

Article

Not peer-reviewed version

Design of Identical Strictly and Rearrangeably Nonblocking Folded Clos Networks with Equally Sized Square Crossbars

[Yamin Li](#) *

Posted Date: 30 May 2025

doi: 10.20944/preprints202505.2501.v1

Keywords: multistage interconnection networks; strictly nonblocking; rearrangeably nonblocking; Clos network; folded Clos network; fat-tree; k-ary n-tree; square crossbar switch; crosspoint ratio



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

Design of Identical Strictly and Rearrangeably Nonblocking Folded Clos Networks with Equally Sized Square Crossbars

Yamin Li 

Computer Architecture Laboratory, Department of Computer Science, Faculty of Computer and Information Sciences, Hosei University, Tokyo 184-8584, Japan; yamin@hosei.ac.jp

Abstract: Clos networks and their folded versions, fat-trees, are widely adopted in interconnection network designs for data centers and supercomputers. There are two main types of Clos networks: strictly nonblocking Clos networks and rearrangeably nonblocking Clos networks. Strictly nonblocking Clos networks can connect an idle input to an idle output without interfering with existing connections. Rearrangeably nonblocking Clos networks can connect an idle input to an idle output with rearrangements of existing connections. Traditional strictly nonblocking Clos networks have two drawbacks. One drawback is the use of crossbars with different numbers of input and output ports, whereas the currently available switches are square crossbars with the same number of input and output ports. Another drawback is that every connection goes through a fixed number of stages, increasing the length of the communication path. A drawback of traditional fat-trees is that the root stage uses different sized crossbar switches than the other stages. To solve these problems, we propose an Identical Strictly NonBlocking folded Clos (ISNBC) network that uses equally sized square crossbars for all switches. Correspondingly, we also propose an Identical Rearrangeably NonBlocking folded Clos (IRNBC) network. Both ISNBC and IRNBC networks can have any number of stages, can use equally sized square crossbars with no unused switch ports, and can utilize shortcut connections to reduce communication path lengths. Moreover, both ISNBC and IRNBC networks have a lower switch crosspoint cost ratio to a single crossbar than their corresponding traditional Clos networks.

Keywords: multistage interconnection networks; strictly nonblocking; rearrangeably nonblocking; Clos network; folded Clos network; fat-tree; k -ary n -tree; square crossbar switch; crosspoint ratio

1. Introduction

Clos networks [1] and fat-trees [2] have been widely used in interconnection network designs for modern data centers and supercomputers [3]. A traditional unidirectional nonblocking Clos network was originally designed for telecommunications. It is a type of multistage circuit-switching network that replaces a single large crossbar to reduce hardware costs in terms of crosspoints.

A 3-stage traditional unidirectional Clos network topology [1] is parameterized by two integers n and m , where n is the number of sources connecting to an ingress stage crossbar switch, the number of destinations connecting to an egress stage crossbar switch is also n , and m is the number of crossbar switches in the middle stage. The ingress stage has n crossbar switches, and the number of crossbar switches in the egress stage is also n . Therefore, the total number of sources is n^2 , and the total number of destinations is also n^2 . A switch in the ingress stage is an $n \times m$ (n inputs and m outputs) crossbar. A switch in the egress stage is an $m \times n$ crossbar. A switch in the middle stage is an $n \times n$ crossbar. There is exactly one connection between each ingress stage switch and each middle stage switch. And there is exactly one connection between each middle stage switch and each egress stage switch. If $m \geq 2n - 1$, it is a strictly nonblocking network, meaning that the network can connect a free source to a free destination without interfering with existing connections. If $m \geq n$, it is a rearrangeably

nonblocking network, meaning that the network can connect a free source to a free destination with rearrangements of existing connections.

Traditional unidirectional strictly nonblocking Clos networks have two drawbacks. One drawback is that all connections from sources to destinations pass through a fixed number of stages. For example, the path length in a 3-stage unidirectional Clos network is always 4. Another drawback is that it uses $n \times m$ and $m \times n$ crossbars with different numbers of input and output ports. However, nowadays the available switches are square crossbars with the same number of input and output ports. If we use the same square crossbars for all switches, there will be a large number of unused ports.

A fat-tree is a folded version of the Clos network [4]. It merges the corresponding ingress and egress switches. Then the merged stage is called a leaf stage and the middle stage is called a root stage. A fat-tree can utilize shortcut connections. That is, a connection does not have to go through all stages. For example, if the source and destination are connected to the same leaf switch, the connection does not need to go to the root stage. Thus, the path length will be 2 instead of 4. A drawback of fat-tree networks is that the root stage uses different sized crossbar switches than the other stages.

A k -ary n -tree [5] is a kind of parametric fat-tree, where k is the arity or number of links of a switch that connects to the previous or next stage, and n is the number of stages. That is, the switch radix is $2k$. A k -ary n -tree Clos network can be constructed with back-to-back k -ary n -fly butterfly [6]. The 2-ary n -tree Clos network is also called the Beneš network [7]. A k -ary fat-tree [8] is a bidirectional $(k/2)$ -ary n -tree Clos network where k is even.

A packet can be routed to an arbitrary middle switch in a rearrangeably nonblocking Clos network or an arbitrary root switch in a rearrangeably nonblocking folded Clos network and then to their ultimate destination. This increases hardware cost and packet latency. Mirrored k -ary n -tree networks [9] and peer k -ary n -tree networks [10] focus on increasing network capacity and reducing hardware costs and packet latency.

A strictly nonblocking folded Clos network using same sized crossbar switches is proposed in [11]. The proposed network has a 2-stage structure and uses multiple links between a leaf switch and a root switch. It may have unused switch ports. The number of unused switch ports is reduced by adjusting the number of leaf switches, but the existence of unused switch ports increases the hardware cost. A flexible folded Clos network is proposed in [12]. To reduce the blocking probability, a second group of switches is added to the root stage. [13] extends the number of groups from two to a general number S . All these networks have only two stages, making it difficult to scale the network.

The new contributions of this paper are summarized as follows. We propose an Identical Strictly NonBlocking folded Clos (ISNBC) network and an Identical Rearrangeably NonBlocking folded Clos (IRNBC) network. Both ISNBC and IRNBC networks can have any number of stages to increase the system's scalability, can use equally sized square crossbars with no unused switch ports to accommodate currently available switches at low costs, and can utilize shortcut connections to reduce communication path lengths. Moreover, both ISNBC and IRNBC networks have a lower crosspoint ratio to a single crossbar than their corresponding traditional nonblocking Clos networks.

The rest of the paper is organized as follows. In Section 2, we review some related multistage interconnection networks. In Section 3, we propose identical strictly and rearrangeably nonblocking folded Clos networks consisting of equally sized square crossbars. In Section 4, we evaluate the hardware cost from the crosspoint perspective and show that the costs of the proposed networks are lower than their corresponding traditional networks. We conclude the paper and suggest some future research topics in Section 5.

2. Related Works

There are many different types of multistage interconnection networks. This section reviews some related multistage interconnection networks.

2.1. Traditional Nonblocking Clos Networks

A traditional unidirectional strictly nonblocking Clos network [1] has odd switch stages. Consider a traditional unidirectional strictly nonblocking Clos network with three switch stages: The ingress stage has n switches, and each switch is an $n \times m$ (n inputs and m outputs) crossbar. The middle stage has m switches, and each switch is an $n \times n$ crossbar. The egress stage has n switches, and each switch is an $m \times n$ crossbar. The m outputs of an ingress switch are connected to m middle switches: Each output is connected to a different middle switch. The n outputs of a middle switch are connected to n egress switches: Each output is connected to a different egress switch. That is, there is exactly one connection between each ingress stage switch and each middle stage switch. And, there is exactly one connection between each middle stage switch and each egress stage switch. The total number of switches is $2n + m$, and the total number of compute nodes is $N = n^2$. We can see that crossbar switches with different numbers of input and output ports are used at the ingress and egress stages.

Assume that $N - 1$ connections have been built, which means that an ingress switch has an idle input and an egress switch has an idle output. We want to build a connection from the idle input to the idle output without interfering with existing connections. This is called strictly nonblocking. Suppose that the idle input is at switch i in the ingress stage, and the idle output is at switch j in the egress stage. That is, $n - 1$ connections were built in switch i , and $n - 1$ connections were built in switch j . In the worst case, these $2(n - 1)$ connections used $2(n - 1)$ switches in the middle stage. If we have another switch in the middle stage, we can build a connection from the idle input to the idle output through that switch without interfering with existing connections. Therefore, the condition to achieve strictly nonblocking is $m \geq 2(n - 1) + 1 = 2n - 1$.

An $x \times y$ (x inputs and y outputs) crossbar has xy crosspoints. Let $m = 2n - 1$. Then there are $nm \times n + n^2 \times m + mn \times n = 3n^2m = 3n^2(2n - 1)$ crosspoints in a traditional unidirectional strictly nonblocking Clos network. There are $N = n^2$ inputs and $N = n^2$ outputs. If we use a single $N \times N$ crossbar, it requires $N^2 = n^4$ crosspoints. For $n = 6$, the Clos network requires $3n^2(2n - 1) = 3 \times 6^2 \times 11 = 1188$ crosspoints, less than $N^2 = n^4 = 6^4 = 1296$ crosspoints in the single crossbar's implementation.

A folded version of the traditional unidirectional strictly nonblocking Clos network can be constructed by combining ingress and egress switches to form leaf switches. The middle switches become root switches. A leaf switch is an $(n + m) \times (m + n)$ crossbar. A root switch is still an $n \times n$ crossbar. We can see that the different sized crossbar switches are used at the leaf and root stages.

A Clos network is rearrangeably nonblocking if and only if $m \geq n$. In such a case, an input of an ingress switch can be connected to an output of an egress switch using a middle switch. A folded version of the traditional unidirectional rearrangeably nonblocking Clos network can be constructed by combining ingress and egress switches.

Traditional unidirectional strictly nonblocking Clos networks use crossbars with different numbers of input and output ports. The root stage of the folded version of the traditional unidirectional strictly and rearrangeably nonblocking Clos networks uses different sized crossbar switches than the other stages.

2.2. K-Ary N-Tree Clos Networks

A k -ary n -tree Clos network can be created with k -ary n -fly butterfly networks. A k -ary n -fly butterfly network [6] has $N = k^n$ compute nodes. It has n stages. Each stage has N/k switches and each switch has k input ports and k output ports ($2k$ is the radix of the switch). A k -ary 1-fly butterfly network has a $k \times k$ switch. There are k input ports and k output ports. Each of $N = k^1 = k$ compute nodes is connected to an input port and an output port. A k -ary 2-fly butterfly network has two stages: stage 0 and stage 1. Each stage has k switches. $N = k^2 = k^2$ compute nodes are connected to the input ports of switches in stage 0 and the output ports of switches in stage 1. Each output port of a switch in stage 0 is connected to an input port of a different switch in stage 1.

Butterfly minimizes the network diameter and reduces the network cost. However, there is a lack of path diversity because there is only one path between the source node and the destination node. Butterfly is a blocking network. In addition, it cannot exploit the locality of traffic because all packets must traverse the diameter of the network [4].

Multistage rearrangeably nonblocking Clos networks are also called k -ary n -tree Clos networks. A k -ary n -tree Clos network can be created by combining two k -ary n -fly butterfly networks [6] back-to-back where the two back stages are fused [4]. There are $2n - 1$ stages. The $n - 1$ stages on the left to the middle stage form an input network and the $n - 1$ stages on the right to the middle stage form an output network. It solves the problem of lack of path diversity in butterfly networks. The input network can route packets from any source compute node to any middle stage switch. The output network can route packets from any middle stage switch to any destination compute node. Like a k -ary n -fly butterfly network, the links in a k -ary n -tree Clos network are also unidirectional.

When $k = 2$, the k -ary n -tree Clos network is also called the Beneš network [7]. There are $(2 \log(N) - 1)$ stages where N is the number of compute nodes. Each stage contains $N/2$ switches and each switch is a 2×2 crossbar. For example, a 3-stage Beneš network has 4 compute nodes, a 5-stage Beneš network has 8 compute nodes, and a 7-stage Beneš network has 16 compute nodes.

In a k -ary n -tree Clos network, a packet needs to be routed first to an arbitrary middle stage switch and then to its ultimate destination. It is a rearrangeably nonblocking network. The cost and latency of a k -ary n -tree Clos network is nearly double that of a k -ary n -fly butterfly network with equal node capacity [4].

Because in a unidirectional k -ary n -tree Clos network, the nodes in the input side and the nodes in the output side at the same row are the same compute nodes, and the ports of a switch are unidirectional, we can fold the unidirectional k -ary n -tree Clos network and combine two unidirectional switches to a bidirectional switch. And, we use bidirectional links to connect switch ports. We call it a k -ary n -tree folded Clos network, or a k -ary n -tree fat-tree [5]. It has n stages and k^n compute nodes.

A k -ary n -tree fat-tree network can be thought of as a bidirectional k -ary n -fly butterfly network with compute nodes connected to stage 0. Compared to the unidirectional k -ary n -tree Clos network, a k -ary n -tree fat-tree network can exploit the traffic locality, because a packet needs to be routed only to a nearest common ancestor (NCA) of the source and destination and then to its ultimate destination. This means that packets may no longer need to be routed to the root switch, reducing the path they take.

We can re-design the switches in a unidirectional k -ary n -tree Clos network so that the switches have bidirectional ports. Also, we use the bidirectional links and double the number of compute nodes (the left nodes and right nodes are distinct compute nodes). Thus, we get a bidirectional k -ary n -tree Clos network. It has $2n - 1$ stages and $2k^n$ compute nodes. All the switches in a bidirectional k -ary n -tree Clos network have the same radix which is $2k$.

When $n = 3$ and k is even, the bidirectional $(k/2)$ -ary n -tree Clos network is also called the k -ary fat-tree network [8]. It has a fixed 3 stages (layers). The root stage is called a core (spine) layer. There are k pods below the core layer. Each pod contains two layers: the aggregation layer in the middle stage and the edge layer in the leaf stage.

2.3. Mirrored and Peer K-Ary N-Tree Networks

To implement the nonblocking routing, the network must provide a high path diversity so that a packet can be routed to an arbitrary middle stage switch in a Clos network or an arbitrary root switch in a fat-tree network and then to their ultimate destination. This approximately doubles the number of switches and links, resulting in a high hardware cost and a high packet latency [4]. Both mirrored k -ary n -tree (MiKANT) networks [9] and peer k -ary n -tree networks [10] focus on increasing network capacity and reducing hardware costs and packet latency.

A k -ary n -tree fat-tree network has n stages. If the number 0 represents the leaf stage, then the root stage is numbered $n - 1$. A MiKANT network [9] consists of two k -ary n -tree fat-tree networks joined back-to-back where the switches in stage $n - 2$ of a fat-tree serve as root switches for the other fat-tree. That is, the MiKANT network has $2n - 2$ stages. Fat-tree 0 and fat-tree 1 are used to distinguish

between two fat-trees. Compared to a k -ary n -tree fat-tree network, MiKANT doubles the number of compute nodes. Compared to a bidirectional k -ary n -tree Clos network, MiKANT uses fewer switches with equal node capacity. If the source and destination nodes belong to the same fat-tree, MiKANT behaves like a k -ary n -tree fat-tree network. MiKANT reduces the path length and path diversity when the source and destination nodes belong to different fat-trees.

