Study of Wide-sense Nonblocking Switching Networks from the approach of upper ideals

BY
Kwok Siu Yu

A THESIS
SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF PHILOSOPHY
DIVISION OF INFORMATION ENGINEERING
THE CHINESE UNIVERSITY OF HONG KONG
JUNE 2000

Acknowledgement

It is a pleasure to express my heartfelt thanks to Prof. S.-Y. Robert Li for his helpful guidance and continued support.

簡介

交換網絡是"廣義無阻塞"如果有一種無阻塞選擇路向程式的存在。而對於一個 3 級的 Clos 交換網絡 $[n\times m, r\times r, m\times n]$ 而言， 如果 $m \geq 2n-1$， 那麼這交換網絡則是"嚴格無阻塞"， 這是比廣義無阻塞更強的性質。 Benes [1] 於 1965 年証明，當 $m \geq \lfloor 3n/2 \rfloor$，則交換網絡$[n\times m, 2\times 2, m\times n]$是廣義無阻塞。 這也証明了廣義無阻塞並不等同於嚴格無阻塞。 而這也帶來了一個問題， 就是究竟除了 Benes 所發現的之外， 還有沒有其他廣義無阻塞但又不是嚴格無阻塞交換網絡。對此問題我們給予一個肯定的答案，就是 $[6\times 10, 3\times 3, 10\times 6]$是廣義無阻塞。 在部份的証明過程當中， 我們使用了電腦搜查作為輔助。

在另一方面， 我們也証明了交換網絡 $[5\times 8, 3\times 3, 8\times 5]$不是廣義無阻塞。 而我們更証明如果在交換網絡 $[n\times m, r\times r, m\times n]$之上存在一種無阻塞的"擠滿路向程式"，則 $m\geq \lfloor 15n/8 \rfloor$。 從$[6\times 10, 3\times 3, 10\times 6]$之例可見:廣義無阻塞交換網絡並不等同於它所用的路向程式是擠滿路向程式。

Abstract

A switching network is said to be *wide-sense nonblocking* if there is a nonblocking algorithm for route selection. The 3-stage Clos network $[n\times m,\ r\times r,\ m\times n]$ is strictly nonblocking when $m \geq 2n-1$ and hence is also wide-sense nonblocking. In 1965, Benes [1] proved that the network $[n\times m,\ 2\times 2,\ m\times n]$ is wide-sense nonblocking when $m \geq \lfloor 3n/2 \rfloor$. This identifies a family of 3-stage networks that are wide-sense nonblocking but not strictly nonblocking. It also raised the question on the existence of any wide-sense nonblocking network $[n\times m,\ r\times r,\ m\times n]$, $r > 2$, that is not strictly nonblocking. We answer this old question affirmatively with an algorithm over the network $[6\times 10,\ 3\times 3,\ 10\times 6]$. The approach pertains to the concept of *upper ideals* in lattice theory. Part of the justification of the algorithm is by exhaustive computer search.

On the other hand, the first smallest 3-stage network with the wide-sense nonblocking property undetermined previously is $[5\times 8,\ 3\times 3,\ 8\times 5]$. From the same upper-ideal approach, we find this network not wide-sense

nonblocking. We also prove that, if there exists a nonblocking packing algorithm over $[n\times m,\ r\times r,\ m\times n]$, then $m \geq \lfloor \frac{15n}{8} \rfloor$. In view of the network $[6\times 10,\ 3\times 3,\ 10\times 6]$, a wide-sense nonblocking network does not necessarily admit a packing algorithm.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background of switching networks

Without switching concept, every user can only communicate with each other by a point-to-point network. The simplest case is there are only two users. They only need one single link for the communication. The situation is shown as Figure 1. However, the situation becomes more complex if the number of users (N) increases. Figure 2 shows six links are needed when $N = 4$. Therefore, the number of links will be increased exponentially if N increases.

Figure 1: A point-to-point network for $N = 2$

In order to make the network simple, reduce the number of connective links, and lower the construction cost, another network is proposed. It is the star network, which is shown on Figure 3 . The intermediate node, which is called *switch*, is a main key to reduce the total number of connective links. In the early beginning, the switch is operated manually by plugs.

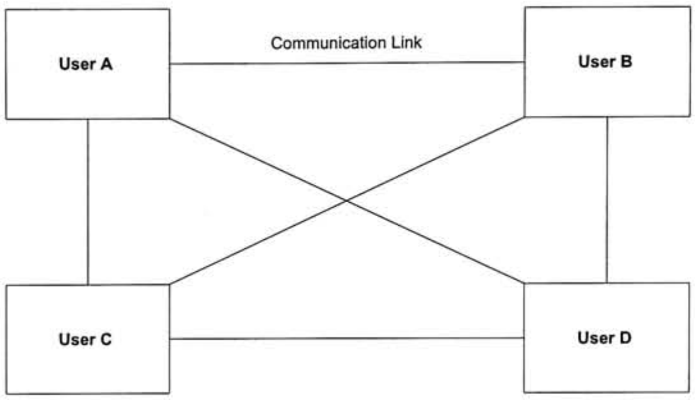

Figure 2: A point-to-point network for $N = 4$

Therefore, the switch is inefficient and insecure. The inefficiency is due to the slow manual switching and insecurity in eavesdropping on conversation by the switch operator.

By the rapid development of electronics, the automated mechanical or electronic switch relays shown in Figure 4 replace the manual switches. However, for large N, the hardware implementation could be difficult, as a consequence of the large number of cross points and the rather extensive connections required.

To solve this problem, multi-interconnection switching networks are used. The idea is to use simple and smaller switches to construct a bigger switch. The disadvantage of this design is the network may lead to *blocking.*

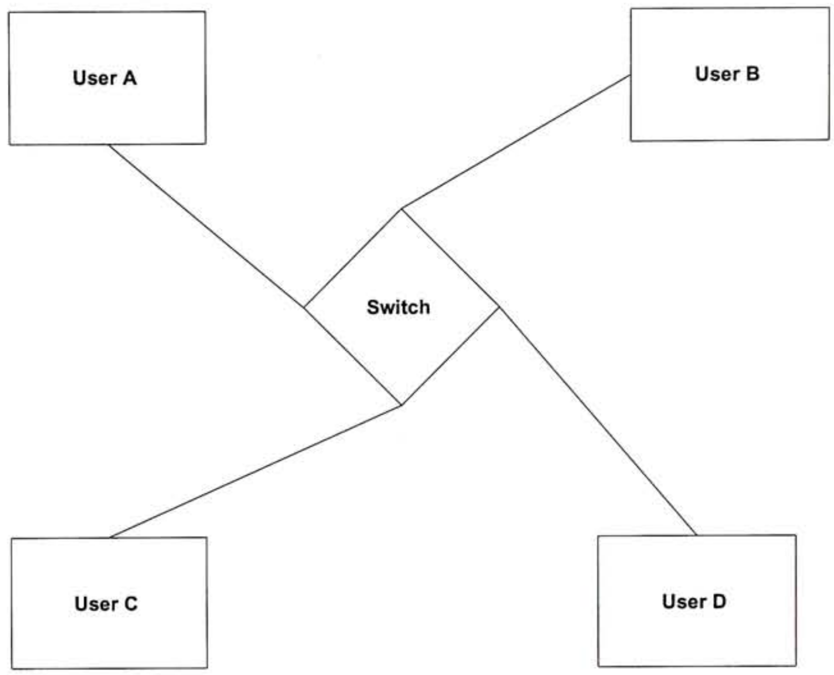

Figure 3: A star network for $N = 4$

Blocking [6] is defined as the network has no ability to complete a connection, even when the input and the output for the connection to be established are idle. Therefore, switches can be divided into two categories. They are blocking and nonblocking switches.

For a two-stage network, establishing a connection between an input and an output is to connect the first-stage switch and the second-stage switch by the one and only one connection link which is shown in Figure 5. The disadvantage of the network is no two inputs of the switch in the first stage

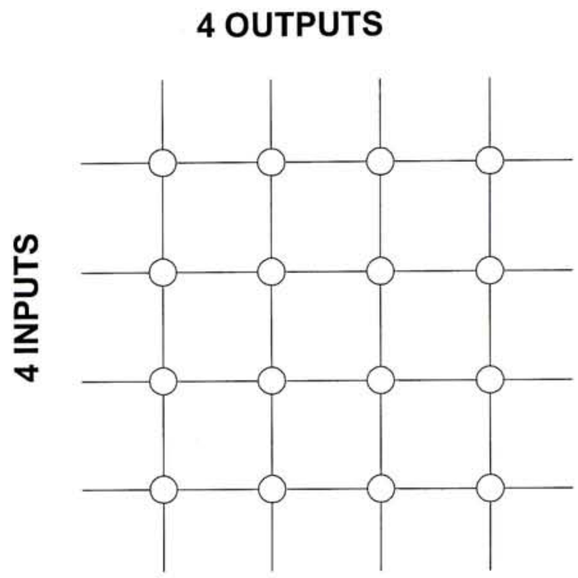

Figure 4: Switch relays for N=4

could be connected respectively to two outputs of the switch in the second stage. For example, connecting input x to output y is impossible in Figure 5 if the input a is already connected with the output b. The reason is there is only one circuit between every switch in the first stage and every switch in the second stage.

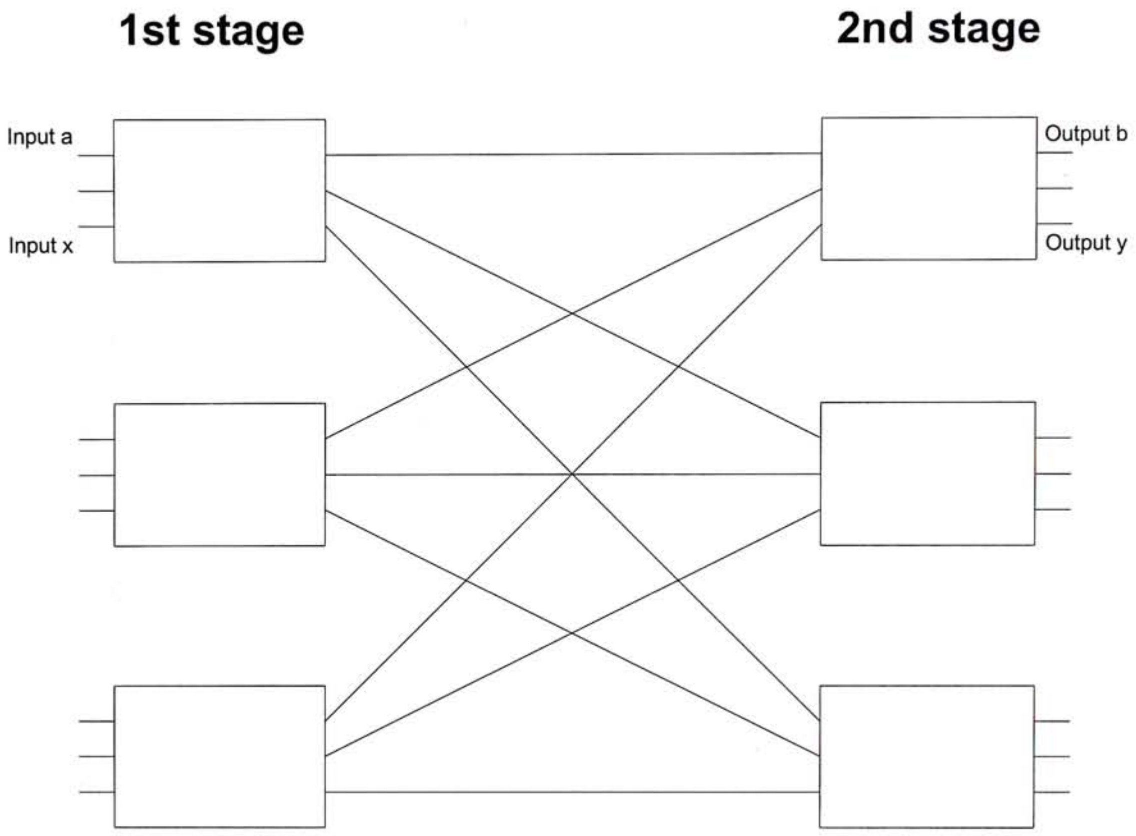

Figure 5: A two-stage switching network for N=9

1.2 Nonblocking properties of 3-stage networks

In 1953, Clos proposed a modification to the network in order to cope with the blocking problem. His suggestion was to add an intermediate stage into a two-stage network. The function of the intermediate stage is to allow alternative pathways for making a connection. Therefore, three-stage network is also called as Clos network. In addition, every switch in the first-(third-) stage having a connection link to each switch in the second stage is classified as the Clos network. There are three types of nonblocking three-stage networks: *strict-sense nonblocking, rearrangeable nonblocking,* and *wide-sense nonblocking.*

A symmetric Clos network has the same number of inputs and outputs in the first- and third-stage switches. In addition, they also have the same number of switches in the first and third stage. Let $[n\times m,\ r\times r,\ m\times n]$ denote a 3-stage Clos network where the first-, second- and third-stage nodes are, respectively, $n\times m$, $r\times r$ and $m\times n$. Figure 6 shows $[4\times4,\ 3\times3,\ 4\times4]$.

