-TLN): two layers have equal sizes, where each node in one layer is associated with a corresponding node in the other layer.

Asymmetric TLN (A-TLN): consists of a complete network and a sparse network with fewer nodes.

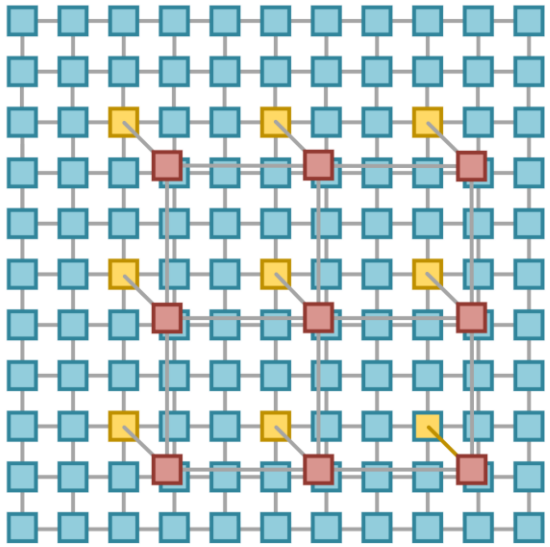


Fig. An asymmetric TLN.

Existing architectures have their own lack including power and area consumption, load balancing and deadlock.

Existed Solution

c) Partition-based NoC

In asymmetric TLN, partition a large network into several smaller subnets and use the sparse net to enable inter-subnet transmission.

Effectively reducing the hop count, thus enhancing overall system efficiency.

Not all inter-subnet transmissions achieve the same reduction in hop count. Fine-grained optimization is often overlooked, which makes it difficult to fully exploit the advantages of TLN.

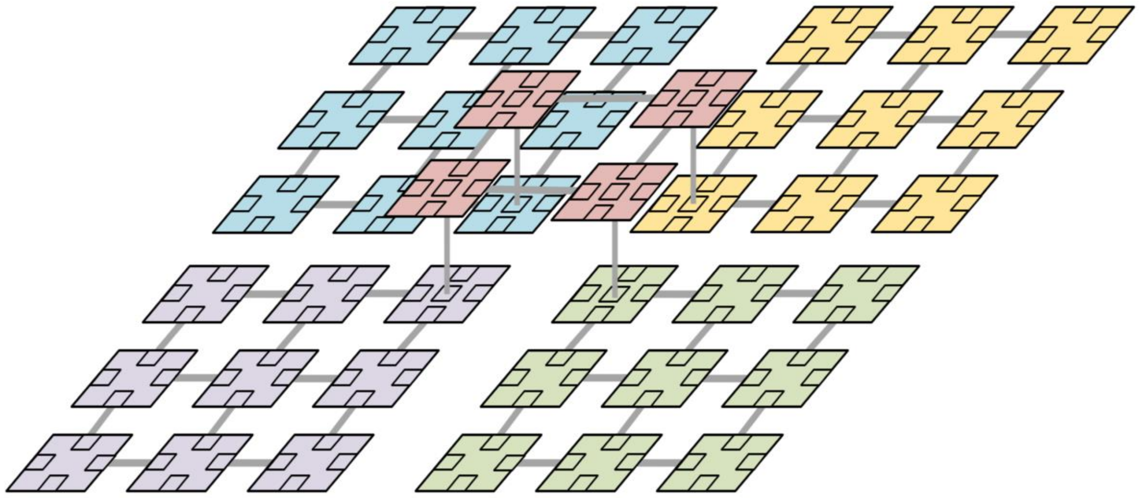


Fig. Partition-based NoC.

Motivation

Load Distribution in TLN

When the traffic pattern tends toward either long paths or short paths, there is always a network that is underutilized.

Suppose the two layers of TLN are Net_A and Net_B, the ideal ratio of the number of packets in Net_A and Net_B is

$$k = \frac{N_A \cdot AvgHops_B}{N_B \cdot AvgHops_A}$$

Hot Nodes

Routing packets to the sparse layer based solely on path length calculation can lead to severe congestion and blocking in the sparse layer and communication nodes.

Motivation

Deadlock and livelock in TLN

Deadlock occurs when dependency ring appears. XY Routing is a widely used deadlock-free routing algorithm.

Even if both layers of network use deadlock-free routing algorithm, inter-layer communication may still cause deadlocks.

A livelock occurs when packets infinitely loop between the two layers of the network without making any progress towards their ultimate destinations.