In a k -ary n -tree fat-tree network, the root switch has radix k and the other switches have radix $2k$. The radix- $2k$ switches can be used at the root stage and k compute nodes can be connected to each root switch. Therefore, the root switches are the same as the leaf switches. It is called a peer k -ary n -tree or a peer fat-tree [10] because there is no difference between root and leaf. The peer fat-tree reduces both hardware cost and average distance, and meanwhile provides nonblocking routing functionality for half of the source and destination node pairs. Note that a peer k -ary n -tree network is not a bidirectional k -ary n -fly butterfly network. A bidirectional k -ary n -fly butterfly network connects compute nodes to either stage 0 or stage $n - 1$. However, a peer k -ary n -tree network connects compute nodes to both stage 0 and stage $n - 1$. Therefore, the number of compute nodes in a peer k -ary n -tree is twice that in a bidirectional k -ary n -fly butterfly network. Actually, a bidirectional k -ary n -fly butterfly network is the same as a k -ary n -tree fat-tree network.

2.4. Twisted-and-Folded Clos Networks

A two-stage twisted-and-folded Clos network using equally sized square crossbar switches is proposed in [11]. There are multiple links between a leaf switch and a root switch. It has k switches in the leaf stage and m switches in the root stage. The condition to achieve strictly nonblocking is $m \geq 2\lfloor(n-1)/v\rfloor + 1$, where n is the number of ports connected to compute nodes at a leaf switch and v is the number of links between a leaf switch and a root switch. Then it has $N = nk$ compute nodes in total. It uses $r \times r$ square crossbars where $r = \max\{kv, n + mv\}$ and k , the number of leaf switches, is determined so that kv is closest to $n + mv$. This keeps the number of unused ports as low as possible. Then if $kv = n + mv$, there is no unused port, otherwise, $|n + mv - kv|$ ports are unused on every switch in either leaf stage (if $kv > n + mv$) or root stage (if $kv < n + mv$).

Let's take a look at the following examples proposed in [11]. For $v = 2$ and $n = 9$, we calculate $m = 2\lfloor(n-1)/v\rfloor + 1 = 2\lfloor(9-1)/2\rfloor + 1 = 9$ and $n + mv = 9 + 9 \times 2 = 27$. There are two possible values of k , as shown below.

1. Let $k = 13$, then $kv = 13 \times 2 = 26$ which is smaller than $n + mv = 27$. The network will use $k + m = 22$ switches and each switch is an $r \times r = (n + mv) \times (n + mv) = 27 \times 27$ crossbar. There is one $(27 - 26)$ unused port in each root switch. The number of compute nodes is $N = nk = 117$.
2. Let $k = 14$, then $kv = 14 \times 2 = 28$ which is larger than $n + mv = 27$. The network will use $k + m = 23$ switches and each switch is an $r \times r = (kv) \times (kv) = 28 \times 28$ crossbar. There is one $(28 - 27)$ unused port in each leaf switch. The number of compute nodes is $N = nk = 126$.

A flexible twisted-and-folded Clos network is proposed in [12]. It eliminates the strict condition $m \geq 2\lfloor(n-1)/v\rfloor + 1$. To reduce the blocking probability, a second group of switches is added to the root stage. If it cannot establish a connection through the root switch in group 1, it will try to use the root switch in group 2 (two-step model).

The number of groups can be extended to a general number S (S -step model), as presented in [13]. The twisted-and-folded Clos networks introduced in [11–13] have only two stages with nk compute nodes.

3. Proposed Identical Nonblocking Folded Clos Networks

In this section, we first present Unidirectional Strictly NonBlocking Clos (USNBC) networks and Unidirectional Rearrangeably NonBlocking Clos (URNBC) networks. Based on these unidirectional networks, we present the construction methods of Identical Strictly NonBlocking folded Clos (ISNBC) networks and Identical Rearrangeably NonBlocking folded Clos (IRNBC) networks, as listed in Table 1. Note that the number of stages in a unidirectional is odd, e.g., $2k - 1 = 3, 5, 7$ for $k = 2, 3, 4$, where k is

the number of stages for corresponding identical folded networks. Here we only show the cases of $k = 2, 3$, and 4 , but for $k > 4$, we can construct USNBC, URNBC, ISNBC, and IRNBC in a similar way.

Table 1. Proposed nonblocking Clos networks.

Strictly or Rearrangeably	Network	Stage, Unfolded or Folded
Strictly nonblocking	USNBC	3-stage, 5-stage, and 7-stage Clos networks
	ISNBC	2-stage, 3-stage, and 4-stage folded Clos networks
Rearrangeably nonblocking	URNBC	3-stage, 5-stage, and 7-stage Clos networks
	IRNBC	2-stage, 3-stage, and 4-stage folded Clos networks

3.1. Proposed Identical Strictly NonBlocking Folded Clos (ISNBC) Networks

As mentioned before, there are two drawbacks in the traditional strictly nonblocking Clos network. One is the use of $n \times m$ and $m \times n$ crossbars with unequal numbers of input and output ports. Another is that all connections pass through a fixed number of stages. A drawback of the traditional nonblocking folded Clos network is that the root stage uses different sized crossbar switches than the other stages. These problems can be solved by using an ISNBC network consisting of equally sized square crossbars. In this subsection, we present the USNBC network and the corresponding ISNBC network.

3.1.1. Two-Stage ISNBC Networks

To construct a 2-stage ISNBC network, we first construct a 3-stage USNBC network. In a 3-stage USNBC network, let n be the number of inputs per switch in the ingress stage, m be the number of switches in the middle stage, and r be the number of switches in the ingress and egress stages. To ensure the strictly nonblocking property, we let $m = 2n$. To ensure that the ISNBC network uses equally sized square crossbars, we let $r = n + m = 3n$.

In the ingress stage, there are r switches and each switch is an $n \times m$ crossbar (n inputs and m outputs). In the middle stage, there are m switches and each switch is an $r \times r$ crossbar. In the egress stage, there are r switches and each switch is an $m \times n$ crossbar. Each output of a switch in the ingress stage is connected to an input of a different switch in the middle stage. Since there are r switches in the ingress stage, the number of inputs of a switch in the middle stage must be r , one input for an output of the r switches in the ingress stage. Each output of a switch in the middle stage is connected to an input of a different switch in the egress stage. Because the number of outputs of a switch in the middle stage is also r , each output is connected to an input of r switches in the egress stage. Then the number of compute nodes is $N = nr = n \times 3n = 3n^2$.

In summary, to construct a 3-stage USNBC network, we determine m and r as shown in Formula (1), where N is the number of compute nodes.

$$\begin{cases} m = 2n \\ r = n + m = 3n \dots \dots (3\text{-stage USNBC}) \\ N = nr = 3n^2 \end{cases} \quad (1)$$

To construct a folded version of a 3-stage USNBC network, the corresponding position switches in the ingress and egress stages are merged and expanded so that there are $n + m$ inputs and $m + n$ outputs in the combined switch. Then these switches have the same number of inputs and outputs which is $n + m$. The switch in the middle stage has r inputs and r outputs with $r = n + m$. Therefore all switches in the folded Clos network use the equally sized $(n + m) \times (m + n) = 3n \times 3n$ crossbars.

Figure 1(a) shows a 3-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = n + m = 6$. It has $N = nr = 3n^2 = 12$ compute nodes. A 2-stage ISNBC network, the folded version of the 3-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = n + m = 6$, is shown in Figure 1(b). It uses the equally sized $(n + m) \times (m + n) = 6 \times 6$ square crossbars for all switches. It can utilize shortcut

connections to reduce communication path lengths. For example, if the source and destination nodes are connected to the same leaf switch, the communication does not need to go through the root switch.

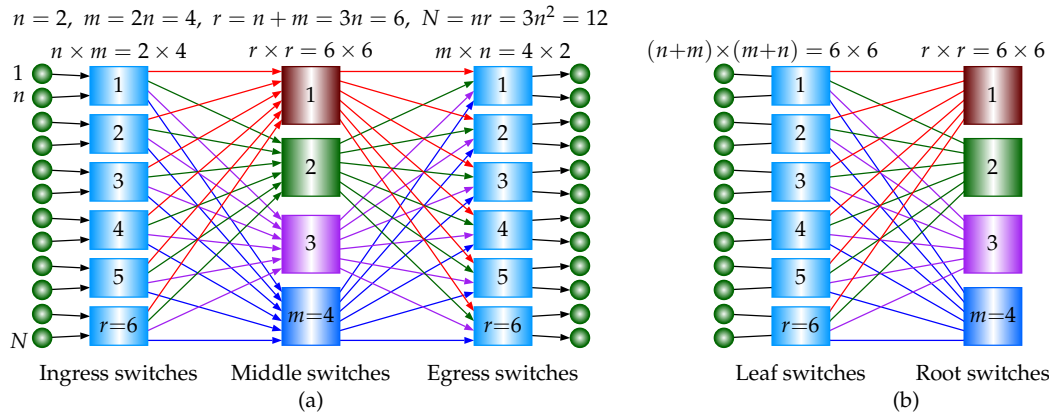


Figure 1. Proposed strictly nonblocking Clos networks ($m = 2n$ and $r = n + m$). (a) A 3-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = n + m = 6$. (b) A 2-stage ISNBC network composed of equally sized $(n+m) \times (m+n) = 6 \times 6$ square crossbars.

The root switch in Figure 1(b) and the middle switch in Figure 1(a) are the same, both are the $r \times r$ crossbar switches. However, the leaf switch in Figure 1(b) is not simply a combination of the $n \times m$ ingress switch and the $m \times n$ egress switch in Figure 1(a). A leaf switch is a square $(n+m) \times (m+n)$ crossbar. Referring to Figure 2, for $n = 2$ and $m = 2n = 4$, the number of crosspoints in Figure 2(a) is $n \times m + m \times n = 4n^2 = 16$. While the number of crosspoints in Figure 2(b) is $(n+m) \times (m+n) = 9n^2 = 36$. Any input x_i can be routed to any output y_j for $i, j = 0, 1, \dots, 5$. A crosspoint can be implemented using two 2-to-1 multiplexers, as shown in Figure 2(c).

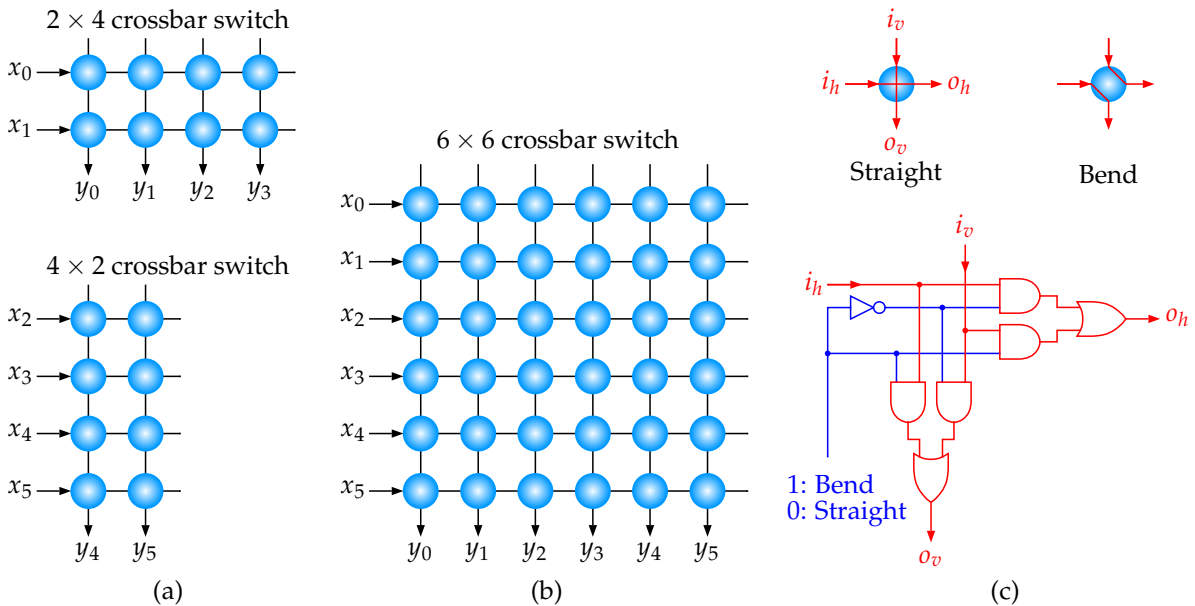


Figure 2. Merging and expanding an $n \times m$ crossbar switch and an $m \times n$ crossbar switch to a big square $(n+m) \times (m+n)$ crossbar switch for $n = 2$ and $m = 2n = 4$. (a) An $n \times m$ crossbar switch and an $m \times n$ crossbar switch. Each has 8 crosspoints. (b) A square $(n+m) \times (m+n)$ crossbar switch. There are 36 crosspoints. (c) Crosspoint states and implementation using two 2-to-1 multiplexers.

Figure 3(a) shows another 3-stage USNBC network with $n = 3$, $m = 2n = 6$, and $r = n + m = 9$. There are $r = 9$ ingress switches, $r = 9$ egress switches, and $m = 6$ middle switches. It has $N = nr = 27$ compute nodes. A 2-stage ISNBC network, the folded version of the 3-stage USNBC network with $n = 3$, $m = 2n = 6$, and $r = n + m = 9$, is shown in Figure 3(b). It uses the equally sized $3n \times 3n =$

9×9 square crossbars for all switches. It can utilize shortcut connections to reduce communication path lengths.