In addition, a strict-sense nonblocking type is a switching network, in which a connection could always be made between an idle input and an idle output without changing the connections already established. The minimum number of middle-stage switches $min(m)$ should be equal to $2n-1$ where n is the number of inputs (outputs) of the first- (third-) stage switch. The theorem is easy to prove. Assume making a connection from an input of a first-stage switch (called a) to an output of a third-stage switch (called x). The remaining $n-1$ inputs of the switch a may be using $n-1$ middle-stage switches for making connections. The case will be

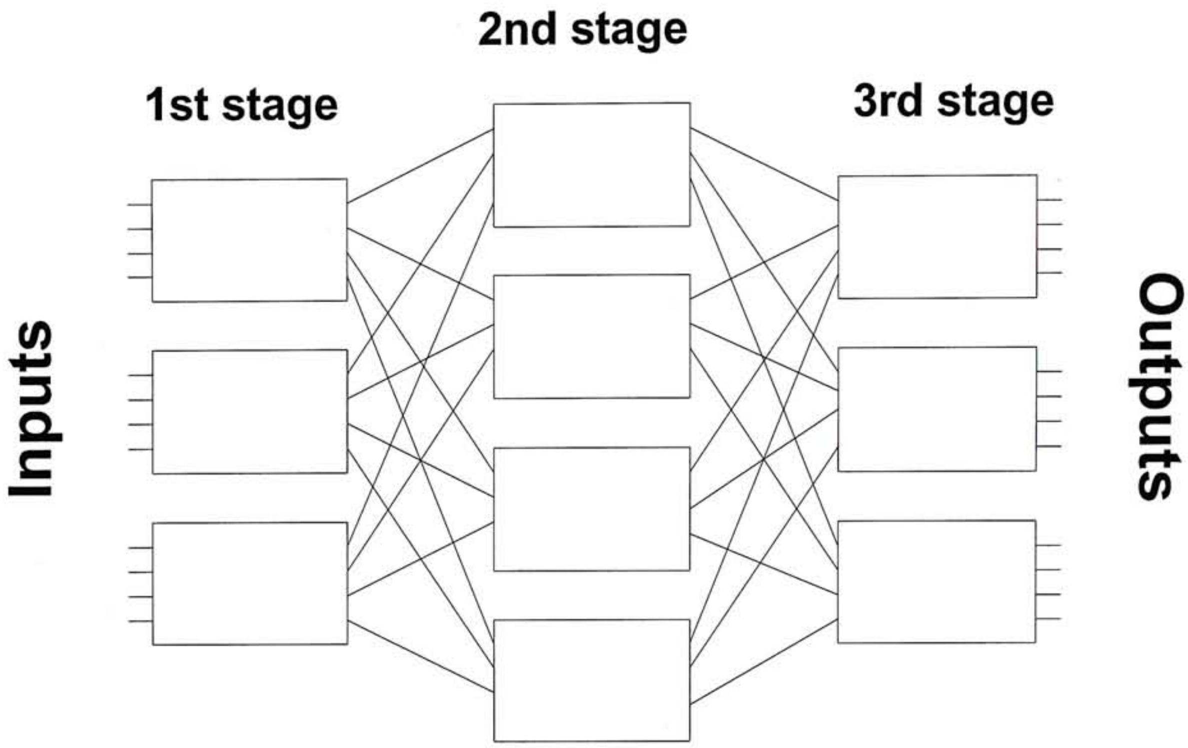

Figure 6: [4×4, 3×3, 4×4] network

similar for the switch x. Therefore, totally $2n-2$ middle-stage switches are needed to handle the worst case. To make the last connection for switch a to switch x, an additional middle-stage switch would be sufficient. That is why $min(m)$ should be equal to $2n-1$. For easy understanding, Figure 7 explains the case for a [3×5, 3×3, 5×3] network. The connection between a and b needs the last empty middle-stage switch.

In order to reduce the complexity of the switching network, adding minimum intermediate switches $min(m)$ to provide a totally nonblocking

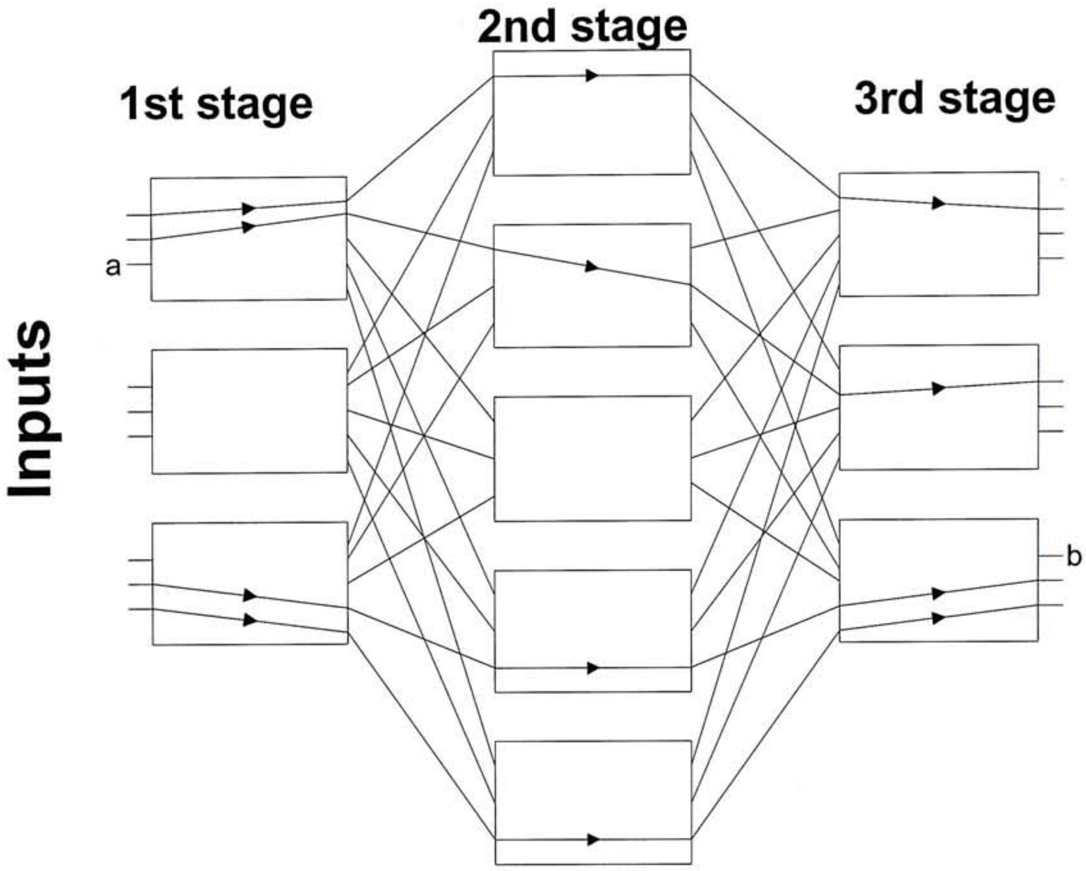

Figure 7: [3×5, 3×3, 5×3] network

network becomes a general question. Thus, if $m < 2n - 1$, blocking may occur. One of the solutions is by rearranging the already established connections in order to avoid the blocking. This is the concept of the rearrangeable nonblocking switching network (RSN). Since there are n inputs and n outputs for the first-stage switch and the third-stage switch respectively, at least n middle switches will be needed to implement the RSN. The main disadvantage of the RSN is to do connection rearrangement for some new connection requests. It will increase the complexity of the switch and also a smart enough algorithm is needed to guarantee the reconnection delay. In addition, for some real time application like telephone conversation, the effect of the rearrangement reconnection may affect the quality of the service.

Based on the above mention, strict-sense nonblocking network will be an ideal switching network . However, $2n - 1$ middle-stage switches will be a great cost if n is large. In commercial or industry field, the network increases the manufacturing cost a lot. In engineering field, too many devices putting together may cost a lot of interference and heating problems. Therefore, unstable switching network would be the result.

Moreover, some may suggest using the RSN. Absolutely, the main disadvantages mentioned before will cause great objection pressure. According to the above consideration, another solution needs to be proposed to balance the two. That means we need to find a switching network which uses less middle-stage switches than strict-sense nonblocking networks and also some smart routing algorithms can avoid blocking without doing any rearrangement. This kind of switching network is called wide-sense nonblocking (WSN) switching network. The network is said to be WSN if there is a nonblocking algorithm for route selection.

A nonblocking switching network needs to fulfill the connection requisition from all users if both ends are idle. Switching becomes more and more important because of the rapid development of the computer network and the communication network. The design of a network for interconnecting a large number of telecommunication end users with high bandwidth could be a complex problem. Therefore, how to provide an efficient and effective nonblocking switch with $min(m)$ is a more crucial issue than ever before.

1.3 Wide-sense nonblocking networks

Denote by $WS(n,\ r)$ the minimum value of m such that $[n\times m,\ r\times r,\ m\times n]$ is WSN. An algorithm is called a *packing* algorithm if the selected route is always through one of the heaviest loaded second-stage node. For $[n\times m,\ 2\times 2,\ m\times n]$, there is a unique packing algorithm, which is obviously optimal in the sense of requiring the minimum number of second-stage nodes. In 1965, this packing algorithm was proved by Beneš [1] to be nonblocking when $m \geq \lfloor 3n/2 \rfloor$. Consequently, $WS(n,\ 2) \leq \lfloor 3n/2 \rfloor$. It is not hard to see that $WS(n,\ 2) \geq \lfloor 3n/2 \rfloor$ and hence $WS(n,\ 2) = \lfloor 3n/2 \rfloor$.

This result identified a family of 3-stage WSN networks that are strictly nonblocking. It also raised the question on the existence of any WSN, but not strictly nonblocking, network in the form of $[n{\times}m,\ r{\times}r,\ m{\times}n]$, where $r > 2$. In Section 3, we shall answer this question affirmatively with a nonblocking algorithm over the network $[6{\times}10,\ 3{\times}3,\ 10{\times}6]$. The algorithm relies on the concept of *upper ideals* to be described in Section 2.