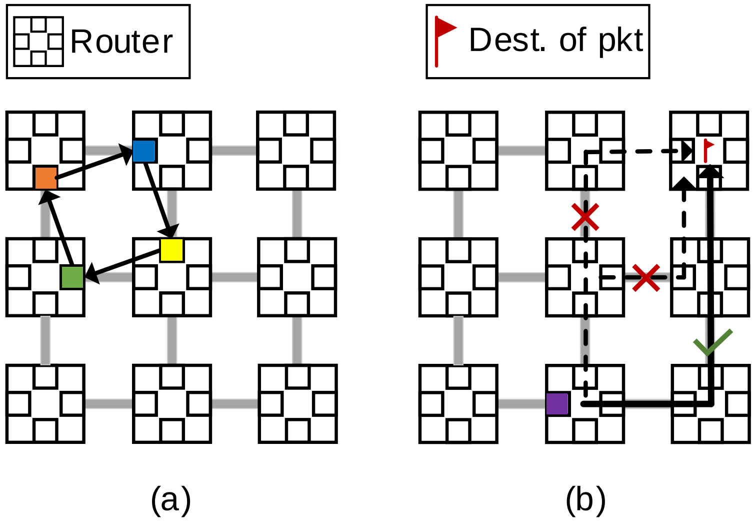


Fig1. (a) Deadlock Example. (b) Deterministic XY Routing.

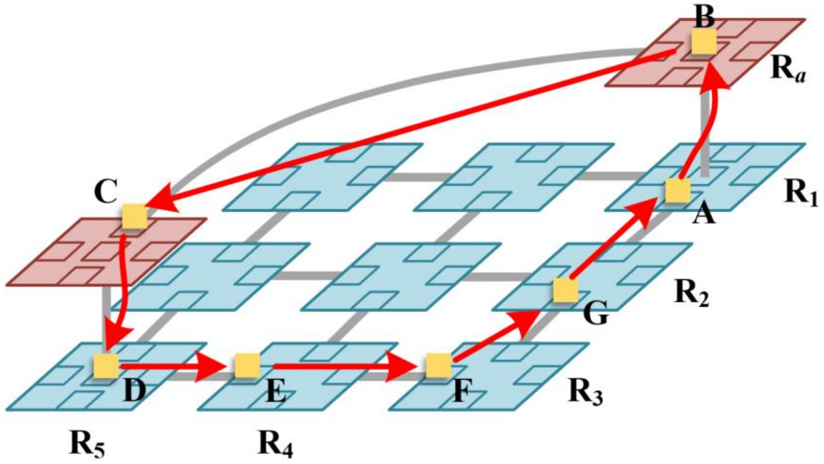


Fig2. A Deadlock in TLN.

Motivation

High-Level Idea

Main Contributions

a) We propose **RUNoC**, a novel **partition-based TLN architecture**, aimed at significantly **reducing long path latency in large-scale NoC**. It consists of a **Main Network (M-NET)** and a **sparse Underground Network (U-NET)**. M-NET can **re-inject** packets to U-NET to alleviate congestion.

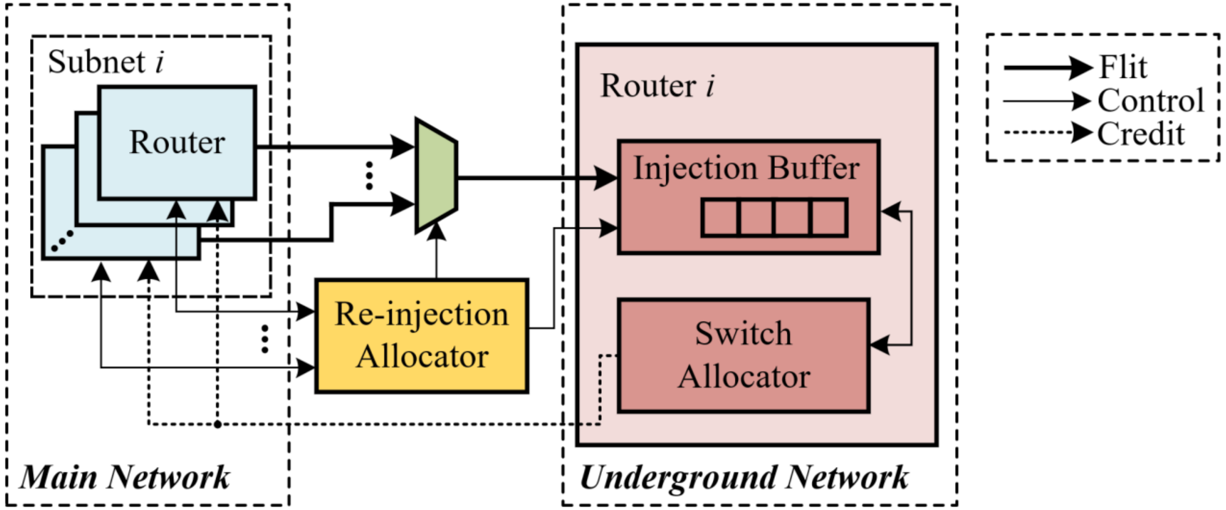


Fig. Subnet architecture.