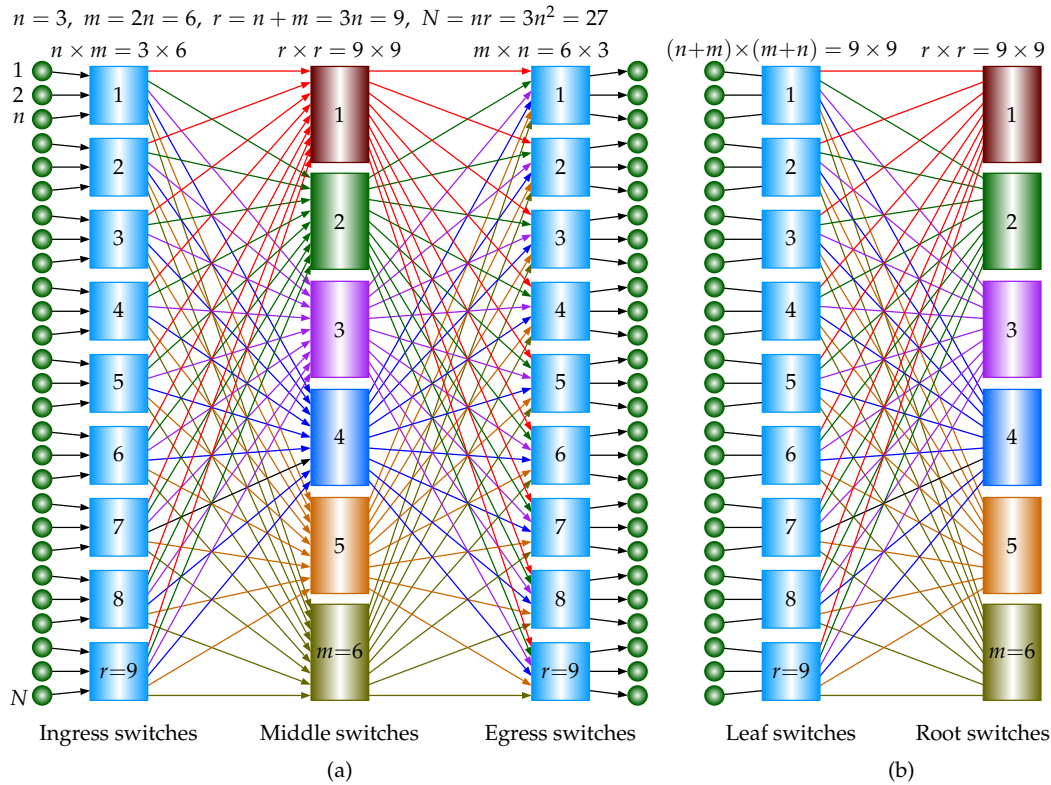


Figure 3. Proposed strictly nonblocking Clos networks ($m = 2n$ and $r = n + m$). (a) A 3-stage USNBC network with $n = 3$, $m = 2n = 6$, and $r = n + m = 9$. (b) A 2-stage ISNBC network composed of equally sized $(n + m) \times (m + n) = 9 \times 9$ square crossbars.

3.1.2. Three-Stage ISNBC Networks

To construct a 3-stage ISNBC network, we first construct a 5-stage USNBC network. By using 3-stage USNBC networks as building blocks, a 5-stage USNBC network can be constructed. For $m = 2n$, a 3-stage USNBC network has $N = (n + m)n = 3n^2$ compute nodes. As a building block, we remove the compute nodes and consider that the 3-stage USNBC network has $(n + m)n = 3n^2$ inputs and $(n + m)n = 3n^2$ outputs. The building blocks can be thought of as virtually $3n^2 \times 3n^2$ crossbars. We arrange m such building blocks in the middle stage. Then, in total, there are $3n^2 \times m = 6n^3$ inputs and $3n^2 \times m = 6n^3$ outputs in the middle stage. Correspondingly, we can arrange the same number of outputs in the ingress stage and the same number of inputs in the egress stage. Let r be the number of switches in the ingress and egress stages, then rm must be equal to $6n^3$. Therefore, we have $r = 6n^3 / m = 3n^2 \times m / m = 3n^2 = (n + m)n$.

In summary, to construct a 5-stage USNBC network, given an n , which is the number of inputs per switch in the ingress stage, there are $r = (n + m)n = 3n^2$ switches in the ingress stage, and each switch is an $n \times m$ crossbar with $m = 2n$. The middle stage has m building blocks and each building block is a 3-stage USNBC network with compute nodes removed. The egress stage has $r = (n + m)n = 3n^2$ switches and each switch is an $m \times n$ crossbar with $m = 2n$, as shown in Formula (2), where N is the number of compute nodes. The linking method is similar to the 3-stage USNBC network: Each output of a switch in the ingress stage is connected to an input of a different building block in the middle stage. Each output of a building block in the middle stage is connected to an input of a different switch in the egress stage. Because $r = 3n^2$, the number of compute nodes $N = nr = 3n^3$.

$$\begin{cases} m = 2n \\ r = (n + m)n = 3n^2 \dots \dots (5\text{-stage USNBC}) \\ N = nr = 3n^3 \end{cases} \quad (2)$$

Figure 4 shows a 5-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = 3n^2 = 12$. It has $N = nr = 3n^3 = 24$ compute nodes. There are $m = 2n = 4$ building blocks in the middle stage and each building block is a 3-stage USNBC network with compute nodes removed. The detailed network of a building block is shown at the bottom of the figure. We can see from the figure how the switches are linked together.

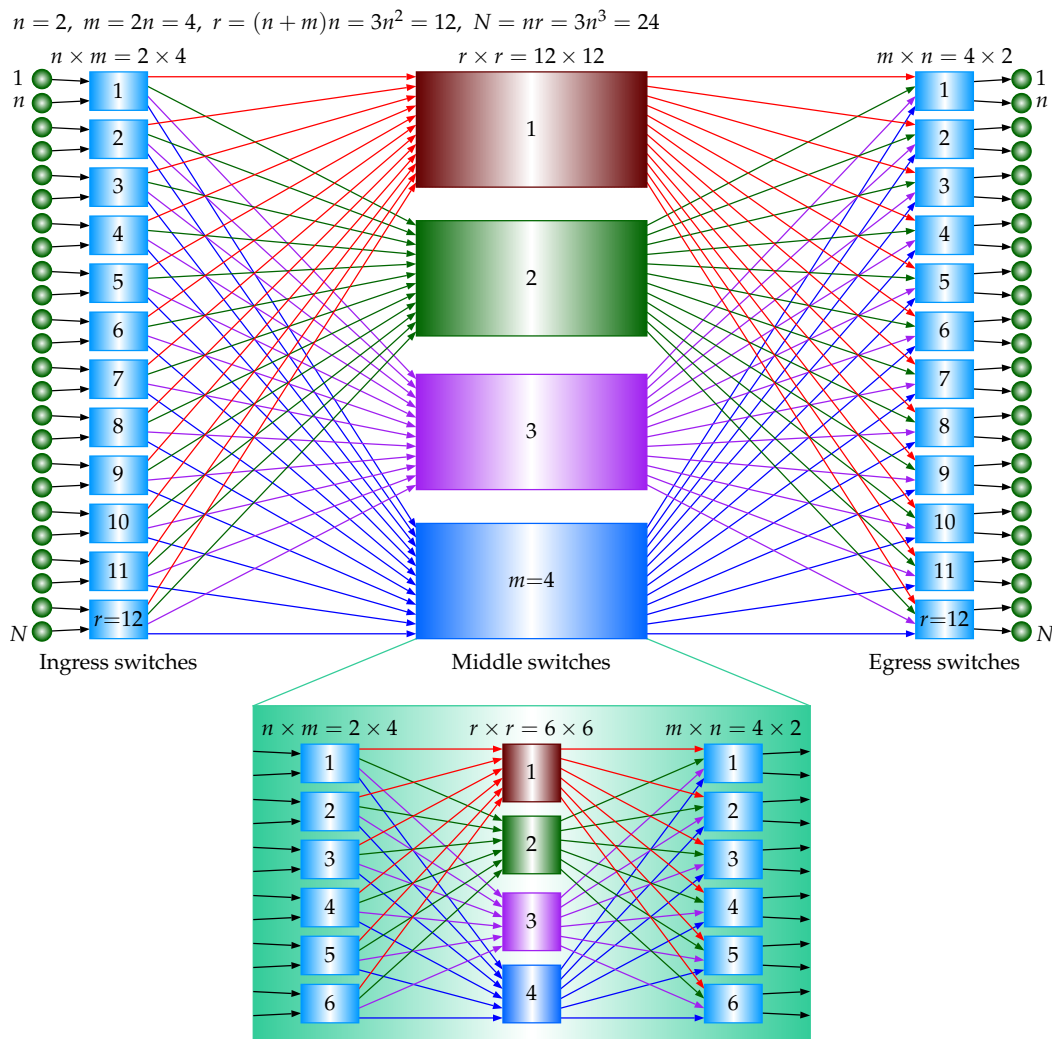


Figure 4. A 5-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = 3n^2 = 12$. There are $m = 2n = 4$ building blocks (3-stage USNBC, Figure 1(a)) in the middle stage.

A 3-stage ISNBC network, the folded version of the 5-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = 3n^2 = 12$, is shown in Figure 5. It uses the equally sized $(n + m) \times (m + n) = 6 \times 6$ square crossbars for all switches. Four building blocks are depicted in the two switch columns on the right, and each building block is a 2-stage ISNBC network with compute nodes removed, as shown in Figure 1(b). It can utilize shortcut connections to reduce communication path lengths.

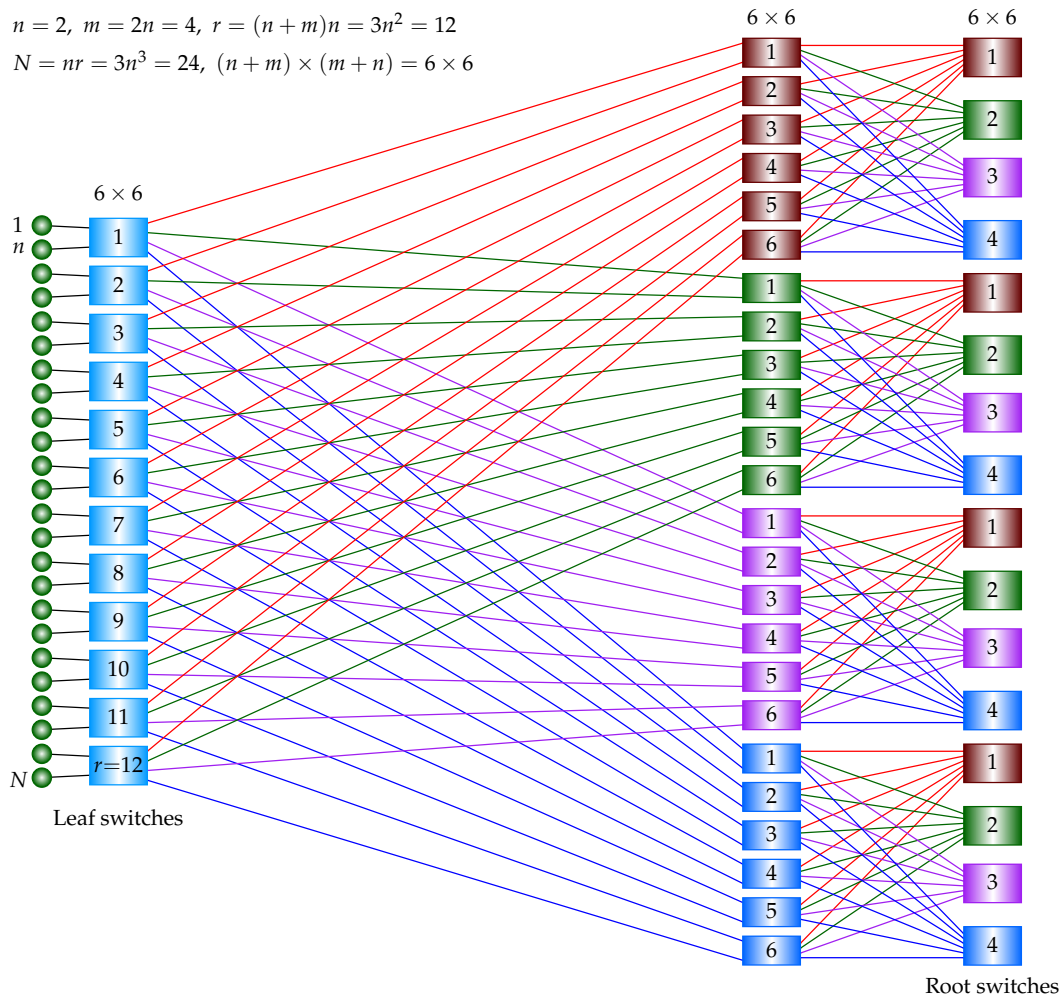


Figure 5. A 3-stage ISNBC network with $n = 2$ composed of equally sized $(n + m) \times (m + n) = 6 \times 6$ square crossbars (folded version of Figure 4).

3.1.3. Four-Stage ISNBC Networks

To construct a 4-stage ISNBC network, we first construct a 7-stage USNBC network. Similarly, by using 5-stage USNBC networks as building blocks, a 7-stage USNBC network can be constructed. For $m = 2n$, a building block of 5-stage has $3n^3$ inputs and $3n^3$ outputs. We arrange m such building blocks in the middle stage. Then, in total, there are $3n^3 \times m = 6n^4$ inputs and $3n^3 \times m = 6n^4$ outputs in the middle stage. Correspondingly, we arrange $r = 3n^3$ switches in the ingress stage and $r = 3n^3$ switches in the egress stage.

In summary, to construct a 7-stage USNBC network, given an n , which is the number of inputs per switch in the ingress stage, there are $r = 3n^3$ switches in the ingress stage, and each switch is an $n \times m$ crossbar with $m = 2n$. There are m building blocks in the middle stage and each building block is a 5-stage USNBC network with compute nodes removed. There are $r = 3n^3$ switches in the egress stage and each switch is an $m \times n$ crossbar with $m = 2n$, as shown in Formula (3), where N is the number of compute nodes. The linking method is similar to the 5-stage USNBC network.

$$\begin{cases} m = 2n \\ r = (n + m)n^2 = 3n^3 \dots \dots (7\text{-stage USNBC}) \\ N = nr = 3n^4 \end{cases} \quad (3)$$

Figure 6 shows a 7-stage USNBC network with $n = 2, m = 2n = 4$, and $r = 3n^3 = 24$. It has $N = nr = 3n^4 = 48$ compute nodes. There are $m = 2n = 4$ building blocks in the middle stage and

each building block is a 5-stage USNBC network with compute nodes removed. The detailed networks of building blocks are shown at the bottom of the figure.