Other known facts on values of $WS(n,\ r)$ are as follows. A strictly nonblocking network is automatically WSN, hence $WS(n,\ r) \leq 2n-1$. E. F. Moore proved that $WS(n,\ r) = 2n-1$ when $r > (n-1)\binom{2n-2}{n-1}$. This result was recorded in Kurshan-Benes [7]. Du-Fishburn-Gao-Hwang [3] proved that $WS(n,\ r) \geq \lfloor 7n/4 \rfloor$ when $r \geq 3$. Li [9] proved that

- $WS(n,\ r) \geq \lfloor (7n+1)/4 \rfloor$ when $r \geq 4$ and
- $WS(n,\ r) \geq \lfloor (7n+2)/4 \rfloor$ when $n \geq 3$ and $r > \rho(\lfloor n/4 \rfloor)$, where the function $\rho(s)$ means the maximum number of $(4s+2)$-element subsets in a $(7s+3)$-element set such that the pairwise intersection between any two of these subsets contains exactly $2s+1$ elements. For example, since $\rho(1) = 4$ and $\rho(2) = 6$. Therefore, $WS(6,\ r) = 11$ for $n \geq 5$ and $WS(10,\ r) \geq 18$ for $r \geq 7$.

n \ r	2	3	4	5 -6	7- 5544	5545- 24024	24025- 102960	102961 - 437580	437581+
2	3								
3	4	5							
4	6	7							
5	7	8 or 9	9						
6	9	10 or 11		11					
7	10	12 or 13				13			
8	12	14 or 15					15		
9	13	15-17	16 or 17					17	
10	15	17-19			18 or 19				19

Table 1: Previously known values and bounds of $WS(n,\ r)$ for $n \leq 10$.

From all these results, Table 1 summarizes the values of $WS(n,\ r)$ for $n \leq 10$. The smallest undetermined value in Table 1 is $WS(5,\ 3)$. In Section 5, we shall prove that the network $[5{\times}8,\ 3{\times}3,\ 8{\times}5]$ is not WSN and hence $WS(5,\ 3) = 9$. The proof again relies on the concept of *upper ideals.*

1.4 Routing algorithms by packing

Define a network as *strictly nonblocking by packing* if every packing algorithm is nonblocking over the network. Similarly, a network is *WSN by packing* if a certain packing algorithm is nonblocking. Smith [10] proved that the number of second-stage nodes required for a WSN network cannot be less than $\lfloor 2n - n/r \rfloor$ under a large class of packing rules. Du-Fishburn-Gao-Hwang [5] proved that $[n\times m,\ r\times r,\ m\times n]$ is WSN by packing only if $m \geq \lfloor \left(2 - \frac{1}{2^{r-1}}\right) n \rfloor$. Yang [11] sharpened this necessary condition to $m \geq \lfloor \left(2 - \frac{1}{F_{2r-1}}\right) n \rfloor$, where F_{2r-1} is the Fibonacci number.

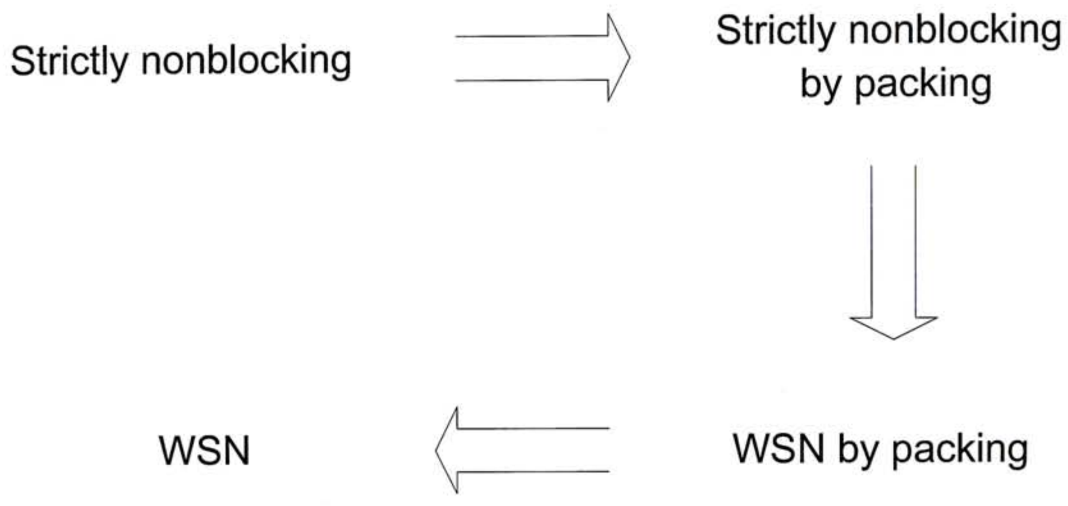

Figure 8: Relationship among various nonblocking properties of networks.

Figure 8 shows the obvious relationship among various nonblocking properties of networks. For $[n\times m,\ 2\times 2,\ m\times n]$ in particular, the two horizontal implications in the figure are reversible since packing obviously minimizes the required number of second-stage nodes. Thus the vertical implication in the figure is not reversible by the aforementioned result of Beneš [1].

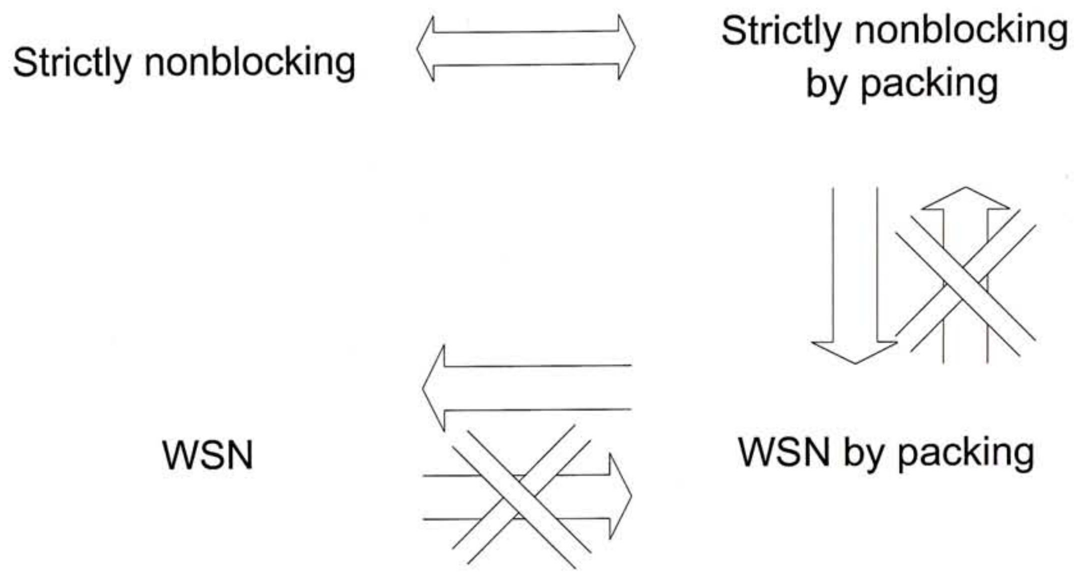

Figure 9: Relationship among various nonblocking properties of the network $[n{\times}m,\ r{\times}r,\ m{\times}n]$.

A natural question is then: for $[n{\times}m,\ r{\times}r,\ m{\times}n]$ in general, are the two horizontal implications in Figure 8 still reversible? Following the proof of Lemma 2.1.6 and Theorem 2.1.7 in Hwang [5], one can establish the necessity of the inequality $m \geq 2n-1$ for $[n{\times}m,\ r{\times}r,\ m{\times}n]$ to be strictly nonblocking by packing. Thus, the upper horizontal implication in Figure 8 remains reversible for a general r. In Section 6, we shall prove that, if $[n{\times}m,\ 3{\times}3,\ m{\times}n]$ is WSN by packing, then $m \geq \lfloor \frac{15n}{8} \rfloor$. Therefore, the WSN network $[6{\times}10,\ 3{\times}3,\ 10{\times}6]$ is not WSN by packing. Figure 9 summarizes these conclusions.

2 The Concept of the Upper Ideals

The focus of the thesis is on the network $[n \times m,\ 3 \times 3,\ m \times n]$. Let the three first-stage nodes be labeled as a, b and c and the three third-stage nodes as x, y and z. As far as a routing algorithm over the 3-stage network is concerned, a connected route can be identified by the node it traverses through at every stage and a connection request can be identified by the first- and third-stage nodes. Label inputs and outputs of a middle-stage node as a, b, c, x, y, and z according to the node each I/O is linked to. This convention is shown in Figure 10. Hereafter a *node* will refer to a middle-stage node unless otherwise specified.

The *state* of a node is the set of connections it is carrying. For example, the set $\{ax, bz\}$ represents the state of carrying two connections, one from a to x and the other from b to z. For the sake of simplicity, the notation for this state will be simplified as $axbz$. The empty state is denoted as 0. Altogether there are 34 possible states displayed in Figure 11. Two states are connected by a line in the figure when one can be obtained by adding a new route to the other.

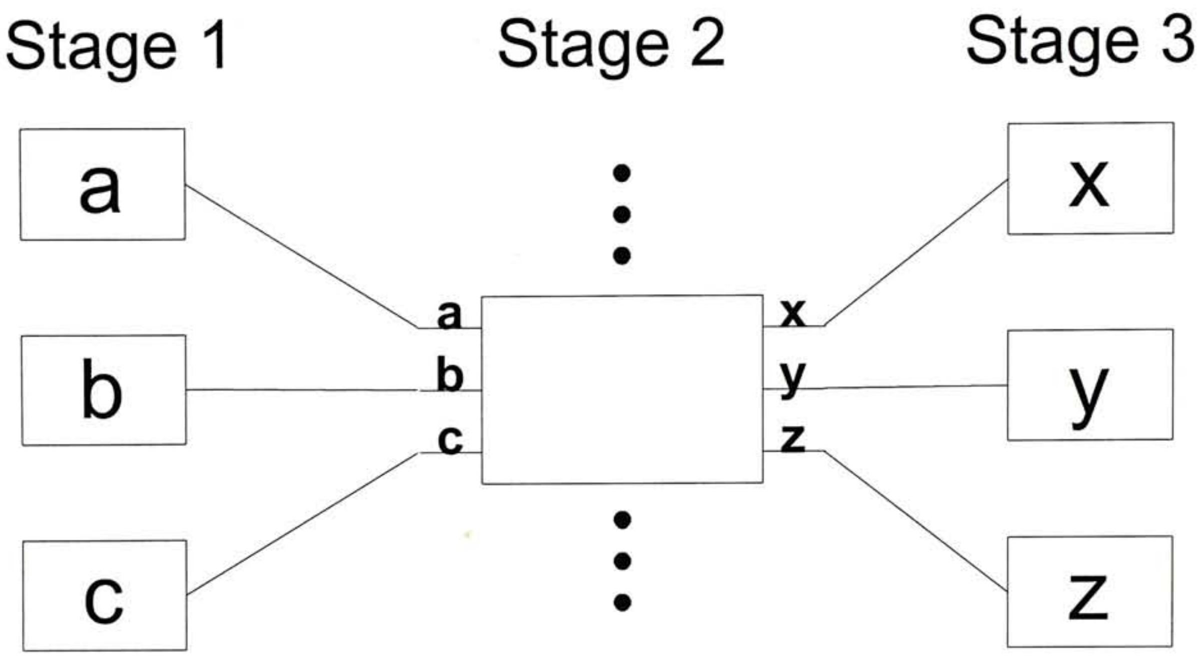

Figure 10: I/O labels of a middle-stage node.

The set-theoretical containment endows a natural order among the 34 states. The state 0 is regarded as the *smallest*. The *network state* of $[n{\times}m,\ r{\times}r,\ m{\times}n]$ is determined by states of the m individual nodes. Following the terminology of lattice theory, we have the definition below.