RUNoC Architecture

High-Level Idea

Main Contributions

b) We use **Shared Row Buffers (SRBs)** and modified **Network Interface (NI)** to ensure that there is **no deadlock** or **livelock.**

c) We develop **CDRU**, which decides when to re-inject packets into U-NET based on the congestion information from M-NET and packet destination distance. CDRU can reduce latency and achieve **load balance** between two layers.

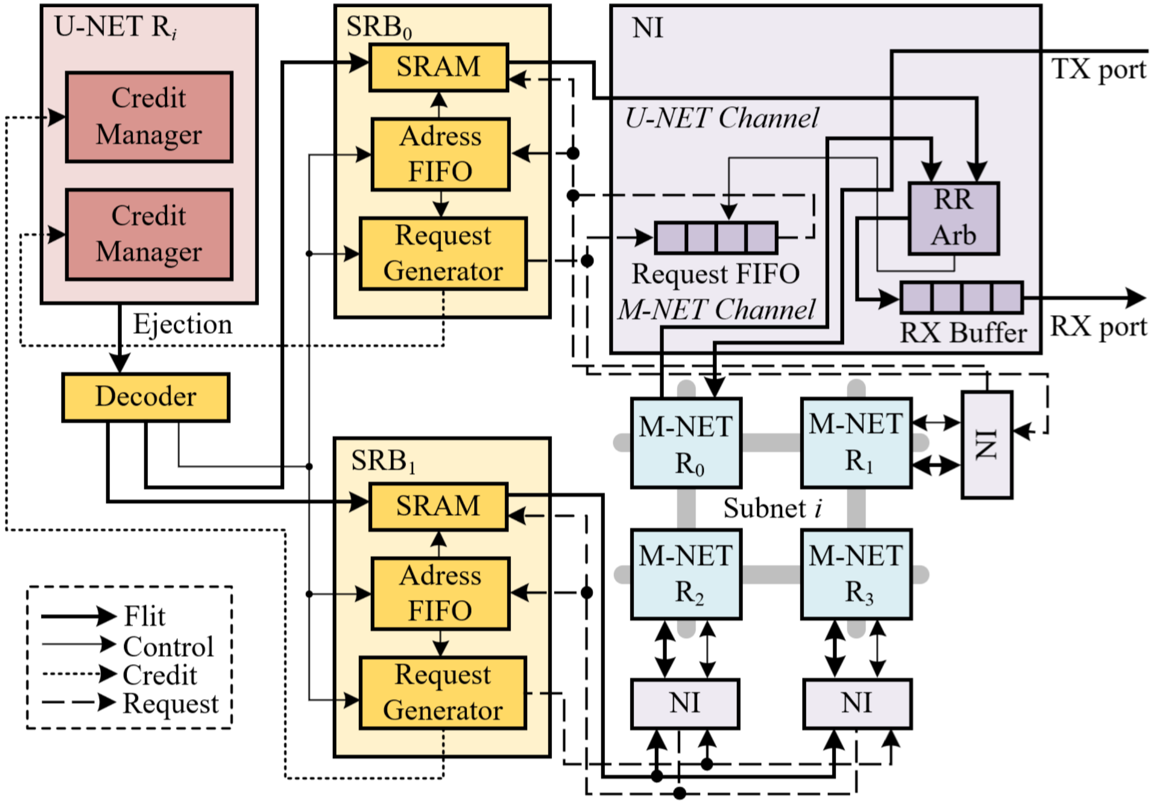


Fig. A subnet with SRBs and modified NI.

RUNoC Architecture

Inter- & Intra-Layer Communication

Inter-layer Communication

Routers in a subnet select packets to be re-injected into U-NET router. The Re-injection Allocator selects the packet to be re-injected in a Round-Robin manner