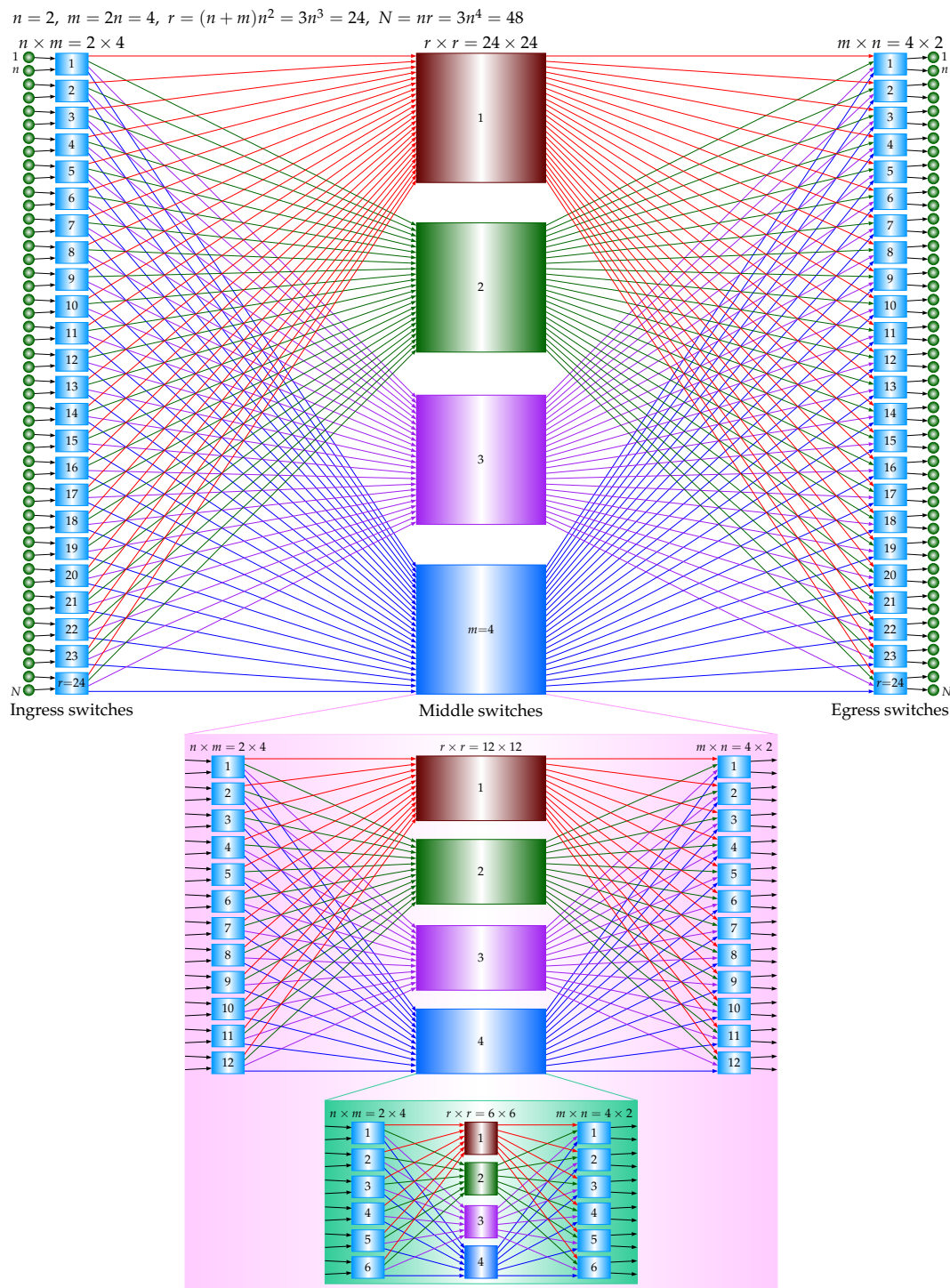


Figure 6. A 7-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = 3n^3 = 24$. There are $m = 2n = 4$ building blocks (5-stage USNBC, Figure 4) in the middle stage.

A 4-stage ISNBC network, the folded version of the 7-stage USNBC network with $n = 2$, $m = 2n = 4$, and $r = 3n^3 = 24$, is shown in Figure 7. It uses the equally sized $(n + m) \times (m + n) = 6 \times 6$ square crossbars for all switches. Four 3-stage ISNBC networks are shown in the three switch columns on the right. It can utilize shortcut connections to reduce communication path lengths.

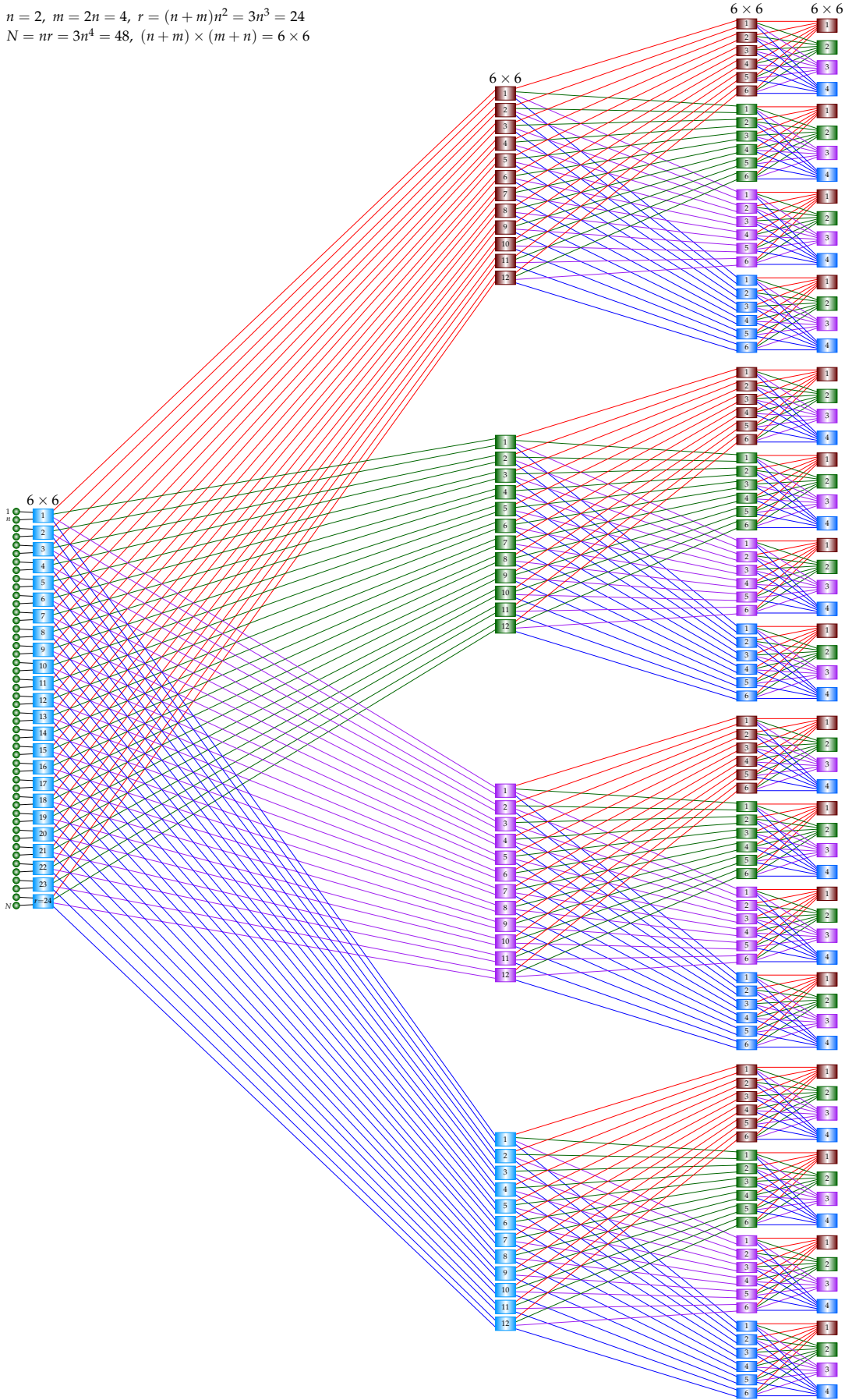


Figure 7. A 4-stage ISNBC network with $n = 2$ composed of equally sized $(n + m) \times (m + n) = 6 \times 6$ square crossbars (folded version of Figure 6).

Table 2 lists the numbers of compute nodes and switches of 2-, 3-, and 4-stage ISNBC networks. Let s be the number of stages. Then the number of compute nodes is $N = 3n^s$ and the number of switches is $(2^{s+1} - 3)n^{s-1}$. The crossbar size is listed in the right column.

Table 2. The numbers of compute nodes and switches in the ISNBC networks.

n	Number of compute nodes			Number of switches			Crossbar
	2-stage	3-stage	4-stage	2-stage	3-stage	4-stage	
2	12	24	48	10	52	232	6×6
3	27	81	243	15	117	783	9×9
4	48	192	768	20	208	1856	12×12
5	75	375	1875	25	325	3625	15×15
6	108	648	3888	30	468	6264	18×18
7	147	1029	7203	35	637	9947	21×21
8	192	1536	12,288	40	832	14,848	24×24
9	243	2187	19,683	45	1053	21,141	27×27
10	300	3000	30,000	50	1300	29,000	30×30

3.2. Proposed Identical Rearrangeably NonBlocking Folded Clos (IRNBC) Networks

In this subsection, we present the URNBC network and the corresponding IRNBC network composed of square crossbars of the same size.

3.2.1. Two-Stage IRNBC Networks

To construct a 2-stage IRNBC network, we first construct a 3-stage URNBC network. In a 3-stage URNBC network, let n be the number of inputs per switch in the ingress stage, m be the number of switches in the middle stage, and r be the number of switches in the ingress and egress stages. To ensure the rearrangeably nonblocking property, we let $m = n$. To ensure that the IRNBC network uses equally sized square crossbars, we let $r = n + m = 2n$.

In the ingress stage, there are r switches and each switch is an $n \times m$ crossbar (n inputs and m output). In the middle stage, there are m switches and each switch is an $r \times r$ crossbar. In the egress stage, there are r switches and each switch is an $m \times n$ crossbar. Each output of a switch in the ingress stage is connected to an input of a different switch in the middle stage. Since there are r switches in the ingress stage, the number of inputs of a switch in the middle stage must be r , one input for an output of the r switches in the ingress stage. Each output of a switch in the middle stage is connected to an input of a different switch in the egress stage. Because the number of outputs of a switch in the middle stage is also r , each output is connected to an input of r switches in the egress stage. Then the number of compute nodes is $N = nr = 2n^2$. In summary, to construct a 3-stage URNBC network, we determine m and r as shown in Formula (4), where N is the number of compute nodes.

$$\begin{cases} m = n \\ r = n + m = 2n \dots \dots (3\text{-stage URNBC}) \\ N = nr = 2n^2 \end{cases} \quad (4)$$

To construct a folded version of a 3-stage URNBC network, the corresponding position switches in the ingress and egress stages are merged and expanded so that there are $n + m$ inputs and $m + n$ outputs in the combined switch. Then these switches have the same number of inputs and outputs which is $n + m$. The switch in the middle stage has r inputs and r outputs with $r = n + m$. Therefore all switches in the folded Clos network use the equally sized $(n + m) \times (m + n) = 2n \times 2n$ crossbars.

Figure 8(a) shows a 3-stage URNBC network with $n = 2$, $m = n = 2$, and $r = n + m = 4$. It has $N = nr = 8$ compute nodes. A 2-stage IRNBC network, the folded version of the 3-stage URNBC network with $n = 2$, $m = n = 2$, and $r = n + m = 4$, is shown in Figure 8(b). It uses the equally sized $2n \times 2n = 4 \times 4$ square crossbars for all switches. It can utilize shortcut connections to reduce

communication path lengths. For example, if the source and destination nodes are connected to the same leaf switch, the communication does not need to go through the root switch.

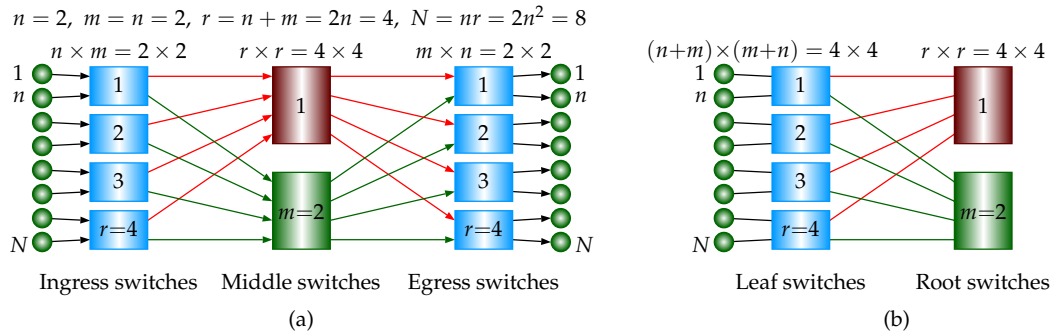


Figure 8. Proposed rearrangeably nonblocking Clos networks ($m = n$ and $r = n + m$). (a) A 3-stage URNBC network with $n = 2$, $m = n = 2$ and $r = n + m = 4$. (b) A 2-stage IRNBC network composed of equally sized $(n + m) \times (m + n) = 4 \times 4$ square crossbars.

Figure 9 shows the blocking and rearrangements in a rearrangeably nonblocking Clos network. Referring to Figure 9(a), connection $1 \rightarrow 4$ cannot be built because source node 2 (connected to the same switch as node 1) and destination node 3 (connected to the same switch as node 4) use different middle switches for their connections $2 \rightarrow 5$ and $7 \rightarrow 3$. Figure 9(b) shows the case of the folded version where a bidirectional link consists of two oppositely oriented unidirectional links. Figures 9(c) - (f) show that the connection $1 \rightarrow 4$ can be constructed after the rearrangements of existing connections.

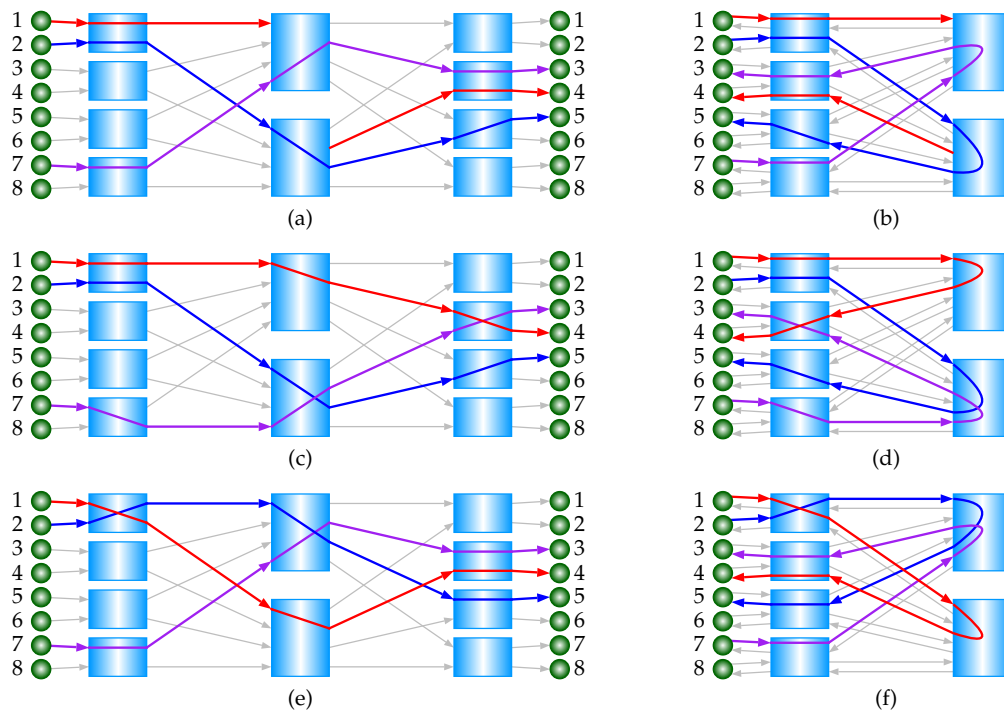


Figure 9. Blocking and rearrangements in a rearrangeably nonblocking Clos network ($m = n = 2$ and $r = n + m = 4$). (a) Two connections ($2 \rightarrow 5$ and $7 \rightarrow 3$) were built. The connection $1 \rightarrow 4$ cannot be built. (b) The case of the folded version of (a). Note that a bidirectional link consists of two oppositely oriented unidirectional links. (c) Two connections ($2 \rightarrow 5$ and $7 \rightarrow 3$) were built. The connection $1 \rightarrow 4$ can also be built by the rearrangements of (a). (d) The case of the folded version of (c). (e) Two connections ($2 \rightarrow 5$ and $7 \rightarrow 3$) were built. The connection $1 \rightarrow 4$ can also be built by the rearrangements of (a). (f) The case of the folded version of (e).