Definition 1 *With respect to a network state of* $[n{\times}m,\ 3{\times}3,\ m{\times}n]$, *a set* S *of nodes is called an* ***upper ideal*** *if, whenever* S *contains a node in the node state* θ, *then* S *contains every node in every node state greater than or equal to* θ. *The upper ideal* generated *by a collection of node states means the smallest upper ideal that includes all nodes in these states.*

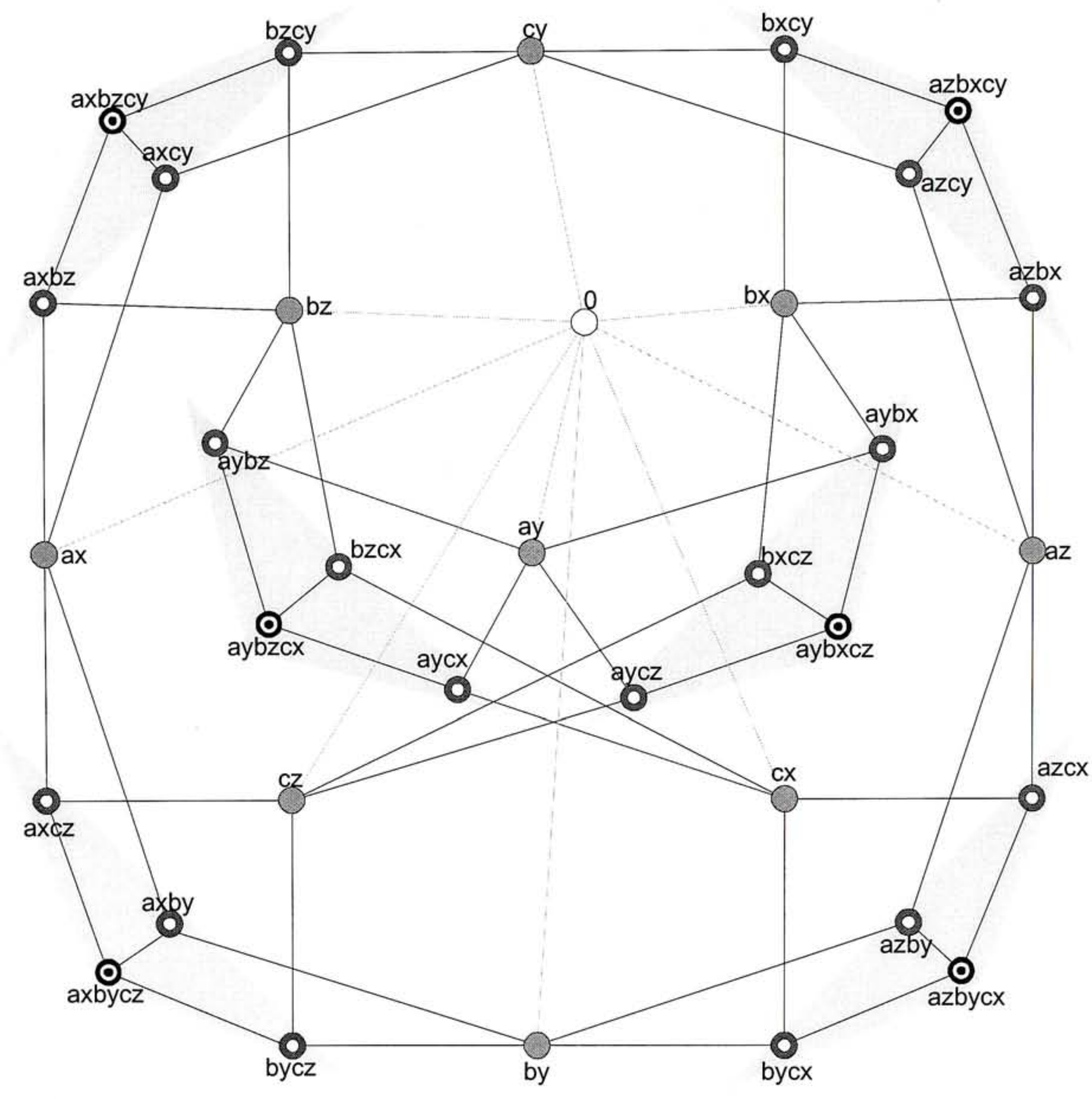

Figure 11: Chessboard for $r = 3$

Notation *We shall use the notation* $< \ldots >$ *for upper ideal generation. Thus, for example,* $<axby>$ *represents the set of nodes in the two states* $axby$ *and* $axbycz$, *while* $<ax,\ by>$ *represents the set of all nodes in the twelve states* ax, by, $axby$, $axbz$, $axcy$, $axcz$, $azby$, $bycx$, $bycz$, $axbycz$, $axbzcy$, *and* $azbycx$. *The cardinality of a set* $\mathcal{S}$ *of nodes will be denoted as* $|\mathcal{S}|$. *For every node state* θ, *let* $\mathcal{S}_\theta$ *denote the set of nodes in the state* θ.

Definition 2 *Define the upper ideals.*

$H_a = <ax,\ ay,\ az>$, $H_x = <ax,\ bx,\ cx>$,

$I_{ax} = <ay,\ az,\ bx,\ cx>$,

$J_{ax} = <ax,\ bycz,\ bzcy>$, *and*

$K_{ax} = <bycz,\ bzcy>$.

For $s \in \{a,\ b,\ c\}$ *and* $t \in \{x,\ y,\ z\}$, *the upper ideals* H_s, H_t, I_{st}, J_{st}, *and* K_{st} *are defined symmetrically.*

Lemma 1 *A network state such that* $|H_a| < n$, $|H_x| < n$, *and* $|H_a \cup H_x| = m$ *leads to blocking regardless the routing algorithm.*

Proof The two inequalities ensure that the first-stage nodes a and the third-stage x are not saturated. Thus a connection from a to x could be requested. The assumption of $|H_a \cup H_x| = m$ blocks this request. □

Lemma 1 can prove the condition $m = 2n-1$ of the strict-sense nonblocking networks easily as follows. For the worst case, $|H_a| = |H_x| = n-1$. Besides, blocking will occur if $m = 2n-2$. One more middle-stage switch is required to handle the last connection ax. Therefore, $m = 2n - 1$ middle-stage nodes are needed regardless of the routing algorithm for a strict-sense nonblocking network.

Actually, the whole state set of a network can be divided into two sub-sets. They are nonblocking and blocking. The nonblocking set is also called as "safe" set for a particular routing algorithm A. If a network state is inside the set, any connection request can be fulfilled by the routing algorithm A. That is no matter how we add or delete the routes. The switch still can keep away from blocking.

Obviously, the blocking set is a pointer to avoid blocking. It indicates that the state of the network will be blocked in future. Therefore, the best way to find out the algorithm is to kick out all the blocking states. On the other hand, the ideal situation is all routing processes are inside the nonblocking set.

Furthermore, blocking set is further divided into two. They are *absolutely blocking set* and *may-be blocking set*. In the absolutely blocking set, no matter which algorithm you are using blocking still occurs. Therefore, the absolute blocking set is algorithm independent. On the other hand, the may-be blocking set is algorithm dependent.

Besides, we can conclude that for the whole network state set. It divides into two sub-sets. The size of them is depends on which algorithm have been chosen. However, inside the blocking set, there is a fixed size and algorithm independent sub-set called absolutely blocking set as shown in Figure 12.

I_{st} is a good element to construct an indicator for the absolutely blocking set. In addition, those network states which cannot fulfil the Theorem 1 can be found blocking in a few steps no matter which routing algorithm is used.

Theorem 1 *For the network* $[n{\times}m,\ 3{\times}3,\ m{\times}n]$ *with* $m < 2n-1$*, a net-*

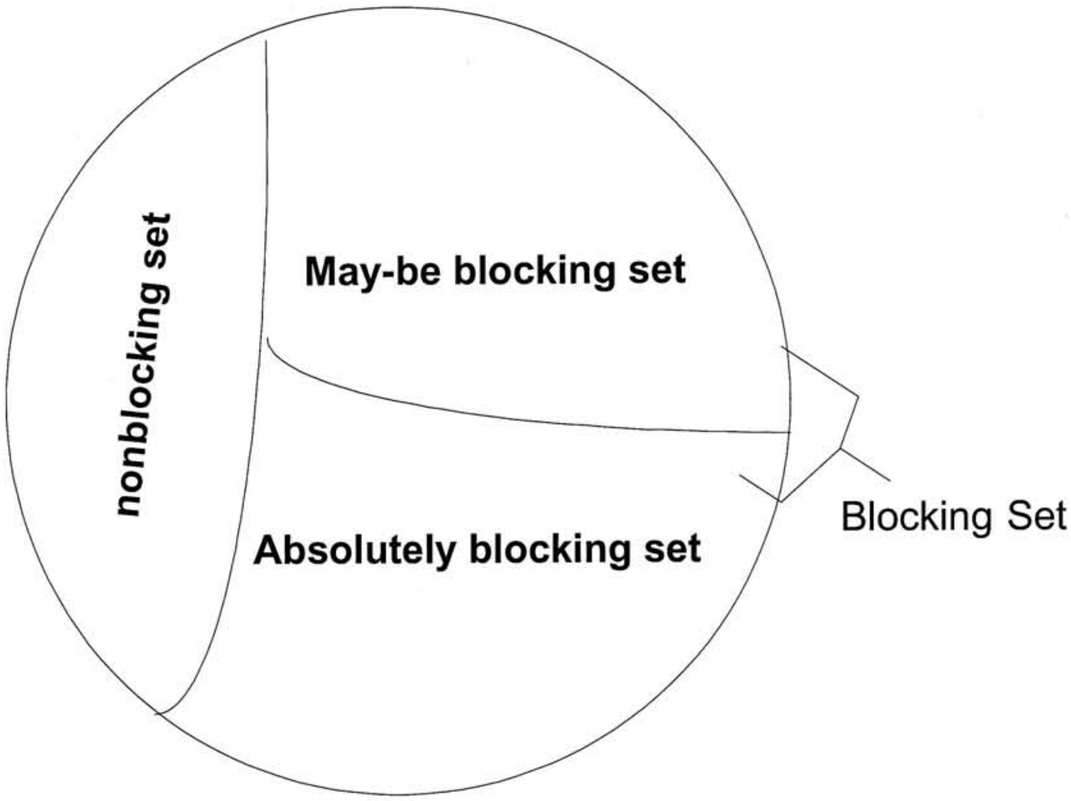

Figure 12: Set diagram

work state such that $|I_{ax}| = m$ *leads to blocking regardless the routing algorithm.*

<u>Proof</u> Delete all routes except those traversing through the first-stage node a or the third-stage node x. Thus every node becomes in the state ay, az, bx, or cx. Let $|\mathcal{S}_{ay}|+|\mathcal{S}_{az}| = i$ and $|\mathcal{S}_{bx}|+|\mathcal{S}_{cx}| = m-i$. Thus $n \geq i \geq m-n$.

Case 1: $i < n$ and $m - i < n$.

By Lemma 1, the network state leads to blocking.

Case 2: $i = n$. Thus $m - i = m - n < n - 1$.

Add a connection from b to x. Then, $|\mathcal{S}_{ay}| + |\mathcal{S}_{az}| = n - 1$, $|\mathcal{S}_{aybx}| + |\mathcal{S}_{azbx}| = 1$ and $|\mathcal{S}_{bx}| + |\mathcal{S}_{cx}| = m - n$.

Delete the old route in the node in the state $aybx$ or $azbx$. Then $|\mathcal{S}_{ay}| + |\mathcal{S}_{az}| = n - 1$ and $|\mathcal{S}_{bx}| + |\mathcal{S}_{cx}| = m - n + 1$.

The network state is as in Case 1.

Case 3 : $m - i = n$. The case is symmetric to Case 2. □

The theorem offers a guideline in routing: Avoid the situation when $|I_{ax}| = m$ or, more generally, when $|I_{st}| = m$ for any s and any t.

Before the introduction of Theorem 2, let us investigate the following example. The focus of the thesis in WSN networks are on the $[n \times \lfloor \frac{7n}{4} \rfloor, 3 \times 3, \lfloor \frac{7n}{4} \rfloor \times n]$.

Example: Consider the given network state and find out the unknown value of X, Y and Z.