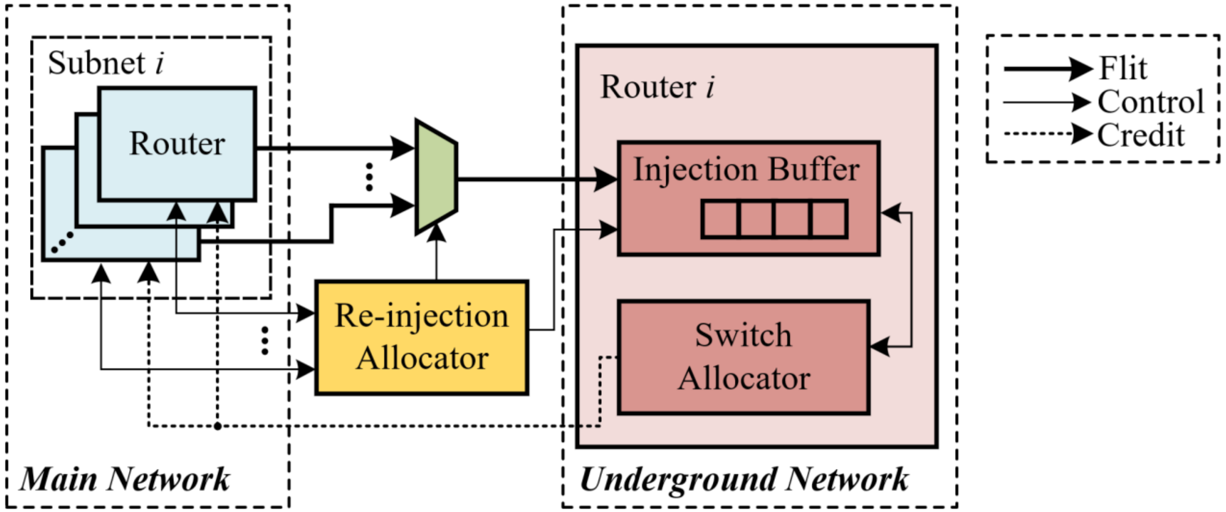


Fig. Subnet inter-layer communication architecture.

Intra-layer Communication

In M-NET, packets are routed via XY routing.

In U-NET, benefit from its sparse topology, we implement a full-connected routing with acceptable hardware complexity. A packet in any router and be routed to another router in only one hop.

Full-connected routing method significantly decreases the latency of U-NET.

Router Architecture

M-NET Router

Routing Unit: CDRU

- Routing packets to their destination normally.
- Deciding whether to re-inject the packet to U-NET based on congestion information and the distance to the packet’s destination.

Enable the sixth direction for output – U-direction

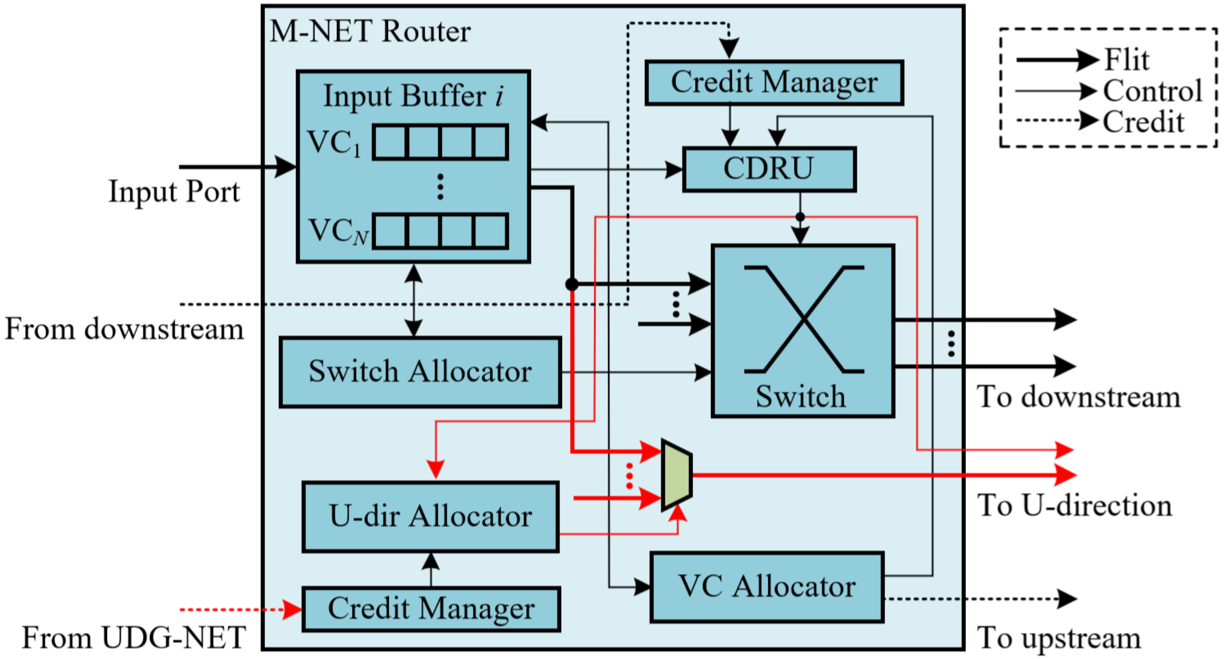


Fig. M-NET router architecture.

Load Distribution

Load Balance Model

CDRU makes decisions based on a probability model that we have designed.

The ideal ratio of the number of packets in M-NET to the number of packets in U-NET is

$$k = \frac{Pkt_{main}}{Pkt_{udg}} = \frac{N_{main} \cdot AvgHops_{udg}}{N_{udg} \cdot AvgHops_{main}}$$

However, not all packets benefit equally when they are re-injected into U-NET.