Figure 10(a) shows a 3-stage URNBC network with $n = 3$, $m = n = 3$, and $r = n + m = 6$. It has $N = nr = 18$ compute nodes. A 2-stage IRNBC network, the folded version of the 3-stage URNBC

network with $n = 3$, $m = n = 3$, and $r = n + m = 6$, is shown in Figure 10(b). It uses the equally sized $2n \times 2n = 6 \times 6$ square crossbars.

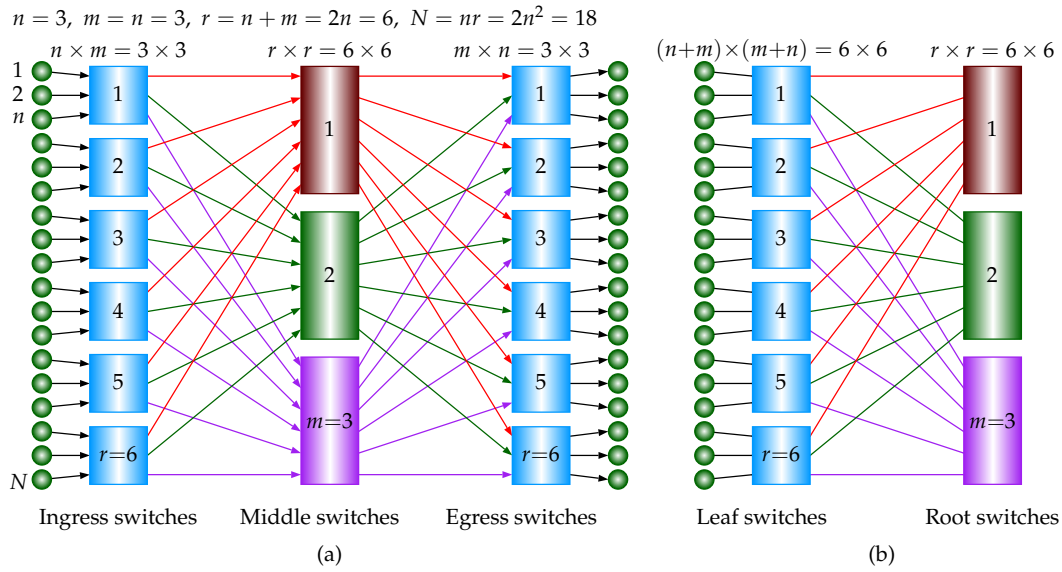


Figure 10. Proposed rearrangeably nonblocking Clos networks ($m = n$ and $r = n + m$). (a) A 3-stage URNBC network with $n = 3$, $m = n = 3$ and $r = n + m = 6$. (b) A 2-stage IRNBC network composed of equally sized $2n \times 2n = 6 \times 6$ square crossbars.

3.2.2. Three-Stage IRNBC Networks

To construct a 3-stage IRNBC network, we first construct a 5-stage URNBC network. By using 3-stage URNBC networks as building blocks, a 5-stage URNBC network can be constructed. For $m = n$, a 3-stage URNBC network has $N = (n + m)n = 2n^2$ compute nodes. As a building block, we remove the compute nodes and consider that the 3-stage URNBC network has $(n + m)n = 2n^2$ inputs and $(n + m)n = 2n^2$ outputs. We arrange m such building blocks in the middle stage. Then, in total, there are $2n^2 \times m = 2n^3$ inputs and $2n^2 \times m = 2n^3$ outputs in the middle stage. Correspondingly, we can arrange the same number of outputs in the ingress stage and the same number of inputs in the egress stage. Let r be the number of switches in the ingress and egress stages, then rm must be equal to $2n^3$. Therefore, we have $r = 2n^3 / m = 2n^2 \times m / m = 2n^2 = (n + m)n$.

In summary, to construct a 5-stage URNBC network, given an n , which is the number of inputs per switch in the ingress stage, there are $r = 2n^2$ switches in the ingress stage, and each switch is an $n \times m$ crossbar with $m = n$. There are m building blocks in the middle stage and each building block is a 3-stage URNBC network with compute nodes removed. There are $r = 2n^2$ switches in the egress stage and each switch is an $m \times n$ crossbar with $m = n$, as shown in Formula (5), where N is the number of compute nodes. The linking method is similar to the 3-stage URNBC network: Each output of a switch in the ingress stage is connected to an input of a different building block in the middle stage. Each output of a building block in the middle stage is connected to an input of a different switch in the egress stage.

$$\begin{cases} m = n \\ r = (n + m)n = 2n^2 \dots \dots (5\text{-stage URNBC}) \\ N = nr = 2n^3 \end{cases} \quad (5)$$

Figure 11 shows a 5-stage URNBC network with $n = 2$, $m = n = 2$, and $r = 2n^2 = 8$. It has $N = nr = 2n^3 = 16$ compute nodes. There are $m = n = 2$ building blocks in the middle stage and each building block is a 3-stage URNBC network with compute nodes removed. The detailed network of a building block is shown at the bottom of the figure.

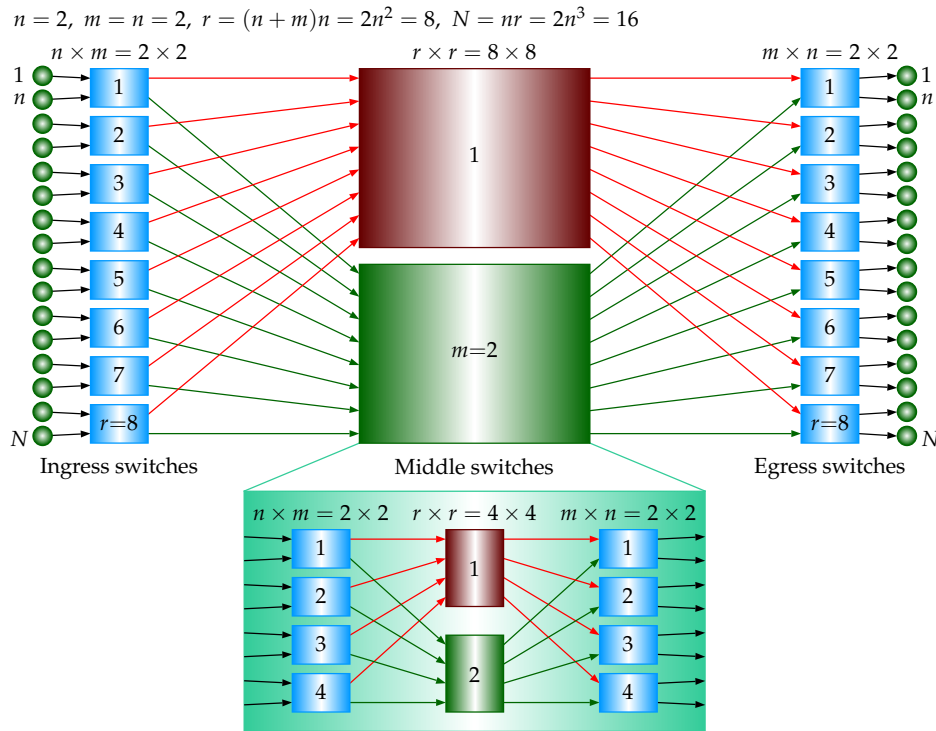


Figure 11. A 5-stage URNBC network with $n = 2$, $m = n = 2$, and $r = 2n^2 = 8$. There are $m = n = 2$ building blocks (3-stage URNBC, Figure 8(a)) in the middle stage.

A 3-stage IRNBC network, the folded version of the 5-stage URNBC network with $n = 2$, $m = n = 2$, and $r = 2n^2 = 8$, is shown in Figure 12. It uses the equally sized $2n \times 2n = 4 \times 4$ square crossbars for all switches. Two 2-stage IRNBC networks (Figure 8(b)) are shown in the two switch columns on the right.

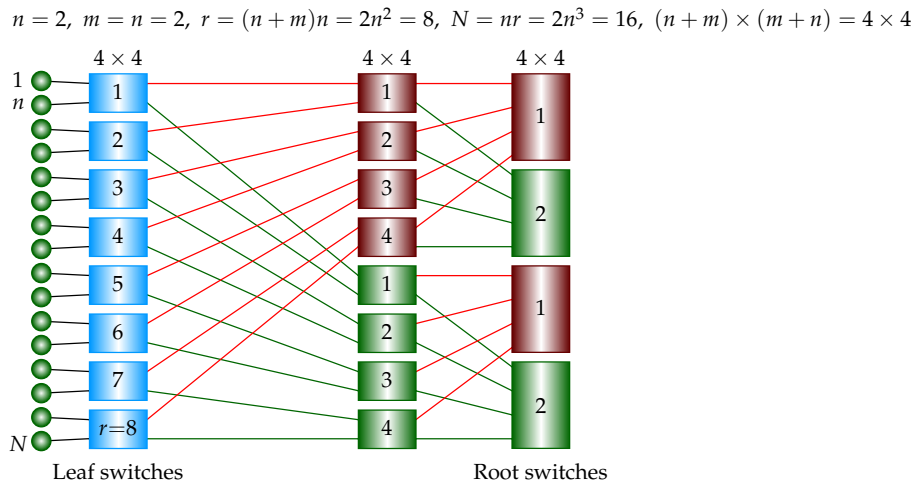


Figure 12. A 3-stage IRNBC network with $n = 2$ composed of equally sized $2n \times 2n = 4 \times 4$ square crossbars (folded version of Figure 11).

3.2.3. Four-Stage IRNBC Networks

To construct a 4-stage IRNBC network, we first construct a 7-stage URNBC network. Similarly, by using 5-stage URNBC networks as building blocks, a 7-stage URNBC network can be constructed. In summary, to construct a 7-stage URNBC network, we determine m and r as shown in Formula (6), where N is the number of compute nodes. The linking method is similar to the 5-stage URNBC network.

$$\begin{cases} m = n \\ r = (n + m)n^2 = 2n^3 \dots \dots (7\text{-stage URNBC}) \\ N = nr = 2n^4 \end{cases} \quad (6)$$

Figure 13 shows a 7-stage URNBC network with $n = 2$, $m = n = 2$, and $r = 2n^3 = 16$. It has $N = nr = 2n^4 = 32$ compute nodes. There are $m = n = 2$ building blocks in the middle stage and each building block is a 5-stage URNBC network with compute nodes removed. The detailed networks of building blocks are shown at the bottom of the figure.

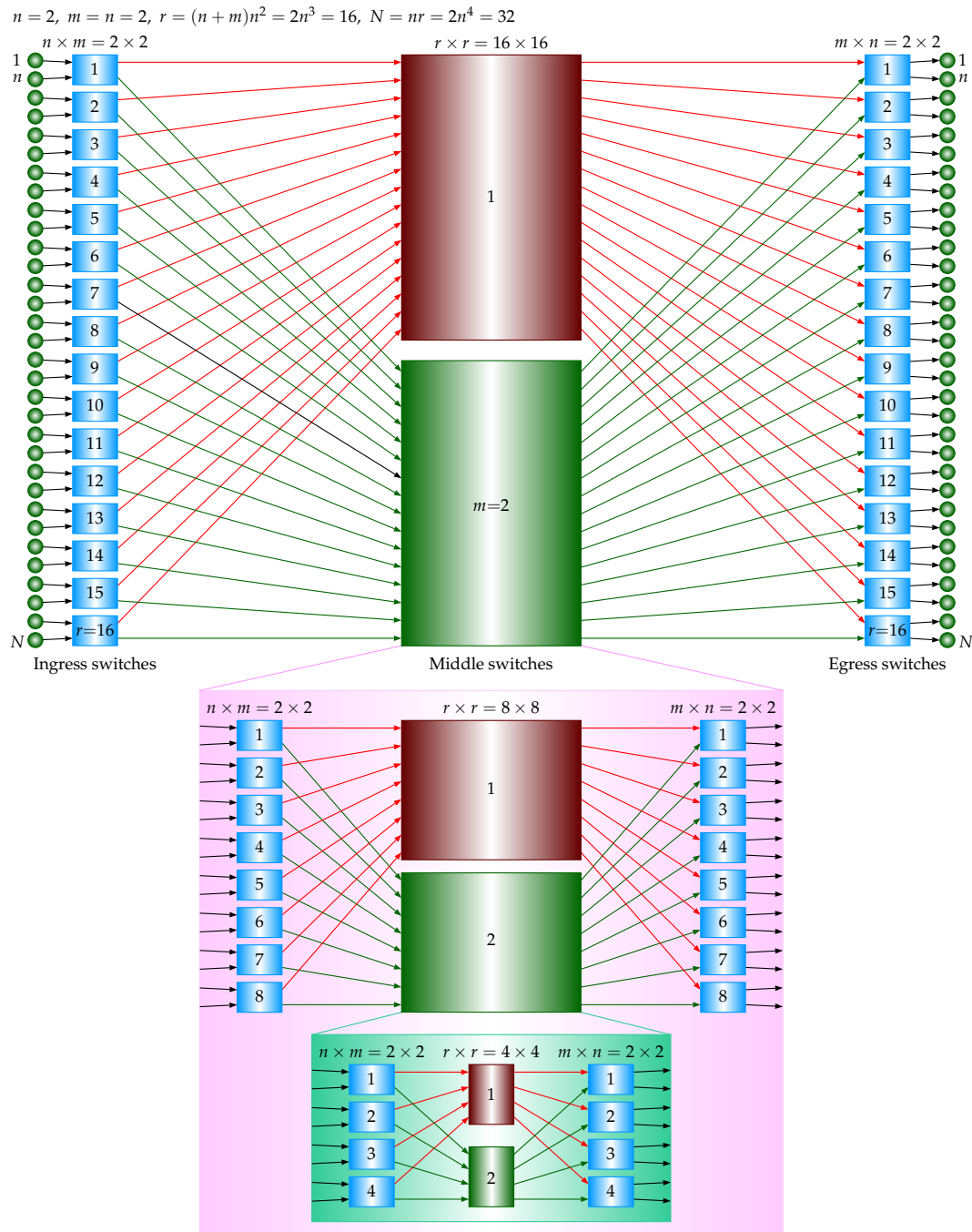


Figure 13. A 7-stage URNBC network with $n = 2$, $m = n = 2$, and $r = 2n^3 = 16$. There are $m = n = 2$ building blocks (5-stage URNBC, Figure 11) in the middle stage.