State: $|S_0| = X$, $|S_{ax}| = Y$, and $|S_{bycz}| = Z$

From the above, we can get the following three inequality :

$$\begin{cases} X + Y \leq n & \cdots\cdots(1) \\ X + Z \leq n & \cdots\cdots(2) \\ X + Y + Z = m = \lfloor \frac{7n}{4} \rfloor & \cdots\cdots(3) \end{cases}$$

$(1) + (2)$:

$$2X + Y + Z \leq 2n$$

sub. into (3) :

$$X \leq 2n - \lfloor \tfrac{7n}{4} \rfloor$$

$$\Rightarrow \qquad X \leq \lceil \tfrac{n}{4} \rceil$$

Form (1) : $\qquad Y \leq n - X$

$$\Rightarrow \qquad Y \geq n - \lceil \tfrac{n}{4} \rceil$$

$$\Rightarrow \qquad Y \geq \lfloor \tfrac{3n}{4} \rfloor$$

similarly, $\qquad Z \geq \lfloor \frac{3n}{4} \rfloor \qquad \square$

From the above example, if all empty node states are occupied by all ay or all az, the network is blocking by Theorem 1. The inequalities indicate that sufficient connection requests from first-stage node a to third-stage node y or z can fully occupy all the middle-stage nodes. Based on this argument, the inequalities can give out a danger signal that Theorem 1 will be violated soon. In order to fulfil the inequalities more easily, we maximize the value of X and minimize the vlaue of Y. Thus, by the solutions of the example, $X = \lceil \frac{n}{4} \rceil$ and $Y = Z = \lfloor \frac{3n}{4} \rfloor$.

Theorem 2 *For the network* $[n{\times}m,\ 3{\times}3,\ m{\times}n]$ *with* $m = \lfloor \frac{7n}{4} \rfloor$, *a network state such that* $|J_{ax}| \geq 2\lfloor \frac{3n}{4} \rfloor$ *and* $\lfloor \frac{3n}{4} \rfloor \leq |K_{ax}| \leq n$ *leads to blocking regardless the routing algorithm.*

<u>Proof</u> Delete some old routes until every node becomes in the state ax, $bycz$ and $bzcy$. Let $|\mathcal{S}_{ax}| = i$. Then, $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = 2\lfloor \frac{3n}{4} \rfloor - i$. Thus $2\lfloor \frac{3n}{4} \rfloor - n \leq i \leq \lfloor \frac{3n}{4} \rfloor$.

Case 1: $i = \lfloor \frac{3n}{4} \rfloor$.

Add $\lceil \frac{n}{4} \rceil$ connections from the first-stage node a to the third-stage node z. Then $|\mathcal{S}_{ax}| = \lfloor \frac{3n}{4} \rfloor$, $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = \lfloor \frac{3n}{4} \rfloor$ and $|\mathcal{S}_{az}| = \lceil \frac{n}{4} \rceil$. Thus $|I_{ay}| = |\mathcal{S}_{ax}| + |\mathcal{S}_{az}| + |\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = \lfloor \frac{3n}{4} \rfloor + \lceil \frac{n}{4} \rceil + \lfloor \frac{3n}{4} \rfloor = \lfloor \frac{3n}{4} \rfloor + n = \lfloor \frac{7n}{4} \rfloor$. By Theorem 1, the network state leads to blocking.

Case 2: $2\lfloor \frac{3n}{4} \rfloor - n \leq i < \lfloor \frac{3n}{4} \rfloor$.

Add $(\lfloor \frac{3n}{4} \rfloor - i)$ connections from a to x. Then $|\mathcal{S}_{ax}| = i + k$, $|\mathcal{S}_{axbycz}| + |\mathcal{S}_{axbzcy}| = \lfloor \frac{3n}{4} \rfloor - i - k$ and $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = \lfloor \frac{3n}{4} \rfloor + k$ where $0 \leq k \leq \lfloor \frac{3n}{4} \rfloor - i$.

Delete the two old routes in every node in the state $axbycz$ or $axbzcy$.

Then $|\mathcal{S}_{ax}| = \lfloor \frac{3n}{4} \rfloor$ and $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = \lfloor \frac{3n}{4} \rfloor$ +k.

Clear routes in k of the nodes in the state $bycz$ or $bzcy$. The situation becomes as in Case 1. □

Theorem 3 *For the network* $[n{\times}m,\ 3{\times}3,\ m{\times}n]$ *with* $m = \lfloor \frac{7n}{4} \rfloor$, *denote by* NS_1, *a network state with* $|\mathcal{S}_{ax}| = n$ *and* $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = \lfloor \frac{3n}{4} \rfloor - \lceil \frac{n}{4} \rceil$. *Then, a network state such that* $|J_{ax}| \geq 2\lfloor \frac{3n}{4} \rfloor$ *leads to either blocking or* NS_1 *regardless the routing algorithm.*

Proof

Delete all routes until every node state becomes ax, $bycz$ and $bzcy$. Let $|\mathcal{S}_{ax}| = i$, $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = 2\lfloor \frac{3n}{4} \rfloor - i$. Thus $2\lfloor \frac{3n}{4} \rfloor - n \leq i \leq n$.

Case 1: $2\lfloor \frac{3n}{4} \rfloor - n \leq i \leq \lfloor \frac{3n}{4} \rfloor$

By Theorem 2, the network state leads to blocking state.

Case 2: $\lfloor \frac{3n}{4} \rfloor < i \leq n$

Add $(n-i)$ connections from the first-stage node a to the third-stage node x. Then $|\mathcal{S}_{ax}| = i + k$, $|\mathcal{S}_{axbycz}| + |\mathcal{S}_{axbzcy}| = n - i - k$ and $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = 2\lfloor \frac{3n}{4} \rfloor - n + k$. Thus $0 \leq k \leq n - i$.

Delete the two old routes in every node in the state $axbycz$ or $axbzcy$. Then the network state becomes $|\mathcal{S}_{ax}| = n$, $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = 2\lfloor \frac{3n}{4} \rfloor - n + k$.

Clear routes in k of the nodes in the state $bycz$ or $bzcy$. The network state becomes in the state NS_1. □

Theorems 2 and 3, as well as Theorem 1, offer guidelines in routing.

3 Routing algorithm over the network [6×10, 3×3, 10×6]

[6×10, 3×3, 10×6] is proved by the *Simulation Program* (SP) that there exists a WSN routing algorithm. Only a counter example is sufficient to prove that a network is blocking. However, it is very difficult to verify that the network is WSN. Therefore, an effective and efficient way to do the proof is by the exhaustive computer search. The SP is designed to do this boring and tedious job. The main function of the SP is to go through all the possible routes. A nonblocking routing algorithm can start from the empty network state and exit the program because no more new network state can be found by the SP.

Since the packing algorithm is proved not good enough in Section 6, a new routing algorithm need to be found to make the network WSN. Therefore, the concept of the upper ideals can be used to avoid the routing algorithm to step into the blocking set. On the other hand, the routing algorithm should be very simple. For a complex routing algorithm, most of the effort is put into filtering which reduce the efficiency.

Algorithm 1 *Let a network state of the network [6×10, 3×3, 10×6] be called* ***admissible*** *if it satisfies the following two conditions*

(1) $|I_{st}| \leq 9$.

(2) Either $|J_{st}| \leq 7$ *or* $|K_{st}| \leq 3$.

for all s and t. Moreover, let an ***admissible*** *state be called* ***preferred*** *when* $|I_{st}| \leq 9$ *and* $|J_{st}| \leq 7$ *for all s and t. Given a network state of the network, an input node* s_0 *that is carrying less than six routes, and an output node* t_0 *that is carrying less than six routes, the algorithm needs to choose a middle-stage node for adding a connection from* s_0 *to* t_0 *such that the resulting state of the network is* ***admissible****. The criteria of the selection of the middle-stage node are as follows.*

- *First, it is preferred that the resulting network state of the network is a* ***preferred*** *one. Among such choices, priority is given to a middle-stage node that has been carrying 1, 0, or 2 routes, by that order. The tiebreaker is the label of the middle-stage node in the increasing order.*

- *When no choice leads to a* ***preferred*** *state, then one leading to an* ***admissible*** *state is adopted. Priority is given to a middle-stage node already carrying 0, 1, or 2 routes, by that order. The tiebreaker is again the label of the middle-stage node.* □

Theorem 4 *The network $[6\times 10,\ 3\times 3,\ 10\times 6]$ is WSN. In fact, Algorithm 1 is nonblocking over the network.*

Proof Firstly, the initial state of the network is the empty state. The evolution of the state is through the addition of a new route to a middle node or the deletion of an existent route at a time. In the case of adding a new route, the number N of possible new states may be 0, 1 or more. When $N = 0$, blocking occurs. When $N > 1$, the new state depends upon the algorithm of route selection. Exhaustive computer search identifies a set of **admissible** states that include the empty state and is "closed" under arbitrary route deletion and under route addition by Algorithm 1. The theorem is thus proved constructively. □

Algorithm 1 is generalized as follows. It is not known whether the generalized algorithm, Algorithm 2, is nonblocking over networks $[n\times\lfloor\frac{7n}{4}\rfloor,\ 3\times 3,\ \lfloor\frac{7n}{4}\rfloor\times n]$, $n \geq 6$.

Algorithm 2 *Let a network state of the network* $[n\times\lfloor\frac{7n}{4}\rfloor,\ 3\times 3,\ \lfloor\frac{7n}{4}\rfloor\times n]$ *be called* ***admissible*** *if it satisfies the following two conditions*

(1) $|I_{st}| \leq m-1$.

(2) Either $|J_{st}| \leq 2\lfloor\frac{3n}{4}\rfloor - 1$ *or* $|K_{st}| \leq \lfloor\frac{3n}{4}\rfloor - 1$.

for all s *and* t*. Moreover, let an* ***admissible*** *state be called* ***preferred*** *when* $|I_{st}| \leq m-1$ *and* $|J_{st}| \leq 2\lfloor\frac{3n}{4}\rfloor - 1$ *for all* s *and* t*. Given a network state of the network, an input node* s_0 *that is carrying less than* n *routes, and an output node* t_0 *that is carrying less than* n *routes, the algorithm needs to choose a middle-stage node for adding a connection from* s_0 *to* t_0 *such that the resulting state of the network is* ***admissible****. The criteria of the selection of the middle-stage node are as follows.*

- *First, it is preferred that the resulting network state of the network is a* ***preferred*** *one. Among such choices, priority is given to a middle-stage node that has been carrying 1, 0, or 2 routes, by that order. The tiebreaker is the label of the middle-stage node in the increasing order.*

- *When no choice leads to a* ***preferred*** *state, then one leading to an* ***admissible*** *state is adopted. Priority is given to a middle-stage node already carrying 0, 1, or 2 routes, by that order. The tiebreaker is again the label of the middle-stage node.* □

4 Simulation Program (SP)

The SP is an important element in developing the routing Algorithm 1. The difficulties in designing this program are due to huge memory size and low computing speed. Therefore, solving these two problems needs great technical skill.

The SP uses a large buffer memory to hold all the possible network states. The data structure to represent a state is shown in Figure 13. Each byte is considered as a node state. Therefore, for [6×10, 3×3, 10×6], there are ten bytes for ten node states. The eleventh byte is using as a pointer for *hanging process* which will be discussed in the coming paragraph. The seventh and eighth bits of the first byte are flags indicating whether the network state does the insertion or deletion process.

The maximum buffer size for [6×10, 3×3, 10×6] are around 13M network states. Therefore, the SP needs at least 143M bytes of internal memory. The buffer is designed in static array in order to reduce the access time. The data is sorted and put into the buffer queue in order to do the binary search, which do the discarding process if the new network state exists in the buffer.