The probability of the packet with a hop count of h to its destination being re-injected at the current node is $p(h)$. $p(0) = 0$.

Hence, the probability of the packet with a hop count of H from the source to the destination being re-injected to U-NET is:

$$P(H) = \begin{cases} p(1), & H = 1 \\ \sum_{i=1}^{H} p(i) \prod_{m=1}^{i-1} [1 - p(m)], & H > 1 \end{cases}$$

RUNoC Architecture

Load Distribution

Load Balance Model

The value of $P(H)$ should satisfy: $P(H) = \frac{Pkt_{udg}}{Pkt_{udg}+Pkt_{main}} = \frac{1}{k+1}$

$p(h_{th}) = 1/(k+1)$, We set up three probabilistic models to describe $p(h)$:

(1) KP Model: $p(h) = \begin{cases} 0, & h < h_{th} \\ \alpha^{h-h_{th}} p(h_{th}), & h \geq h_{th} \end{cases}$

(2) L Model: $p(h) = \begin{cases} 0, & h < h_{th} \\ \frac{\beta(h - h_{th}) + h_{th}}{h_{th}} p(h_{th}), & h \geq h_{th} \end{cases}$

(3) CA Model: Hybrid

The load balance model is pre-calculated and is stored into **CDRU** in each M-NET router, which is based on look-up table.

RUNoC Architecture

Router Architecture

U-NET Router

UDG Routing Unit (URU) within the router directly routes the packets to the corresponding output port of destination.

Deadlock & Livelock Freedom

Since no packet travels from U-NET to M-NET, there is no risk of inter-layer deadlock and livelock.

No packet is directly injected into U-NET by NI, thus, there is no protocol level deadlock in U-NET.

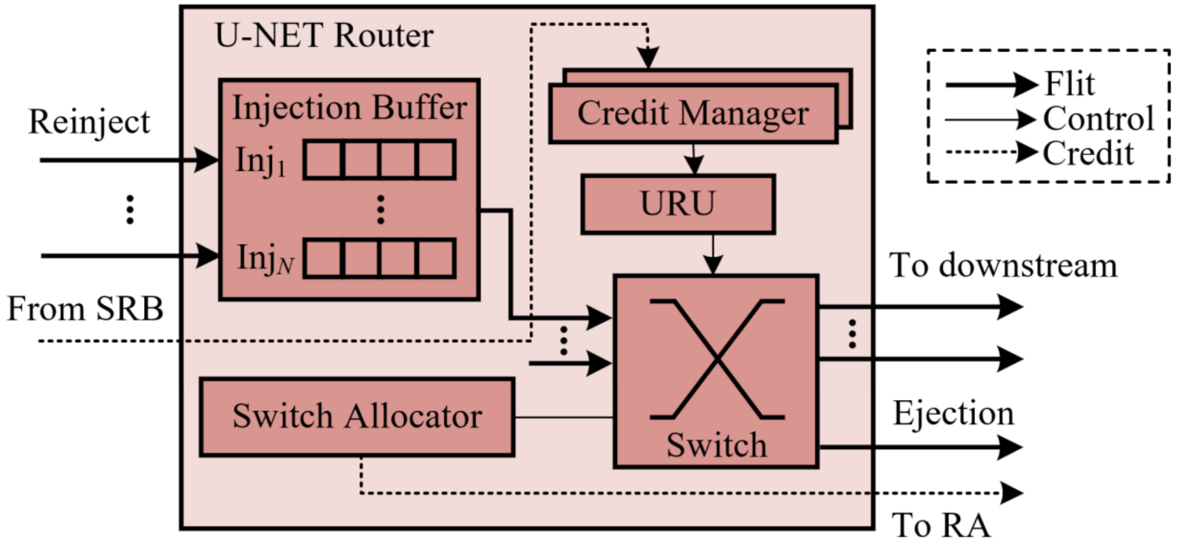


Fig. U-NET router architecture, the number of ports matches the number of U-NET nodes.

Shared Row Buffer and Network Interface

Shared Row Buffer

Each row of a subnet has an SRB.

SRAM: Storing the packets from the U-NET router.

Address FIFO: Managing free space of SRAM.

Request Generator: Generating requests to NI.

Initially, all the index of the SRAM are stored in the address FIFO.