A 4-stage IRNBC network, the folded version of the 7-stage URNBC network with $n = 2$, $m = n = 2$, and $r = 2n^3 = 16$, is shown in Figure 14. It uses the equally sized $2n \times 2n = 4 \times 4$ square crossbars for all switches. Two 3-stage IRNBC networks (Figure 12) are shown in the three switch columns on the right.

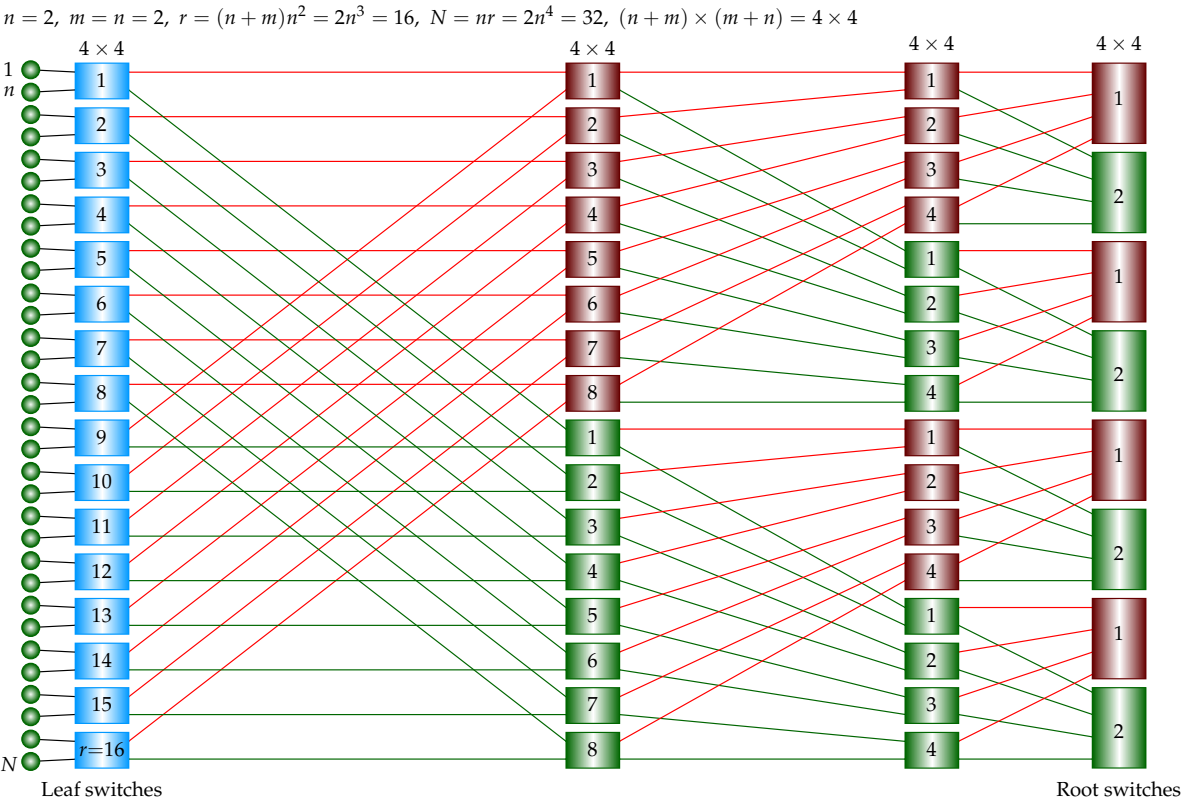


Figure 14. A 4-stage IRNBC network with $n = 2$ composed of equally sized $2n \times 2n = 4 \times 4$ square crossbars (folded version of Figure 13).

Table 3 lists the numbers of compute nodes and switches of 2-, 3-, and 4-stage IRNBC networks. Let s be the number of stages. Then the number of compute nodes is $N = 2n^s$ and the number of switches is $(2s - 1)n^{s-1}$. The crossbar size is listed in the right column.

Table 3. The numbers of compute nodes and switches in the IRNBC networks.

n	Number of compute nodes			Number of switches			Crossbar
	2-stage	3-stage	4-stage	2-stage	3-stage	4-stage	
2	8	16	32	6	20	56	4×4
3	18	54	162	9	45	189	6×6
4	32	128	512	12	80	448	8×8
5	50	250	1250	15	125	875	10×10
6	72	432	2592	18	180	1512	12×12
7	98	686	4802	21	245	2401	14×14
8	128	1024	8192	24	320	3584	16×16
9	162	1458	13,122	27	405	5103	18×18
10	200	2000	20,000	30	500	7000	20×20
11	242	2662	29,282	33	605	9317	22×22
12	288	3456	41,472	36	720	12,096	24×24
13	338	4394	57,122	39	845	15,379	26×26
14	392	5488	76,832	42	980	19,208	28×28
15	450	6750	101,250	45	1125	23,625	30×30

4. Cost Evaluations

This section evaluates the hardware cost for USNBC, ISNBC, URNBC, and IRNBC networks, from the perspective of switch crosspoints, and compares them to the corresponding traditional Clos networks. That is, we evaluate the cost of the 24 networks listed in Table 4.

Table 4. Nonblocking Clos networks for cost evaluations.

Stric or Rearr	Unidir or Bidir	Network	Stage, Unfolded or Folded
Strictly nonblocking	Unidirectional	USNBC	3-, 5-, and 7-stage Clos networks
		Traditional	3-, 5-, and 7-stage Clos networks
	Bidirectional	ISNBC	2-, 3-, and 4-stage folded Clos networks
		Traditional	2-, 3-, and 4-stage folded Clos networks
Rearrangeably nonblocking	Unidirectional	URNBC	3-, 5-, and 7-stage Clos networks
		Traditional	3-, 5-, and 7-stage Clos networks
	Bidirectional	IRNBC	2-, 3-, and 4-stage folded Clos networks
		Traditional	2-, 3-, and 4-stage folded Clos networks

4.1. Cost Evaluations of Strictly Nonblocking Clos Networks

In this subsection, we investigate the crosspoint ratios for Unidirectional Strictly Nonblocking Clos (USNBC) networks and Identical Strictly Nonblocking Folded Clos (ISNBC) networks relative to a single crossbar. These ratios are compared to the corresponding traditional Clos networks.

4.1.1. Cost Evaluations of USNBC Networks

An $n \times m$ crossbar (n inputs and m outputs) has nm crosspoints. In our strictly nonblocking Clos networks, we have $m = 2n$. Referring to Figures 1(a), in a 3-stage USNBC network, there are $r = n + m = 3n$ switches in the ingress stage and each switch is an $n \times m = n \times 2n$ crossbar. There are m switches in the middle stage and each switch is an $r \times r = 3n \times 3n$ crossbar. There are $r = m + n = 3n$ switches in the egress stage and each switch is an $m \times n = 2n \times n$ crossbar. The number of total crosspoints is $nm \times r + r^2 \times m + mn \times r = 2n^2 \times 3n + 9n^2 \times 2n + 2n^2 \times 3n = 30n^3$. There are $N = nr = 3n^2$ inputs and $N = 3n^2$ outputs. If we use a single $N \times N$ crossbar, it requires $N \times N = 9n^4$ crosspoints. The crosspoint ratio of the 3-stage USNBC network to the single crossbar is $30n^3/(9n^4) = 10/(3n)$ that is less than 1 if $n \geq 4$. For example, when $n = 4$, the 3-stage USNBC network requires $30n^3 = 1920$ crosspoints, less than $9n^4 = 2304$ crosspoints in the single crossbar's implementation. In contrast, a traditional strictly nonblocking Clos network requires $n \geq 6$, as mentioned in Section 2.

Referring to Figures 4, in the 5-stage case, there are $2n^2 \times 3n^2$ crosspoints in the ingress stage, there are $30n^3 \times 2n$ crosspoints in the middle stage, and there are $2n^2 \times 3n^2$ crosspoints in the egress stage, where $30n^3$ is the number of crosspoints in a 3-stage USNBC network, as derived above. The total number of the crosspoints is $6n^4 + 60n^4 + 6n^4 = 72n^4$. There are $N = nr = 3n^3$ inputs and $3n^3$ outputs. A single $N \times N$ crossbar requires $9n^6$ crosspoints. The crosspoint ratio of the 5-stage USNBC network to the single crossbar is $72n^4/(9n^6) = 24/(3n^2)$ which is less than 1 if $n \geq 3$.

Referring to Figures 6, in the 7-stage case, there are $2n^2 \times 3n^3$ crosspoints in the ingress stage, there are $72n^4 \times 2n$ crosspoints in the middle stage, and there are $2n^2 \times 3n^3$ crosspoints in the egress stage, where $72n^4$ is the number of crosspoints in a 5-stage USNBC network, as derived above. The total number of the crosspoints is $6n^5 + 144n^5 + 6n^5 = 156n^5$. There are $N = nr = 3n^4$ inputs and $N = 3n^4$ outputs. A single $N \times N$ crossbar requires $9n^8$ crosspoints. The crosspoint ratio of the 7-stage USNBC network to the single crossbar is $156n^5/(9n^8) = 52/(3n^3)$ which is less than 1 if $n \geq 3$.

Now we examine the number of crosspoints for the traditional unidirectional strictly nonblocking Clos network [1]. The 3-stage traditional unidirectional strictly nonblocking Clos network has n switches in the ingress stage, $m = 2n - 1$ switches in the middle stage, and n switches in the egress stage.

An ingress stage switch is an $n \times m$ crossbar, a middle stage switch is an $n \times n$ crossbar, and an egress stage switch is an $m \times n$ crossbar. Then, the total number of crosspoints is $nm \times n + n^2 \times m + mn \times n = 3n^2m = 3n^2(2n - 1)$. The total number of compute nodes is $N = n^2$. If we use a single $N \times N$ crossbar, it requires n^4 crosspoints. The crosspoint ratio of the 3-stage traditional unidirectional strictly nonblocking Clos network to the single crossbar is $3n^2(2n - 1)/n^4 = 3(2n - 1)/n^2$. To guarantee that the ratio is less than 1, $n \geq 6$ is needed.

Consider the case of 5-stage. There are $nm \times n^2$ crosspoints in the ingress stage, there are $3n^2(2n - 1) \times m$ crosspoints in the middle stage, and there are $mn \times n^2$ crosspoints in the egress stage, where $3n^2(2n - 1)$ is the number of crosspoints in a 3-stage traditional unidirectional strictly nonblocking Clos network, as derived above. The total number of crosspoints is $nm \times n^2 + 3n^2(2n - 1) \times m + mn \times n^2 = n^3(2n - 1) + 3n^2(2n - 1)(2n - 1) + n^3(2n - 1) = n^2(8n - 3)(2n - 1)$. The total number of compute nodes is $N = n^3$. A single $N \times N$ crossbar requires n^6 crosspoints. The crosspoint ratio of the 5-stage traditional unidirectional strictly nonblocking Clos network to the single crossbar is $n^2(8n - 3)(2n - 1)/n^6 = (8n - 3)(2n - 1)/n^4$. To guarantee that the ratio is less than 1, $n \geq 4$ is needed.

Consider the case of 7-stage. There are $nm \times n^3$ crosspoints in the ingress stage, there are $n^2(8n - 3)(2n - 1) \times m$ crosspoints in the middle stage, and there are $mn \times n^3$ crosspoints in the egress stage, where $n^2(8n - 3)(2n - 1)$ is the number of crosspoints in a 5-stage traditional unidirectional strictly nonblocking Clos network, as derived above. The total number of crosspoints is $nm \times n^3 + n^2(8n - 3)(2n - 1) \times m + mn \times n^3 = n^4(2n - 1) + n^2(8n - 3)(2n - 1)(2n - 1) + n^4(2n - 1) = n^2(18n^2 - 14n + 3)(2n - 1)$. The total number of compute nodes is $N = n^4$. A single $N \times N$ crossbar requires n^8 crosspoints. The crosspoint ratio of the 7-stage traditional unidirectional strictly nonblocking Clos network to the single crossbar is $n^2(18n^2 - 14n + 3)(2n - 1)/n^8 = (18n^2 - 14n + 3)(2n - 1)/n^6$. To guarantee that the ratio is less than 1, $n \geq 4$ is needed.

We summarize the crosspoint ratio to the single crossbar for the traditional unidirectional strictly nonblocking Clos networks and USNBC networks in Table 5.

Table 5. Crosspoint ratio for unidirectional strictly nonblocking Clos networks.

Network	3-stage	5-stage	7-stage
Traditional	$3(2n-1)/n^2$	$(8n-3)(2n-1)/n^4$	$(18n^2-14n+3)(2n-1)/n^6$
USNBC	$10/(3n)$	$24/(3n^2)$	$52/(3n^3)$

Figure 15 plots the crosspoint ratio to the single crossbar for the unidirectional strictly nonblocking Clos networks, showing that USNBC networks have a lower crosspoint cost than traditional strictly nonblocking Clos networks.

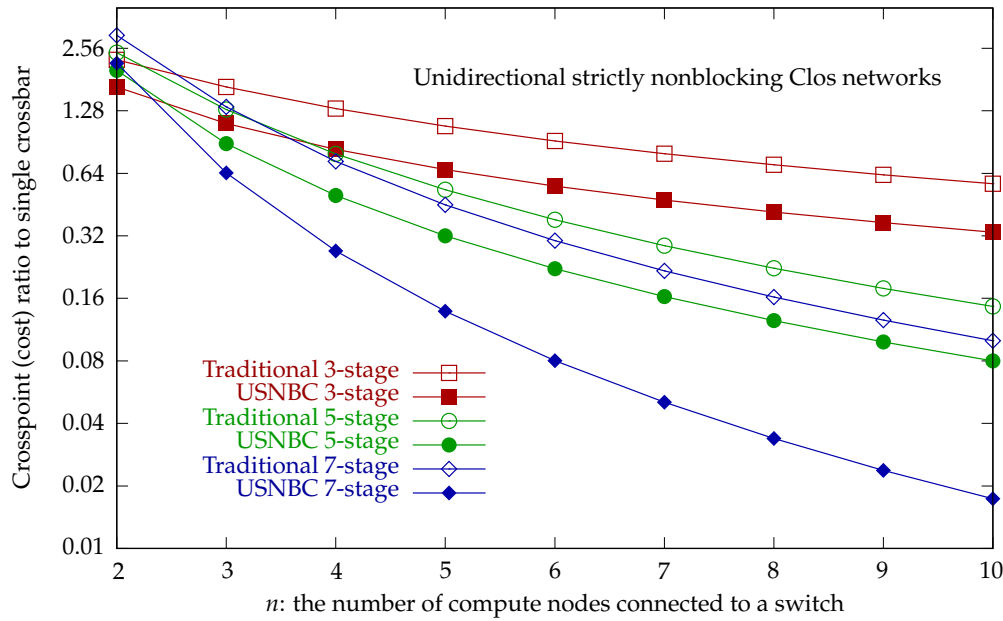


Figure 15. Crosspoint ratio to the single crossbar in unidirectional strictly nonblocking Clos networks.