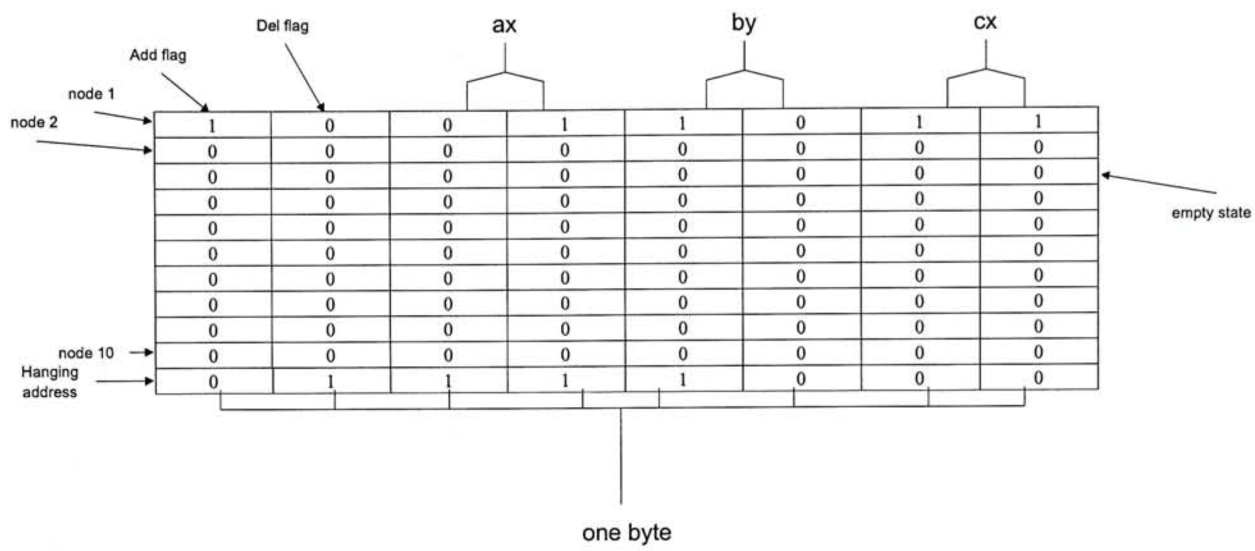

Figure 13: Data structure for $n = 6$

The whole buffer is divided into three parts. They are considered as "main" buffer, "temp" buffer and "hanging" buffer. The size of the "temp" buffer and the "hanging" buffer is fixed in 256 blocks individually. The remaining memory is the "main" buffer. Furthermore, a new state will be either put into "main" or "hanging" buffer. The function of the "hanging" buffer is to reduce the time in doing the shifting process. Let us explain in a more practice way. For example, the status of arrays is "1, 3, 45, empty" as shown in Figure 14. If a new number 40 is inserted, 45 must be shifted to an empty space and insert the new number 40. The new status of the array is "1, 3, 40, 45". Certainly, it wastes a lot of time in shifting process if lots of elements need to be shifted in each insertion. Therefore, a "hanging" buffer is used to save the new insertion data 40. The pointer of the data 3 is used to point to the location of the "hanging" buffer. The situation is shown in Figure 15. Therefore, the shifting process does not necessary require in each insertion. In addition, the shifting process can be done after the "hanging" buffer is full.

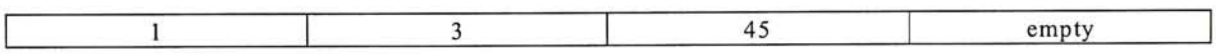

Figure 14: Status of an array

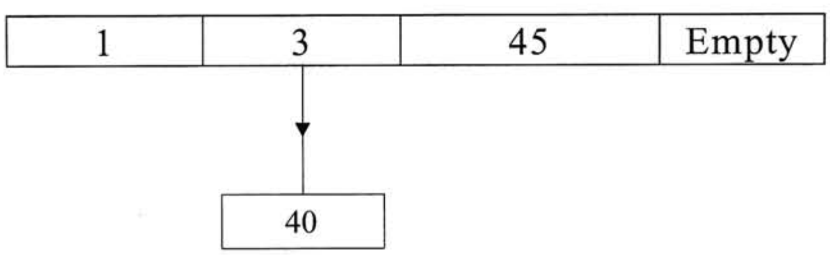

Figure 15: Hanging example

The "temp" buffer is designed to speed up the shifting process by moving the "main" buffer forward 256 blocks of memory instead of doing swapping. Therefore, only two steps (read and write) require to shift data to a new location. In each shifting process, the location of "temp", "main", and "hanging" buffer will be shifted by 256 blocks in circular as shown in Figure 16.

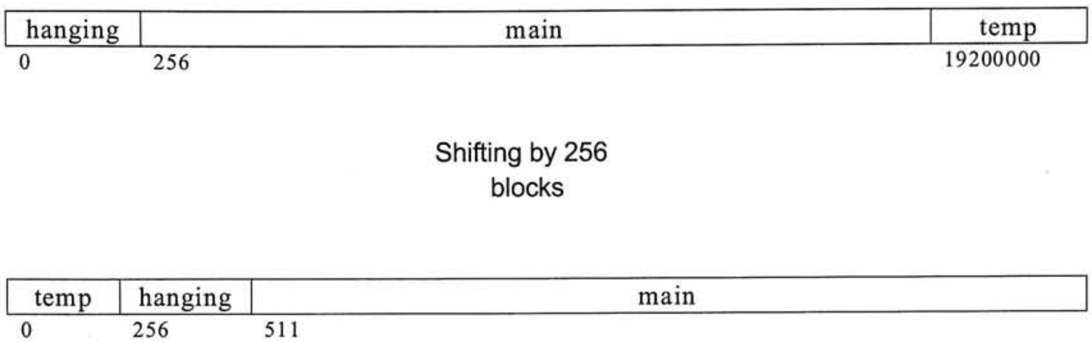

Figure 16: Shifting process

Another way to minimize the size of the database is using canonical form. A canonical network state represents a group of network states which are equal to the canonical one after doing some canonical transform CT. Network state i is equal to network state j if and only if i is equal to j after doing the CT. One example is shown in Figure 17.

Actually, the canonical state depends on what conditions you have set. In the SP, we set the following conditions.

1. $N(a) \geq N(b) \geq N(c)$.

2. $N(z) \geq N(y) \geq N(x)$.

3. The ten row vectors are sorted descending from up to down.

where the function N(t) is number of routes from (to) first- (third-)stage node t.

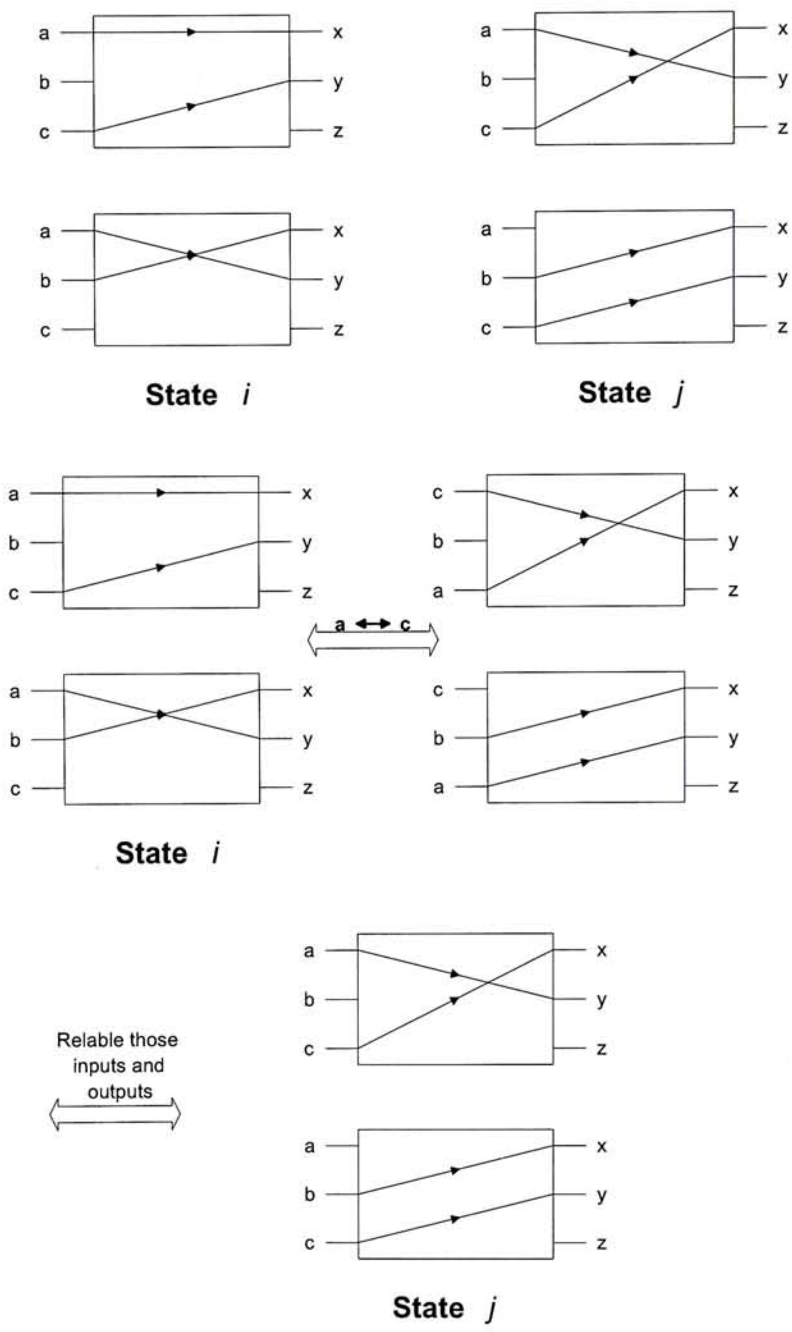

Figure 17: An example of CT

Although the SP is implemented by the above data structures, the running time is still too long. For example, we need 200M for memory, 500M for backup and 6 days to find out that Algorithm 1 is a WSN routing algorithm over [6×10, 3×3, 10×6] by a powerful ultra 5 machine. Therefore, it still needs some improvements.

5 Nonexistence of routing algorithm over the network [5×8, 3×3, 8×5]

The networks $[n\times\lfloor \frac{7n}{4} \rfloor, 3\times3, \lfloor \frac{7n}{4} \rfloor\times n]$ are strict-sense nonblocking for all $n \leq 4$ because $\lfloor\frac{7n}{4}\rfloor = 2n-1$ if $n \leq 4$. Therefore, the first smallest possible WSN network may be the network [5×8, 3×3, 8×5] which is shown in Figure 18.

Notation *Label the nodes by 1, 2, ..., m. For $j \leq k$, let $[j, k]$ denote the set of nodes $j, j+1, \ldots, k$.*

Theorem 5 *For the network $[5\times8, 3\times3, 8\times5]$, the network state NS_1 leads to blocking regardless the routing algorithm.*