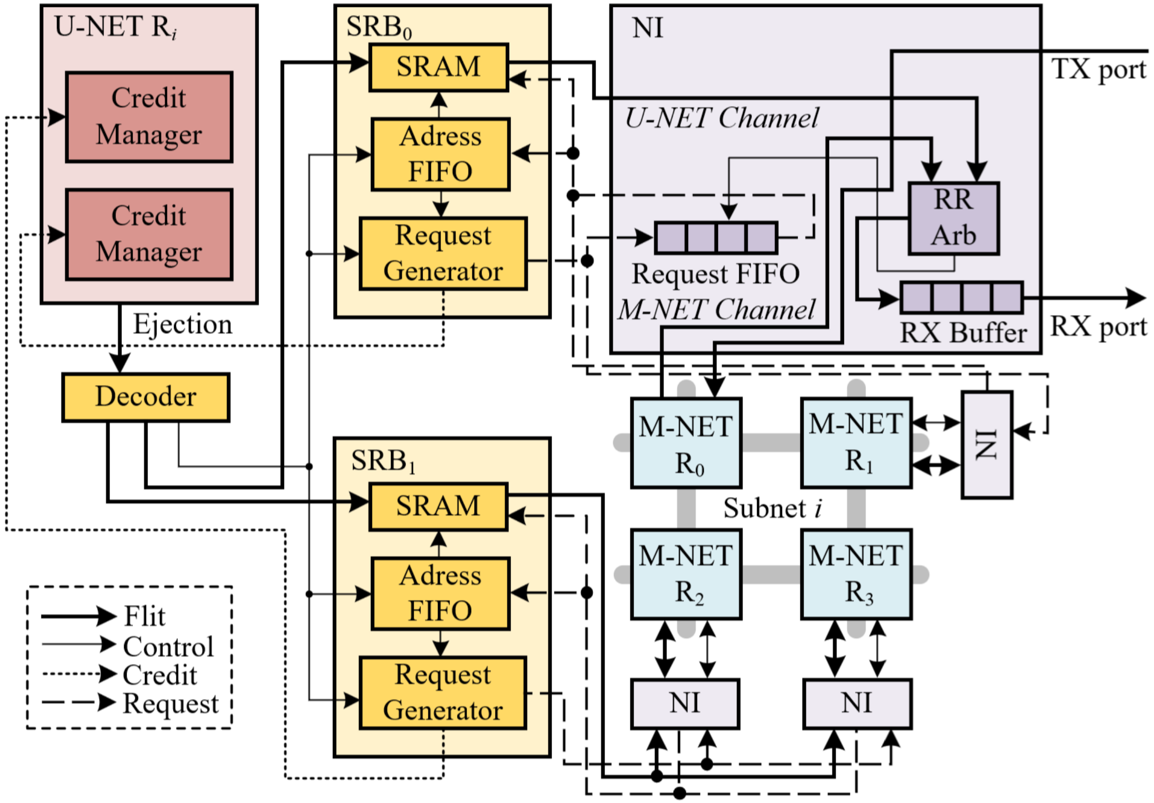


Fig. A subnet with SRBs and modified NI.

RUNoC Architecture

Shared Row Buffer and Network Interface

Network Interface

TX channel: Connected to the injection port of M-NET router and utilizing credit-based flow control mechanism.

RX channel:

M-NET channel: Ejection of M-NET router.

U-NET channel: Receiving packets from SRB.

Request FIFO: Communication with SRB.