4.1.2. Cost Evaluations of ISNBC Networks

Now we examine the number of crosspoints for the ISNBC network that uses the equally sized square crossbar of $(n + m) \times (m + n) = 3n \times 3n$ for $m = 2n$. Referring to Figures 1(b), the ISNBC network based on the 3-stage USNBC network has two stages. There are $r = n + m = 3n$ leaf switches and m root switches. The total number of switches is $r + m = 5n$ and each switch is a square $(n + m) \times (m + n) = 3n \times 3n$ crossbar. Then, the total number of crosspoints is $3n \times 3n \times 5n = 45n^3$. The total number of compute nodes is $N = nr = 3n^2$. A single $N \times N$ crossbar requires $9n^4$ crosspoints. The crosspoint ratio of the 2-stage ISNBC network to the single crossbar is $45n^3 / (9n^4) = 5/n$. To guarantee that the ratio is less than 1, $n \geq 6$ is needed.

Consider the 3-stage ISNBC network. Referring to Figure 5, there are $r = (n + m)n = 3n^2$ switches in the leaf stage, and there are m building blocks and each building block is a 2-stage ISNBC network whose number of switches is $5n$, as derived above. The total number of switches is $3n^2 + 5n \times 2n = 13n^2$ and each switch is a square $(n + m) \times (m + n) = 3n \times 3n$ crossbar. Then, the total number of crosspoints is $3n \times 3n \times 13n^2 = 117n^4$. The total number of compute nodes is $N = nr = 3n^3$. A single $N \times N$ crossbar requires $9n^6$ crosspoints. The crosspoint ratio of the 3-stage ISNBC network to the single crossbar is $117n^4 / (9n^6) = 13/n^2$. To guarantee that the ratio is less than 1, $n \geq 4$ is needed.

Consider the 4-stage ISNBC network. Referring to Figure 7, there are $r = (n + m)n^2 = 3n^3$ switches in the leaf stage, and there are m building blocks and each building block is a 3-stage ISNBC network whose number of switches is $13n^2$, as derived above. The total number of switches is $3n^3 + 13n^2 \times 2n = 29n^3$ and each switch is a square $(n + m) \times (m + n) = 3n \times 3n$ crossbar. Then, the total number of crosspoints is $3n \times 3n \times 29n^3 = 261n^5$. The total number of compute nodes is $N = nr = 3n^4$. A single $N \times N$ crossbar requires $9n^8$ crosspoints. The crosspoint ratio of the 4-stage ISNBC network to the single crossbar is $261n^5 / (9n^8) = 29/n^3$. To guarantee that the ratio is less than 1, $n \geq 4$ is needed.

Table 6 lists the crosspoints of ISNBC networks. The “Crossbar” column shows the number of crosspoints in the single crossbar. The “ISNBC” column shows the number of crosspoints in the ISNBC network. The crosspoint number in blue color is better (smaller) than the number of the single crossbar.

Table 6. The number of crosspoints in the ISNBC networks.

n	2-stage			3-stage			4-stage		
	Node	Crossbar	ISNBC	Node	Crossbar	ISNBC	Node	Crossbar	ISNBC
2	12	144	360	24	576	1872	48	2304	8352
3	27	729	1215	81	6561	9477	243	59,049	63,423
4	48	2304	2880	192	36,864	29,952	768	589,824	267,264
5	75	5625	5625	375	140,625	73,125	1875	3,515,625	815,625
6	108	11,664	9720	648	419,904	151,632	3888	15,116,544	2,029,536
7	147	21,609	15,435	1029	1,058,841	280,917	7203	51,883,209	4,386,627
8	192	36,864	23,040	1536	2,359,296	479,232	12,288	150,994,944	8,552,448
9	243	59,049	32,805	2187	4,782,969	767,637	19,683	387,420,489	15,411,789
10	300	90,000	45,000	3000	9,000,000	1,170,000	30,000	900,000,000	26,100,000

Now we examine the number of crosspoints for the traditional strictly nonblocking folded Clos network that uses crossbars of different sizes. The 2-stage traditional strictly nonblocking folded Clos network has n switches in the leaf stage and $m = 2n - 1$ switches in the root stage. A leaf switch is an $(n + m) \times (m + n) = (3n - 1) \times (3n - 1)$ crossbar. A root switch is an $n \times n$ crossbar. Then, the total number of crosspoints is $(3n - 1)^2 \times n + n^2 \times m = n(11n^2 - 7n + 1)$. The total number of compute nodes is $N = n^2$. A single $N \times N$ crossbar requires n^4 crosspoints. The crosspoint ratio of the 2-stage traditional strictly nonblocking folded Clos network to the single crossbar is $n(11n^2 - 7n + 1)/n^4 = (11n^2 - 7n + 1)/n^3$. To guarantee that the ratio is less than 1, $n \geq 11$ is needed.

Consider the case of 3-stage. There are n^2 switches in the leaf stage and each switch is an $(n + m) \times (m + n) = (3n - 1) \times (3n - 1)$ crossbar. There are $m = 2n - 1$ building blocks and each building block has $n(11n^2 - 7n + 1)$ crosspoints, as derived above. Then the total number of crosspoints is $(3n - 1)^2 \times n^2 + n(11n^2 - 7n + 1) \times (2n - 1) = n(31n^3 - 31n^2 + 10n - 1)$. The total number of compute nodes is $N = n^3$. A single $N \times N$ crossbar requires n^6 crosspoints. The crosspoint ratio of the 3-stage traditional strictly nonblocking folded Clos network to the single crossbar is $n(31n^3 - 31n^2 + 10n - 1)/n^6 = (31n^3 - 31n^2 + 10n - 1)/n^5$. To guarantee that the ratio is less than 1, $n \geq 6$ is needed.

Consider the case of 4-stage. There are n^3 switches in the leaf stage and each switch is an $(n + m) \times (m + n) = (3n - 1) \times (3n - 1)$ crossbar. There are $m = 2n - 1$ building blocks and each building block has $n(31n^3 - 31n^2 + 10n - 1)$ crosspoints, as derived above. Then the total number of crosspoints is $(3n - 1)^2 \times n^3 + n(31n^3 - 31n^2 + 10n - 1) \times m = n(71n^4 - 99n^3 + 52n^2 - 12n + 1)$. The total number of compute nodes is $N = n^4$. A single $N \times N$ crossbar requires n^8 crosspoints. The crosspoint ratio of the 4-stage traditional strictly nonblocking folded Clos network to the single crossbar is $n(71n^4 - 99n^3 + 52n^2 - 12n + 1)/n^8 = (71n^4 - 99n^3 + 52n^2 - 12n + 1)/n^7$. To guarantee that the ratio is less than 1, $n \geq 4$ is needed.

We summarize the crosspoint ratio to the single crossbar for the traditional strictly nonblocking folded Clos networks and ISNBC networks in Table 7. The general formula for calculating the ISNBC crosspoint ratio to the single crossbar is $(2^{s+1} - 3)/n^{s-1}$, where s is the number of stages, and the number of compute nodes is $N = 3n^s$.

Table 7. Crosspoint ratio for strictly nonblocking folded Clos networks.

Network	2-stage	3-stage	4-stage
Traditional	$(11n^2 - 7n + 1)/n^3$	$(31n^3 - 31n^2 + 10n - 1)/n^5$	$(71n^4 - 99n^3 + 52n^2 - 12n + 1)/n^7$
ISNBC	$5/n$	$13/n^2$	$29/n^3$

Figure 16 plots the crosspoint ratio to the single crossbar for the strictly nonblocking folded Clos networks, showing that the ISNBC networks have a lower crosspoint cost than the traditional strictly

nonblocking folded Clos networks. Also note that the ISNBC networks use the equally sized square crossbar for all switches in the network.

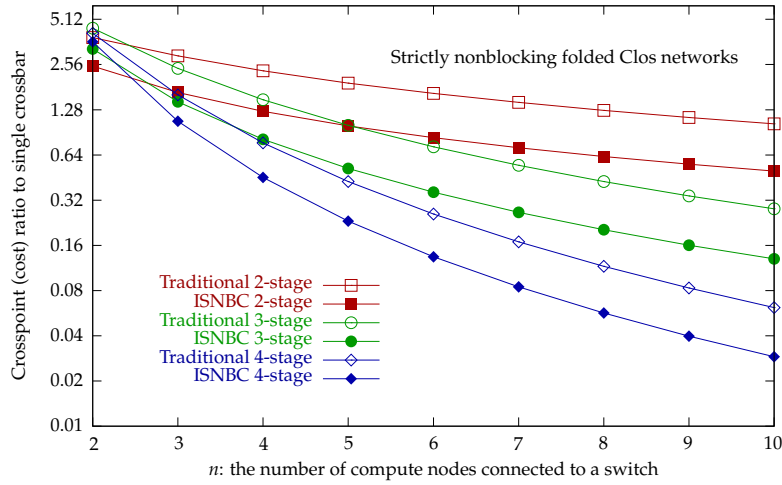


Figure 16. Crosspoint ratio to the single crossbar in strictly nonblocking folded Clos networks.

4.2. Cost Evaluations of Rearrangeably Nonblocking Clos Networks

In this subsection, we investigate the crosspoint ratios for the Unidirectional Rearrangeably NonBlocking Clos (URNBC) networks and the Identical Rearrangeably NonBlocking folded Clos (IRNBC) networks relative to a single crossbar. These ratios are compared to the corresponding traditional Clos networks. It is unfair to compare a rearrangeably nonblocking Clos network to a single crossbar since a single crossbar is a strictly nonblocking network. The reason we present the ratios here is to allow us to see the difference between the proposed network and the traditional network.

4.2.1. Cost Evaluations of URNBC Networks

As described in the previous section, in URNBC networks, we let $m = n$ and $r = n + m = 2n$. Referring to Figures 8(a), in a 3-stage URNBC network, there are $r = n + m = 2n$ switches in the ingress stage and each switch is an $n \times m$ crossbar, there are m switches in the middle stage and each switch is an $r \times r$ crossbar, and there are $r = n + m = 2n$ switches in the egress stage and each switch is an $m \times n$ crossbar. The total number of crosspoints is $nm \times r + r^2 \times m + mn \times r = 2n^3 + 4n^3 + 2n^3 = 8n^3$. There are $N = nr = 2n^2$ compute nodes. If we use a single $N \times N$ crossbar, it requires $N \times N = 4n^4$ crosspoints. The crosspoint ratio of the 3-stage URNBC network to the single crossbar is $8n^3 / (4n^4) = 2/n$.

Referring to Figure 11, in a 5-stage URNBC network, there are $r = (n + m)n = 2n^2$ switches in the ingress stage and each switch is an $n \times m$ crossbar, there are m 3-stage URNBC networks and each URNBC network has $8n^3$ crosspoints, as derived above, and there are $r = (n + m)n = 2n^2$ switches in the egress stage and each switch is an $m \times n$ crossbar. The total number of crosspoints is $nm \times 2n^2 + 8n^3 \times m + mn \times 2n^2 = 2n^4 + 8n^4 + 2n^4 = 12n^4$. There are $N = nr = 2n^3$ compute nodes. A single $N \times N$ crossbar requires $4n^6$ crosspoints. The crosspoint ratio of the 5-stage URNBC network to the single crossbar is $12n^4 / (4n^6) = 3/n^2$.

Referring to Figure 13, in a 7-stage URNBC network, there are $r = (n + m)n^2 = 2n^3$ switches in the ingress stage and each switch is an $n \times m$ crossbar, there are m 5-stage URNBC networks and each URNBC network has $12n^4$ crosspoints, as derived above, and there are $r = (n + m)n^2 = 2n^3$ switches in the egress stage and each switch is an $m \times n$ crossbar. The total number of crosspoints is $nm \times 2n^3 + 12n^4 \times m + mn \times 2n^3 = 2n^5 + 12n^5 + 2n^5 = 16n^5$. There are $N = nr = 2n^4$ compute nodes. A single $N \times N$ crossbar requires $4n^8$ crosspoints. The crosspoint ratio of the 7-stage URNBC network to the single crossbar is $16n^5 / (4n^8) = 4/n^3$.

Now we examine the number of crosspoints for the traditional unidirectional rearrangeably nonblocking Clos network. In a 3-stage traditional unidirectional rearrangeably nonblocking Clos

network, we have $m = n$ and $N = n^2$. The total number of crosspoints is $nm \times n + n^2 \times m + mn \times n = n^3 + n^3 + n^3 = 3n^3$. A single $N \times N$ crossbar requires n^4 crosspoints. Therefore, the crosspoint ratio of the 3-stage traditional unidirectional rearrangeably nonblocking Clos network to the single crossbar is $3n^3/n^4 = 3/n$.

In the case of 5-stage, we have $m = n$ and $N = n^3$. The total number of crosspoints is $nm \times n^2 + 3n^3 \times m + mn \times n^2 = n^4 + 3n^4 + n^4 = 5n^4$, where $3n^3$ is the number of crosspoints in a 3-stage traditional unidirectional rearrangeably nonblocking Clos network, as derived above. A single $N \times N$ crossbar requires n^6 crosspoints. Therefore, the crosspoint ratio of the 5-stage traditional unidirectional rearrangeably nonblocking Clos network to the single crossbar is $5n^4/n^6 = 5/n^2$.

In the case of 7-stage, we have $m = n$ and $N = n^4$. The total number of crosspoints is $nm \times n^3 + 5n^4 \times m + mn \times n^3 = n^5 + 5n^5 + n^5 = 7n^5$, where $5n^4$ is the number of crosspoints in a 5-stage traditional unidirectional rearrangeably nonblocking Clos network, as derived above. A single $N \times N$ crossbar requires n^8 crosspoints. Therefore, the crosspoint ratio of the 7-stage traditional unidirectional rearrangeably nonblocking Clos network to the single crossbar is $7n^5/n^8 = 7/n^3$.

We summarize the crosspoint ratio to the single crossbar for the traditional unidirectional rearrangeably nonblocking Clos networks and URNBC networks in Table 8. The cost ratios for the proposed URNBC networks to the traditional networks are 66.67%, 60.00%, and 57.14% for 3-stage, 5-stage, and 7-stage networks, respectively.

Table 8. Crosspoint ratio for unidirectional rearrangeably nonblocking Clos networks.

Network	3-stage	5-stage	7-stage
Traditional	$3/n$	$5/n^2$	$7/n^3$
URNBC	$2/n$	$3/n^2$	$4/n^3$
URNBC/Traditional	66.67%	60.00%	57.14%

Figure 17 plots the crosspoint ratio to the single crossbar for the unidirectional rearrangeably nonblocking Clos networks, showing that the URNBC networks have a lower crosspoint cost than the traditional rearrangeably nonblocking Clos networks.