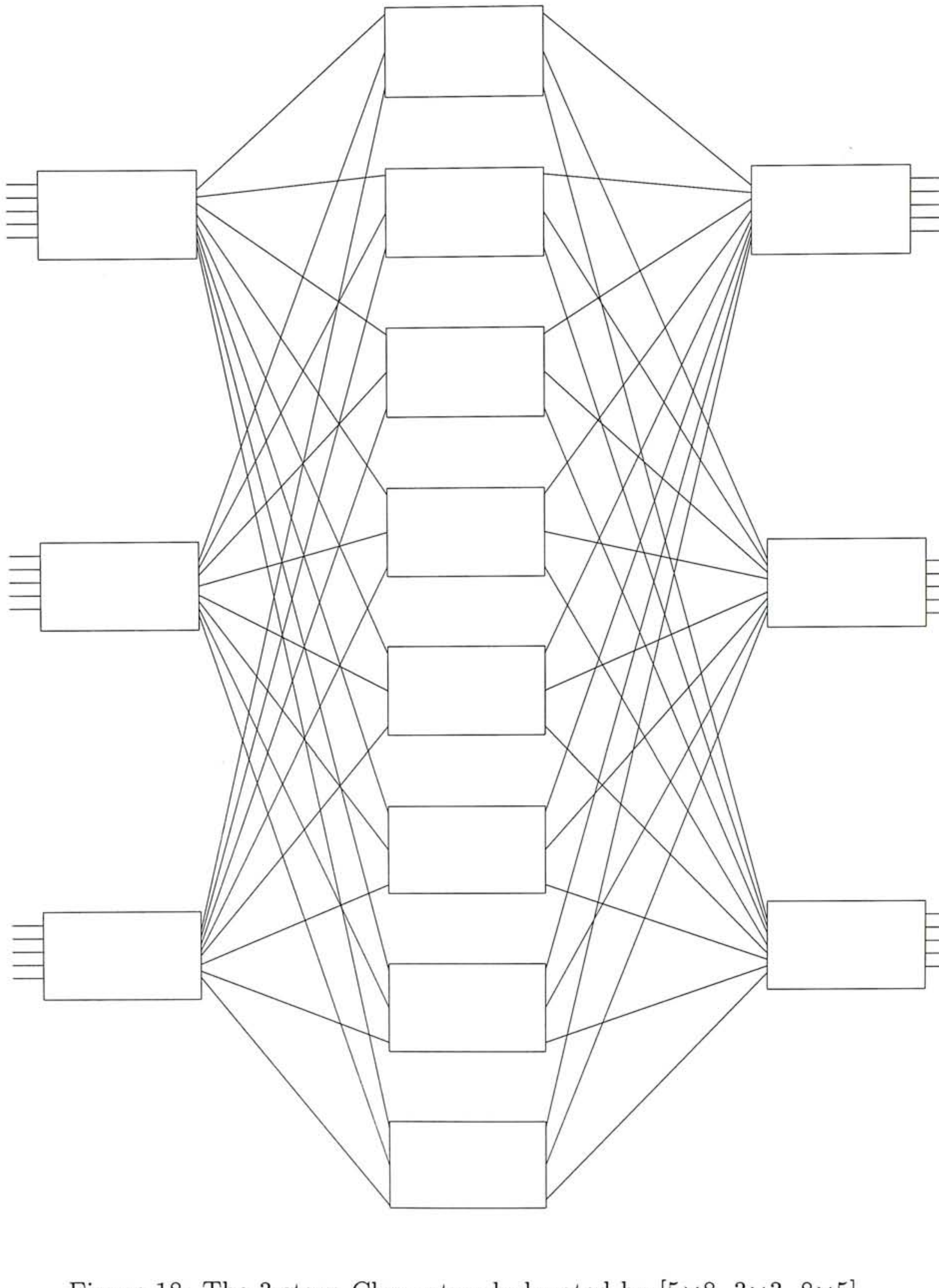

Figure 18: The 3-stage Clos network denoted by [5×8, 3×3, 8×5]

Proof Let $Th2(J_{ax})$ represents that the network state leads to blocking by Theorem 2 with $|\mathcal{S}_{ax}| = \lfloor \frac{3n}{4} \rfloor$ and $|\mathcal{S}_{bycz}| + |\mathcal{S}_{bzcy}| = \lfloor \frac{3n}{4} \rfloor$.

Let $Th1(I_{ax})$ represents that the network state leads to blocking by Theorem 1 with $|\mathcal{S}_{ay}| + |\mathcal{S}_{az}| + |\mathcal{S}_{bx}| + |\mathcal{S}_{cx}| = m$.

Let the notation "Add one ax" represents "Add one connection from frist-stage node a to the third-stage node x" and so on.

State $T1$: $\mathcal{S}_{ay}$ =[1, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Add four bz: the state will become $T11$, $T12$ or $T13$ as specified below.

$T11$: $\mathcal{S}_{aybz}$ =[1, 4], $\mathcal{S}_{ay}$ =[5, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Add two cx: the new state become $T111$, $T112$, $T113$, $T114$ or $T115$.

$T111$: $\mathcal{S}_{aybzcx}$ =[1, 2], $\mathcal{S}_{aybz}$ =[3, 4], $\mathcal{S}_{ay}$ =[5, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Delete every node becomes in the state ay, $bxcz$ or $bzcx$.

Then $Th2(J_{ay})$.

$T112$: $\mathcal{S}_{aybzcx}$ =[1, 1], $\mathcal{S}_{aybz}$ =[2, 4], $\mathcal{S}_{aycx}$ =[5, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Delete every node becomes in the state cx, $aybz$ or $azby$.

Then $Th2(J_{cx})$.

$T113$: $\mathcal{S}_{aybzcx}$ =[1, 1], $\mathcal{S}_{aybz}$ =[2, 4], $\mathcal{S}_{ay}$ =[5, 5], $\mathcal{S}_{bzcx}$ =[6, 6],

$\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_0$ =[8, 8]; ditto

$T114$: $\mathcal{S}_{aybz}$ =[1, 4], $\mathcal{S}_{aycx}$ =[5, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7],

$\mathcal{S}_0$ =[8, 8]; ditto.

$T115$: $\mathcal{S}_{aybz}$ =[1, 4], $\mathcal{S}_{ay}$ =[5, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 8]; ditto.

$T12$: $\mathcal{S}_{aybz}$ =[1, 3], $\mathcal{S}_{ay}$ =[4, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_{bz}$ =[7, 7], $\mathcal{S}_0$ =[8, 8].

Add three cx: there are two cases.

Case $T121$: middle-stage nodes 1, 2, 3 carry at least two new cx;

$T121$: $\mathcal{S}_{aybzcx}$ =[1, 2], $\mathcal{S}_{aybz}$ =[3, 3], $\mathcal{S}_{ay}$ =[4, 5], $\mathcal{S}_{bzcx}$ =[6, 6],

$\mathcal{S}_{bzcx}$ =[7, 7], $\mathcal{S}_0$ =[8, 8]; Delete every node becomes in

the state ay, $bxcz$ or $bzcx$. Then $Th2(J_{ay})$.

Case $T122$: middle-stage nodes 4 through 8 carry at least two new cx;

The case will Th $2(J_{cx})$ because node 1 to 3 carry three "$aybz$" and

node 4 to 8 carry at least three "cx".

$T13$: $\mathcal{S}_{aybz}$ =[1, 2], $\mathcal{S}_{ay}$ =[3, 5], $\mathcal{S}_{bzcx}$ =[6, 6], $\mathcal{S}_{bz}$ =[7, 8]; $Th1(I_{az})$.

All the possible routes are considered beginning from the state NS_1. Therefore, the state NS_1 leads to blocking over the network [5×8, 3×3, 8×5] regardless the routing algorithms. □

Theorem 6 *The network [5×8, 3×3, 8×5] is not WSN.*

Proof Let the notation "5-I-P $|I_{ax}|$" represents that $|I_{ax}| = m = 8$. Then, by Theorem 1, the network state leads to blocking if the state is 5-I-P $|I_{st}|$ for some s and some t.

Let the notation "5-J-P $|J_{ax}|$" represents that $|J_{ax}| \geq 2\lfloor \frac{3n}{4} \rfloor$. By Theorem 3, the network state leads to blocking if the state is 5-J-P $|J_{st}|$ for some s and some t.

By the 5-I-P and 5-J-P, it suffices to force an empty network state leading to blocking.

Add five az: the new state becomes $\mathcal{S}_{az}$ =[1, 5], $\mathcal{S}_0$ =[6,8].

Add five by: the new state becomes T1, T2, T3, or T4 as specified below.

T1: $\mathcal{S}_{azby}$ =[1, 5], $\mathcal{S}_0$ =[6, 8].

Add five cx: the new state becomes T11, T12, T13 and T14.

T11: $\mathcal{S}_{azbycx}$ =[1, 5], $\mathcal{S}_0$ =[6, 8].

Delete routes: $\mathcal{S}_{azby}$ =[1, 3], $\mathcal{S}_{azcx}$ =[4, 5], $\mathcal{S}_0$ =[6, 8].

Add two bx: $\mathcal{S}_{azby}$ =[1, 3], $\mathcal{S}_{azcx}$ =[4, 5], $\mathcal{S}_{bx}$ =[6, 7], $\mathcal{S}_0$ =[8, 8].

Add two cy: the new state becomes T111, and T112.

T111: $\mathcal{S}_{azby}$ =[1, 3], $\mathcal{S}_{azcx}$ =[4, 5], $\mathcal{S}_{bxcy}$ =[6, 7], $\mathcal{S}_0$ =[8, 8]: 0;

5-J-P $|J_{az}|$.

T112: $\mathcal{S}_{azby}$ =[1, 3], $\mathcal{S}_{azcx}$ =[4, 5], $\mathcal{S}_{bxcy}$ =[6, 6], $\mathcal{S}_{bx}$ =[7, 7],

$\mathcal{S}_{cy}$ =[8, 8]; 5-J-P $|J_{az}|$.

T12: $\mathcal{S}_{azbycx}$ =[1, 4], $\mathcal{S}_{azby}$ =[5, 5], $\mathcal{S}_{cx}$ =[6, 6], $\mathcal{S}_0$ =[7, 8]; 5-J-P $|J_{cx}|$.

T13: $\mathcal{S}_{azbycx}$ =[1, 3], $\mathcal{S}_{azby}$ =[4, 5], $\mathcal{S}_{cx}$ =[6, 7], $\mathcal{S}_0$ =[8, 8]; 5-J-P $|J_{cx}|$.

T14: $\mathcal{S}_{azbycx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 5], $\mathcal{S}_{cx}$ =[6, 8]; 5-I-P $|I_{bx}|$.

T2: $\mathcal{S}_{azby}$ =[1, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Add three cx: consider four cases T21, T22, T23, T24 and T25.

T21: the state of node 5 (az) carries a new connection from c to x;

5-J-P $|J_{by}|$ case.

T22: the state of node 6 (by) carries a new "cx" route; 5-J-P $|J_{az}|$ case.

T23: both middle-stage nodes 7 and 8 carry new connections;

5-J-P $|J_{cx}|$ case.

T24: $\mathcal{S}_{azbycx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7],

$\mathcal{S}_0$ =[8, 8].

Delete routes: $\mathcal{S}_{azcx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6],

$\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_0$ =[8, 8].

Add two bx: $\mathcal{S}_{azcx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{azbx}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6],

$\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_{bx}$ =[8, 8].

Add two cy: $\mathcal{S}_{azcx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{azbxcy}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6],

$\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_{bxcy}$ =[8, 8]; 5-J-P $|J_{az}|$.

T25: $\mathcal{S}_{azbycx}$ =[1, 3], $\mathcal{S}_{azby}$ =[4, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Add one more cx: the new state becomes T251 or T252.

T251: $\mathcal{S}_{azbycx}$ =[1, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_0$ =[7, 8].

Add one more cx. We cannot add the cx into node 5 and node 6 because of $T21$ and $T22$.

the state: $\mathcal{S}_{azbycx}$ =[1, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_0$ =[8,8].

Delete routes: $\mathcal{S}_{azcx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_0$ =[8, 8].

Add two bx: $\mathcal{S}_{azcx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{azbx}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_{bx}$ =[8, 8].

Add two cy: $\mathcal{S}_{azcx}$ =[1, 2], $\mathcal{S}_{azby}$ =[3, 4], $\mathcal{S}_{azbxcy}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_{bxcy}$ =[8, 8]; 5-J-P $|J_{az}|$.

T252: $\mathcal{S}_{azbycx}$ =[1, 3], $\mathcal{S}_{azby}$ =[4, 4], $\mathcal{S}_{az}$ =[5, 5], $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_0$ =[8, 8].

Add one cx: $\mathcal{S}_{azbycx}$ =[1, 3], $\mathcal{S}_{azby}$ =[4, 4], $\mathcal{S}_{az}$ =[5, 5]: az, $\mathcal{S}_{by}$ =[6, 6], $\mathcal{S}_{cx}$ =[7, 7], $\mathcal{S}_{cx}$ =[8, 8]; 5-J-P $|J_{cx}|$.

T3: $\mathcal{S}_{azby}$ =[1, 3], $\mathcal{S}_{az}$ =[4, 5], $\mathcal{S}_{by}$ =[6, 7], $\mathcal{S}_0$ =[8, 8].

Add five cx: consider T31, T32 and T33.

T31: middle-stage node 4 or 5 carries a new route; 5-J-P $|J_{by}|$.

T32: middle-stage node 6 or 7 carries a new route; 5-J-P $|J_{az}|$.

T33: after eliminating the states T31 and T32, we can only add four cx.

Then the network state leads to blocking.

T4: $\mathcal{S}_{azby}$ =[1, 2], $\mathcal{S}_{az}$ =[3, 5], $\mathcal{S}_{by}$ =[6, 8]; 5-I-P $|I_{bz}|$ case. □

In view of Table 1, we conclude that $WS(5,\ 3)$=9.