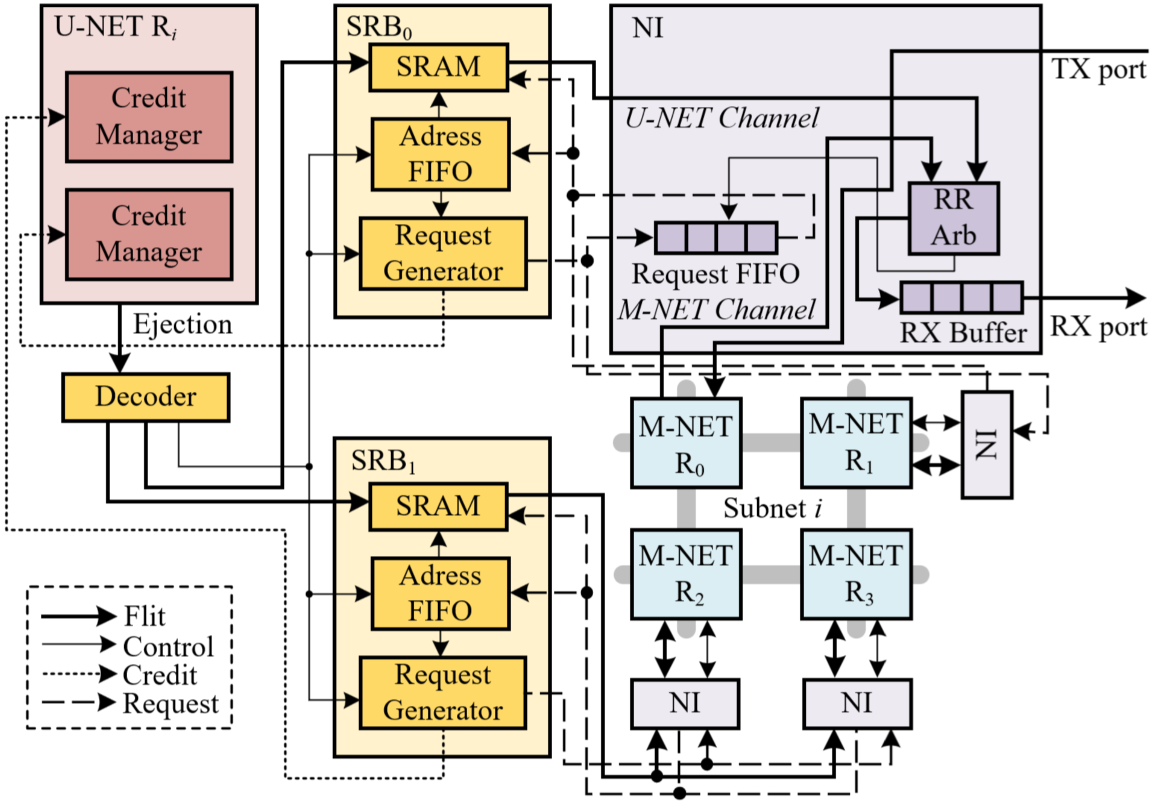


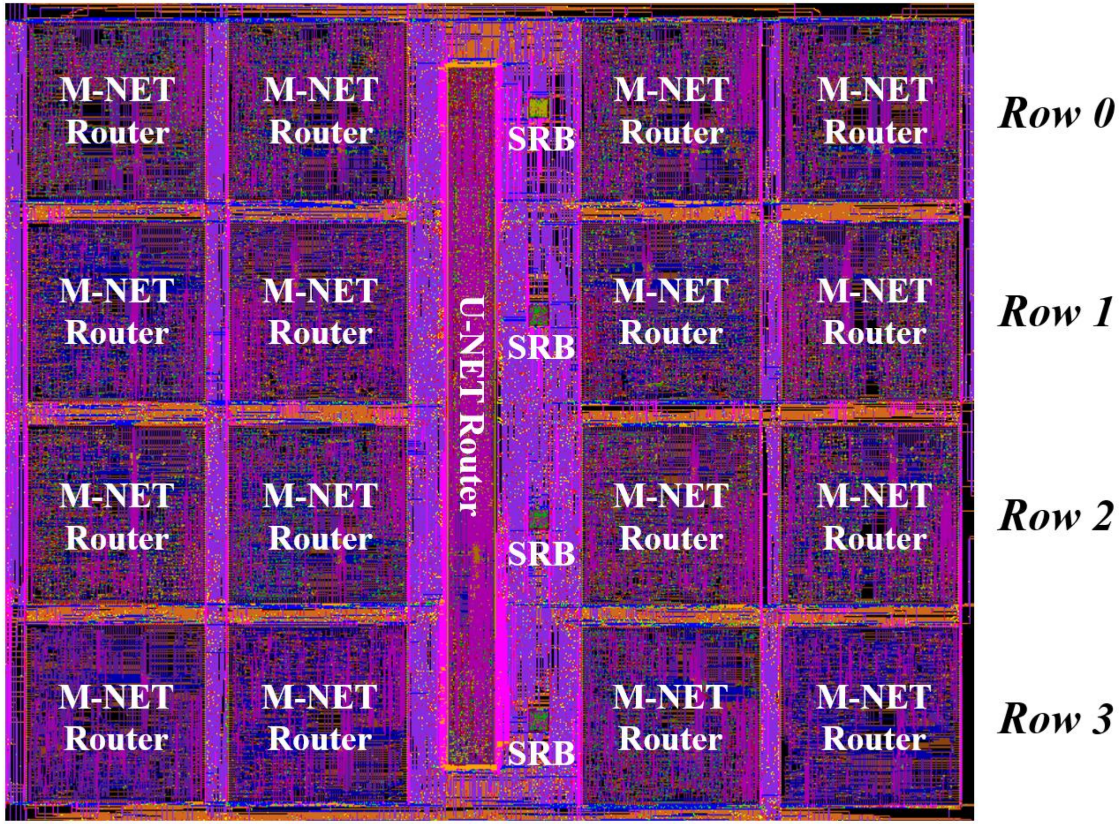
Fig. A subnet with SRBs and modified NI.