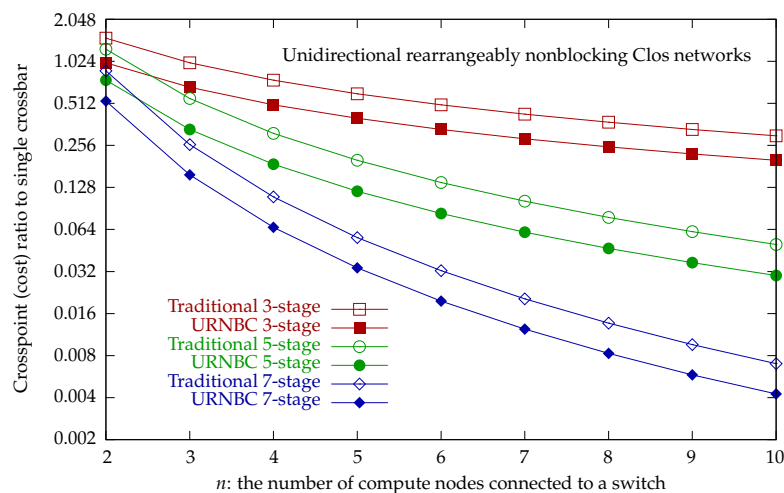


Figure 17. Crosspoint ratio to the single crossbar in unidirectional rearrangeably nonblocking Clos networks.

4.2.2. Cost Evaluations of IRNBC Networks

Now we examine the number of crosspoints for the IRNBC network that uses the equally sized square crossbar of $(n + m) \times (m + n) = 2n \times 2n$ for $m = n$. Referring to Figures 8(b), in a 2-stage IRNBC network, there are $r = n + m = 2n$ switches in the leaf stage and m switches in the root stage. The total number of crosspoints is $(n + m) \times (m + n) \times (2n + m) = 12n^3$. There are $N = nr = 2n^2$

compute nodes. If we use a single $N \times N$ crossbar, it requires $N \times N = 4n^4$ crosspoints. The crosspoint ratio of the 2-stage IRNBC network to the single crossbar is $12n^3/(4n^4) = 3/n$.

Referring to Figure 12, in a 3-stage IRNBC network, the total number of crosspoints is $(n + m) \times (m + n) \times 2n^2 + 12n^3 \times m = 8n^4 + 12n^4 = 20n^4$, where $12n^3$ is the number of crosspoints in a 2-stage IRNBC network, as derived above. There are $N = nr = 2n^3$ compute nodes. A single $N \times N$ crossbar requires $4n^6$ crosspoints. The crosspoint ratio of the 3-stage IRNBC network to the single crossbar is $20n^4/(4n^6) = 5/n^2$.

Referring to Figure 14, in a 4-stage IRNBC network, the total number of crosspoints is $(n + m) \times (m + n) \times 2n^3 + 20n^4 \times m = 8n^5 + 20n^5 = 28n^5$, where $20n^4$ is the number of crosspoints in a 3-stage IRNBC network, as derived above. There are $N = nr = 2n^4$ compute nodes. A single $N \times N$ crossbar requires $4n^8$ crosspoints. The crosspoint ratio of the 4-stage IRNBC network to the single crossbar is $28n^5/(4n^8) = 7/n^3$.

Table 9 lists the crosspoints of IRNBC networks. The “Crossbar” column shows the number of crosspoints in the single crossbar. The “IRNBC” column shows the number of crosspoints in the IRNBC network. The crosspoint numbers in blue color are better (smaller) than the number of the single crossbar.

Table 9. The number of crosspoints in the IRNBC networks.

n	2-stage			3-stage			4-stage		
	Node	Crossbar	IRNBC	Node	Crossbar	IRNBC	Node	Crossbar	IRNBC
2	8	64	96	16	256	320	32	1024	896
3	18	324	324	54	2916	1620	162	26,244	6804
4	32	1024	768	128	16,384	5120	512	262,144	28,672
5	50	2500	1500	250	62,500	12,500	1250	1,562,500	87,500
6	72	5184	2592	432	186,624	25,920	2592	6,718,464	217,728
7	98	9604	4116	686	470,596	48,020	4802	23,059,204	470,596
8	128	16,384	6144	1024	1,048,576	81,920	8192	67,108,864	917,504
9	162	26,244	8748	1458	2,125,764	131,220	13,122	172,186,884	1,653,372
10	200	40,000	12,000	2000	4,000,000	200,000	20,000	400,000,000	2,800,000

Now we examine the number of crosspoints for the traditional rearrangeably nonblocking folded Clos networks with $m = n$. In a 2-stage traditional rearrangeably nonblocking folded Clos network, the total number of crosspoints is $(n + m)(m + n) \times n + n^2 \times m = 4n^3 + n^3 = 5n^3$. There are $N = n^2$ compute nodes. If we use a single $N \times N$ crossbar, it requires $N \times N = n^4$ crosspoints. The crosspoint ratio of the 2-stage traditional rearrangeably nonblocking folded Clos network to the single crossbar is $5n^3/n^4 = 5/n$.

In the case of 3-stage, the total number of crosspoints is $(n + m)(m + n) \times n^2 + 5n^3 \times m = 4n^4 + 5n^4 = 9n^4$, where $5n^3$ is the number of crosspoints in a 2-stage traditional rearrangeably nonblocking folded Clos network, as derived above. There are $N = n^3$ compute nodes. A single $N \times N$ crossbar requires n^6 crosspoints. The crosspoint ratio of the 3-stage traditional rearrangeably nonblocking folded Clos network to the single crossbar is $9n^4/n^6 = 9/n^2$.

In the case of 4-stage, the total number of crosspoints is $(n + m)(m + n) \times n^3 + 9n^4 \times m = 4n^5 + 9n^5 = 13n^5$, where $9n^4$ is the number of crosspoints in a 3-stage traditional rearrangeably nonblocking folded Clos network, as derived above. The crosspoint ratio of the 4-stage traditional rearrangeably nonblocking folded Clos network to the single crossbar is $13n^5/n^8 = 13/n^3$.

We summarize the crosspoint ratio to the single crossbar for the traditional rearrangeably nonblocking folded Clos networks and IRNBC networks in Table 10. The cost ratios for the proposed IRNBC networks to the traditional networks are 60.00%, 55.56%, and 53.85% for 3-stage, 5-stage, and 7-stage networks, respectively. The general formula for calculating the IRNBC crosspoint ratio to the single crossbar is $(2s - 1)/n^{s-1}$, where s is the number of stages, and the number of compute nodes is $N = 2n^s$.

Table 10. Crosspoint ratio for rearrangeably nonblocking folded Clos networks.

Network	2-stage	3-stage	4-stage
Traditional	$5/n$	$9/n^2$	$13/n^3$
IRNBC	$3/n$	$5/n^2$	$7/n^3$
IRNBC/Traditional	60.00%	55.56%	53.85%

Figure 18 plots the crosspoint ratio to the single crossbar for the rearrangeably nonblocking folded Clos networks, showing that the IRNBC networks have a lower crosspoint cost than the traditional rearrangeably nonblocking folded Clos networks.

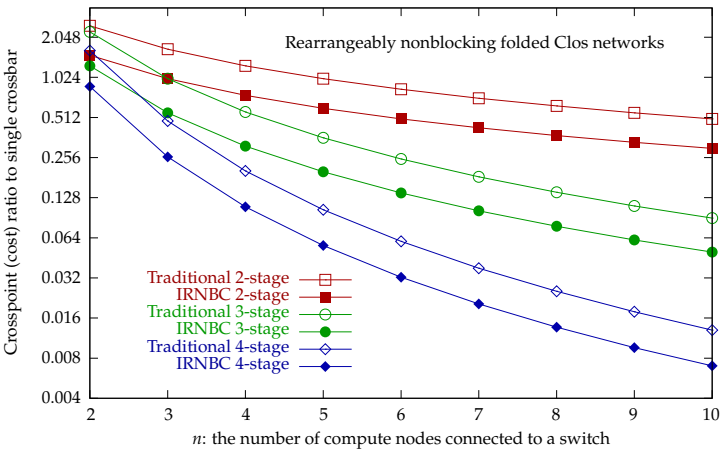


Figure 18. Crosspoint ratio to the single crossbar in rearrangeably nonblocking folded Clos networks.

From the above discussion in this section, we can see that the crosspoint ratios of ISNBC and IRNBC networks are lower than their corresponding traditional folded Clos networks. And, the crosspoint ratio of IRNBC is lower than ISNBC because IRNBC is a rearrangeably nonblocking folded Clos network that requires rearrangements of existing connections to make new connections and ISNBC is a strictly nonblocking folded Clos network that does not require rearrangements of existing connections to make new connections. Figure 19 compares the crosspoint ratios of the ISNBC and IRNBC networks for different numbers of compute nodes. From this figure, the number of stages can be selected for a given number of compute nodes such that the system has a low crosspoint ratio.

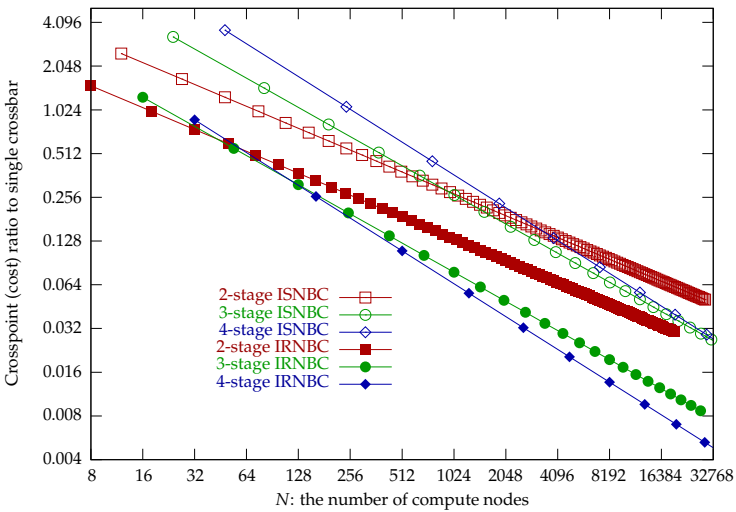


Figure 19. Crosspoint ratios versus the number of compute nodes in the proposed networks.

The proposed ISNBC and IRNBC networks are summarized in Table 11 where the “Node” column shows the number of compute nodes, the “Crossbar” column shows the number of crosspoints in the

single crossbar, the “Crosspoint” column shows the number of crosspoints in the proposed network, and the “Switch” column shows the number of switches in the proposed network. Both networks use square crossbar switches.

Table 11. Summary of the proposed ISNBC and IRNBC networks where n is the number of compute nodes connected to a leaf switch and s is the number of stages.

Network	Node	Crossbar	Crosspoint	Switch	Switch size
ISNBC	$3n^s$	$9n^{2s}$	$9(2^{s+1} - 3)n^{s+1}$	$(2^{s+1} - 3)n^{s-1}$	$3n \times 3n$
IRNBC	$2n^s$	$4n^{2s}$	$4(2s - 1)n^{s+1}$	$(2s - 1)n^{s-1}$	$2n \times 2n$

5. Conclusions

Nowadays the available switches are square crossbars with the same number of input and output ports. We proposed an Identical Strictly NonBlocking folded Clos (ISNBC) network and an Identical Rearrangeably NonBlocking folded Clos (IRNBC) network. Compared with other nonblocking Clos networks, the proposed ISNBC and IRNBC networks

- (1) use the equally sized square crossbars with no unused switch ports,
- (2) can contain any number of stages to increase the system’s scalability,
- (3) can utilize shortcut connections to reduce communication path lengths, and
- (4) have a lower cost ratio in terms of switch crosspoints.

Future work should develop load-balancing adaptive routing algorithms and fault-tolerant routing algorithms for the proposed identical strictly and rearrangeably nonblocking folded Clos networks.

References

1. Clos, C. A study of non-blocking switching networks. *The Bell System Technical Journal* **1953**, *32*, 406–424. <https://doi.org/10.1002/j.1538-7305.1953.tb01433.x>.
2. Leiserson, C.E. Fat-trees: Universal networks for hardware-efficient supercomputing. *IEEE Transactions on Computers* **1985**, *C-34*, 892–901. <https://doi.org/10.1109/TC.1985.6312192>.
3. TOP500. *Supercomputer Sites*; <http://top500.org/>, 2024.
4. Abts, D.; Kim, J. *High Performance Datacenter Networks: Architectures, Algorithms, and Opportunities*; Morgan and Claypool, 2011. <https://doi.org/10.2200/S00341ED1V01Y201103CAC014>.
5. Petrini, F.; Vanneschi, M. k-ary n-trees: high performance networks for massively parallel architectures. In *Proceedings of the Proceedings 11th International Parallel Processing Symposium*, 1997, pp. 87–93. <https://doi.org/10.1109/IPPS.1997.580853>.
6. Dally, W.J.; Towles, B.P. *Principles and Practices of Interconnection Networks*; The Morgan Kaufmann Series in Computer Architecture and Design, Elsevier Science, 2004. Available at <https://books.google.co.jp/books?id=oOqpcB5191sC>.
7. Beneš, V.E. Permutation groups, complexes, and rearrangeable connecting networks. *The Bell System Technical Journal* **1964**, *43*, 1619–1640. <https://doi.org/10.1002/j.1538-7305.1964.tb04102.x>.
8. Al-Fares, M.; Loukissas, A.; Vahdat, A. A scalable, commodity data center network architecture. *ACM SIGCOMM Computer Communication Review* **2008**, *38*, 63–74. <https://doi.org/10.1145/1402946.1402967>.
9. Li, Y.; Chu, W. MiKANT: A Mirrored K-Ary N-Tree for Reducing Hardware Cost and Packet Latency of Fat-Tree and Clos Networks. In *Proceedings of the The 18th IEEE International Conference on Scalable Computing and Communications*, 2018, pp. 1643–1650. <https://doi.org/10.1109/SmartWorld.2018.00281>.
10. Li, Y.; Chu, W. Fault Tolerance and Packet Latency of Peer Fat-Trees. In *Proceedings of the Parallel and Distributed Computing, Applications and Technologies*, Cham, 2023; pp. 413–425. https://doi.org/10.1007/978-3-031-29927-8_32.
11. Mano, T.; Inoue, T.; Mizutani, K.; Akashi, O. Redesigning the Nonblocking Clos Network to Increase Its Capacity. *IEEE Transactions on Network and Service Management* **2023**, *20*, 2558–2574. <https://doi.org/10.1109/TNSM.2022.3228909>.
12. Taka, H.; Inoue, T.; Oki, E. Twisted and Folded Clos-Network Design Model With Two-Step Blocking Probability Guarantee. *IEEE Networking Letters* **2024**, *6*, 60–64. <https://doi.org/10.1109/LNET.2023.3322172>.

13. Taka, H.; Inoue, T.; Oki, E. Design model of a twisted and folded Clos network with multi-step grouped intermediate switches guaranteeing admissible blocking probability. *Journal of Optical Communications and Networking* **2024**, *16*, 328–341. <https://doi.org/10.1364/JOCN.513898>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.