Setup and Implementation of RUNoC

Setup

Network Parameters	
Network Size	M-NET: 12*12, U-NET: 3*3
Routing Algorithm	XY
Packet Size	Single Flit
Router Latency	One Cycle
Virtual Channels	1
FIFO Depth	8
SRB Size	16
h_{th}	4

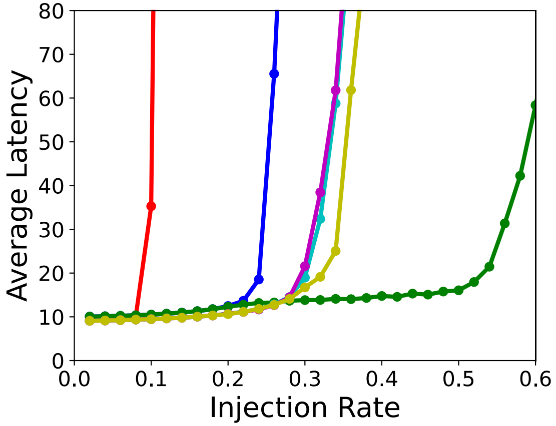
Layout of One Subnet



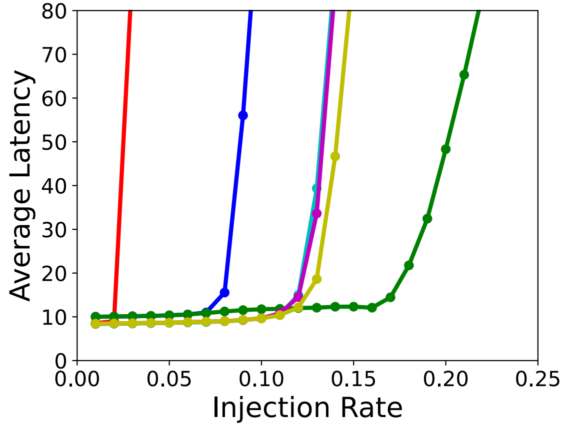
Experiments

Network Performance with Typical Traffic

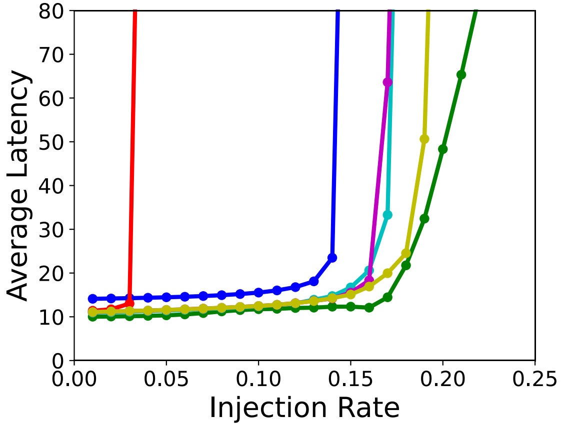
Network Latency



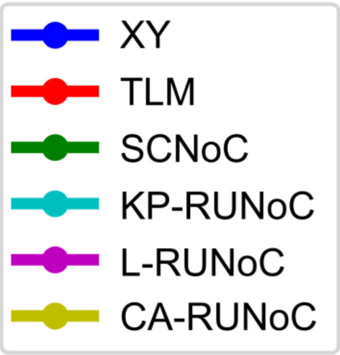
(a) Uniform Random



(b) Transpose



(c) Bit-complement



Saturation throughput

	Saturation Throughput		
	Uniform random	Bit-complement	Transpose
XY	0.27	0.15	0.10
SCNoC	0.67	0.23	0.23
KP-RUNoC	0.36	0.18	0.15
L-RUNoC	0.36	0.18	0.15
CA-RUNoC	0.39	0.20	0.16

Single model: 20% - 50% improvement compared to XY.
CA model: 33% - 60% improvement compared to XY.
Close to SCNoC under Bit-complement.

Experiments

Area and Power

Area and Power of RUNoC and other schemes

	Area (um^2)	Power (mW)
XY	2.84	330.38
SCNoC	9.31	1132.80
TLM	5.40	655.11
RUNoC	6.04	729.57

Area efficiency of RUNoC and other schemes

RUNoC exhibits the highest 34% improvement with the two non-random traffic patterns compared to state-of-the-art. RUNoC is well-suited for large-scale requirements.

	Saturation Throughput / Area ($10^4 mm^{-2}$)		
	Uniform random	Bit-complement	Transpose
SCNoC	719.76	247.08	247.08
KP-RUNoC	595.60	297.80	248.17
L-RUNoC	595.60	297.80	248.17
CA-RUNoC	645.24	330.89	264.71

Experiments

Conclusion and Future Work

Conclusion

- We propose a new partition-based asymmetric TLN architecture named RUNoC. RUNoC utilizes congestion-aware probabilistic model implemented in CDRU to balance the workload of the two layers. We develop SRBs and modified NI to achieve effective function.
- RUNoC achieves up to 60% improvement in performance compared to XY routing and up to 34% improvement in area efficiency compared to the state-of-the-art.

Future Work

- Real-world traffic experiment.
- Embedding RUNoC into a many-core system.

Thank you!
Q & A