6 Packing algorithms

By the following Theorem 7, we prove if the network $[n\times m,\ 3\times 3,\ m\times n]$ is WSN by packing, then $m \geq \lfloor \frac{15n}{8} \rfloor$.

Lemma 2 $\lfloor \frac{7n}{8} \rfloor \geq \lfloor \frac{3n}{2} \rfloor - \lfloor \frac{3n}{4} \rfloor$ *for* $n \geq 3$

Proof

We know that,

$$k - 1 \leq \lfloor k \rfloor \leq k$$

and

$$k \leq \lceil k \rceil \leq k+1 \quad \text{where k is a natural number.}$$

Therefore,

$$\begin{aligned}
& \lfloor \frac{7n}{8} \rfloor - \lfloor \frac{3n}{2} \rfloor + \lfloor \frac{3n}{4} \rfloor \\
= & \lfloor \frac{7n}{8} \rfloor - n - \lfloor \frac{n}{2} \rfloor + \lfloor \frac{3n}{4} \rfloor \\
= & -\lceil \frac{n}{8} \rceil - \lfloor \frac{n}{2} \rfloor + \lfloor \frac{3n}{4} \rfloor \\
\geq & (\frac{3n}{4} - 1) - \frac{n}{2} - (\frac{n}{8} + 1) \\
= & \frac{n}{8} - 2.
\end{aligned}$$

$\frac{n}{8} - 2 \geq 0$ for all $n \geq 16$. we could use computer or calculator to prove whether the lemma is correct for those cases $3 \leq n < 16$. The results are as follows.

n	$\lfloor \frac{7n}{8} \rfloor - \lfloor \frac{3n}{2} \rfloor + \lfloor \frac{3n}{4} \rfloor$	n	$\lfloor \frac{7n}{8} \rfloor - \lfloor \frac{3n}{2} \rfloor + \lfloor \frac{3n}{4} \rfloor$
3	0	10	0
4	0	11	1
5	0	12	1
6	0	13	1
7	1	14	1
8	1	15	2
9	0		

Thus the inequality holds for all $n \geq 3$. □

Theorem 7 *If the network [n×m, 3×3, m×n] is nonblocking under some packing algorithms, then* $m \geq \lfloor \frac{15n}{8} \rfloor$.

Proof Firstly, we prove that the theorem is true for $n \geq 3$. The proof starts from an empty state.

Add n ax into the empty state: $\mathcal{S}_{ax}$ =[1, n].

Add n by: $\mathcal{S}_{axby}$ =[1, n].

Add n cz: $\mathcal{S}_{axbycz}$ =[1, n].

Delete routes : $\mathcal{S}_{axby}$ =[1, $\lfloor \frac{n}{2} \rfloor$], $\mathcal{S}_{axcz}$ =[$\lfloor \frac{n}{2} \rfloor + 1$, n].

Add $\lfloor \frac{n}{2} \rfloor$ bz: $\mathcal{S}_{axby}$ =[1, $\lfloor \frac{n}{2} \rfloor$], $\mathcal{S}_{axcz}$ =[$\lfloor \frac{n}{2} \rfloor + 1$, n], $\mathcal{S}_{bz}$ =[$n + 1$, $\lfloor \frac{3n}{2} \rfloor$].

Add $\lfloor \frac{n}{2} \rfloor$ cy: $\mathcal{S}_{axby}$ =[1, $\lfloor \frac{n}{2} \rfloor$], $\mathcal{S}_{axcz}$ =[$\lfloor \frac{n}{2} \rfloor + 1$, n], $\mathcal{S}_{bzcy}$ =[$n + 1$, $\lfloor \frac{3n}{2} \rfloor$].

Delete routes: $\mathcal{S}_{ax}$ =[1, n], $\mathcal{S}_{bzcy}$ =[$n + 1$, $\lfloor \frac{3n}{2} \rfloor$].

Add $\lceil \frac{n}{4} \rceil$ bz: $\mathcal{S}_{ax}$ =[1, $\lfloor \frac{3n}{4} \rfloor$], $\mathcal{S}_{axbz}$ =[$\lfloor \frac{3n}{4} \rfloor + 1$, n], $\mathcal{S}_{bzcy}$ =[$n + 1$, $\lfloor \frac{3n}{2} \rfloor$].

Add $\lceil \frac{n}{4} \rceil$ cy: $\mathcal{S}_{ax}$ =[1, $\lfloor \frac{3n}{4} \rfloor$], $\mathcal{S}_{axbzcy}$ =[$\lfloor \frac{3n}{4} \rfloor + 1$, n], $\mathcal{S}_{bzcy}$ =[$n + 1$, $\lfloor \frac{3n}{2} \rfloor$].

Delete routes: $\mathcal{S}_{ax}$ =[1, $\lfloor \frac{3n}{4} \rfloor$], $\mathcal{S}_{bzcy}$ =[$\lfloor \frac{3n}{4} \rfloor + 1$, $\lfloor \frac{3n}{2} \rfloor$].

Add $(\lfloor \frac{7n}{4} \rfloor - \lfloor \frac{3n}{2} \rfloor)$ az: $\mathcal{S}_{ax}$ =[1, $\lfloor \frac{3n}{4} \rfloor$], $\mathcal{S}_{bzcy}$ =[$\lfloor \frac{3n}{4} \rfloor + 1$, $\lfloor \frac{3n}{2} \rfloor$],

$\mathcal{S}_{az}$ =[$\lfloor \frac{3n}{2} \rfloor + 1$, $\lfloor \frac{7n}{4} \rfloor$].

Delete routes: $\mathcal{S}_{ax}$ =[1, $\lfloor \frac{3n}{4} \rfloor$], $\mathcal{S}_{cy}$ =[$\lfloor \frac{3n}{4} \rfloor + 1$, $\lfloor \frac{3n}{2} \rfloor$],

$\mathcal{S}_{az}$ =[$\lfloor \frac{3n}{2} \rfloor + 1$, $\lfloor \frac{7n}{4} \rfloor$]. . . . (*)

Sort nodes: $\mathcal{S}_{ax}$ =[1, $\lfloor \frac{3n}{4} \rfloor$], $\mathcal{S}_{az}$ =[$\lfloor \frac{3n}{4} \rfloor + 1$, $2\lfloor \frac{3n}{4} \rfloor - \lfloor \frac{n}{2} \rfloor$],

$\mathcal{S}_{cy}$ =[$2\lfloor \frac{3n}{4} \rfloor - \lfloor \frac{n}{2} \rfloor + 1$, $\lfloor \frac{7n}{4} \rfloor$].

Since $\lfloor\frac{7n}{4}\rfloor \geq 2\lfloor\frac{7n}{8}\rfloor$, the state can be divided into two parts. The size of each part is greater than or equal to the value $\lfloor\frac{7n}{8}\rfloor$. The desired network state will be like : $\mathcal{S}_{cy} \cup \mathcal{S}_{by}$ =[1, $\lfloor\frac{7n}{8}\rfloor$], $\mathcal{S}_{ax} \cup \mathcal{S}_{az}$ =[$\lfloor\frac{7n}{8}\rfloor + 1$, $\lfloor\frac{7n}{4}\rfloor$].

$| < cy > | = \lfloor\frac{3n}{2}\rfloor - \lfloor\frac{3n}{4}\rfloor$. Therefore, by Lemma 2, the $| < cy > | \leq \lfloor\frac{7n}{8}\rfloor$ and $| < az,\ ax > | \geq \lfloor\frac{7n}{8}\rfloor$. In order to achieve the desired state, $(\lfloor\frac{7n}{8}\rfloor - \lfloor\frac{3n}{2}\rfloor + \lfloor\frac{3n}{4}\rfloor)$ "*by*" are inserted.

Add $(\lfloor\frac{7n}{8}\rfloor - \lfloor\frac{3n}{2}\rfloor + \lfloor\frac{3n}{4}\rfloor)$ *by*: $\mathcal{S}_{ax} \cup \mathcal{S}_{az}$ =[1, $\lfloor\frac{7n}{4}\rfloor - \lfloor\frac{7n}{8}\rfloor$], $\mathcal{S}_{axby} \cup \mathcal{S}_{azby}$ =[$\lfloor\frac{7n}{4}\rfloor - \lfloor\frac{7n}{8}\rfloor + 1$, $2\lfloor\frac{3n}{4}\rfloor - \lfloor\frac{n}{2}\rfloor$], $\mathcal{S}_{cy}$ =[$2\lfloor\frac{3n}{4}\rfloor - \lfloor\frac{n}{2}\rfloor + 1$, $\lfloor\frac{7n}{4}\rfloor$].

Delete routes: $\mathcal{S}_{ax} \cup \mathcal{S}_{az}$ =[1, $\lfloor\frac{7n}{4}\rfloor - \lfloor\frac{7n}{8}\rfloor$], $\mathcal{S}_{by}$ =[$\lfloor\frac{7n}{4}\rfloor - \lfloor\frac{7n}{8}\rfloor + 1$, $2\lfloor\frac{3n}{4}\rfloor - \lfloor\frac{n}{2}\rfloor$], $\mathcal{S}_{cy}$ =[$2\lfloor\frac{3n}{4}\rfloor - \lfloor\frac{n}{2}\rfloor + 1$, $\lfloor\frac{7n}{4}\rfloor$].

Since $| < by,\ cy > | = \lfloor\frac{7n}{8}\rfloor$ and $| < ax, az > | \geq \lfloor\frac{7n}{8}\rfloor$, the number of free "*ay*" is $n - \lfloor\frac{7n}{4}\rfloor + \lfloor\frac{7n}{8}\rfloor = \lfloor\frac{7n}{8}\rfloor - \lfloor\frac{3n}{4}\rfloor$. By $\lfloor\frac{7n}{8}\rfloor \geq \lfloor\frac{3n}{4}\rfloor$, The free connection request "*ay*" is always positive.

Add $(\lfloor\frac{7n}{8}\rfloor - \lfloor\frac{3n}{4}\rfloor)$ *ay*: $\mathcal{S}_{ax} \cup \mathcal{S}_{az}$ =[1, $\lfloor\frac{7n}{8}\rfloor$], $\mathcal{S}_{by}$ =[$\lfloor\frac{7n}{8}\rfloor + 1$, $2\lfloor\frac{3n}{4}\rfloor - \lfloor\frac{n}{2}\rfloor$], $\mathcal{S}_{cy}$ =[$2\lfloor\frac{3n}{4}\rfloor - \lfloor\frac{n}{2}\rfloor + 1$, $\lfloor\frac{7n}{4}\rfloor$], $\mathcal{S}_{ay}$ =[$\lfloor\frac{7n}{4}\rfloor + 1$, $\lfloor\frac{15n}{8}\rfloor$].

The final node should be $\lfloor \frac{15n}{8} \rfloor$ because $(\lfloor \frac{7n}{8} \rfloor - \lfloor \frac{3n}{4} \rfloor) + \lfloor \frac{7n}{4} \rfloor = n + \lfloor \frac{7n}{8} \rfloor = \lfloor \frac{15n}{8} \rfloor$.

Since Lemma 2 only applied for $n \geq 3$, the above proof is only valid for $n \geq 3$. Therefore, two remaining cases, $n = 1$ and $n = 2$ should be considered individually.

Obviously, when $n = 1$ and $m = \lfloor \frac{15}{8} \rfloor = 1$, the theorem is trivial true.

For $n = 2$,

Add two ax into the empty state : $\mathcal{S}_{ax}$ =[1, 2].

Add two by : $\mathcal{S}_{axby}$ =[1, 2].

Add two cz : $\mathcal{S}_{axbycz}$ =[1, 2].

Delete routes : $\mathcal{S}_{axby}$ =[1], $\mathcal{S}_{axcz}$ =[2].

Add one cy : $\mathcal{S}_{axby}$ =[1], $\mathcal{S}_{axcz}$ =[2], $\mathcal{S}_{cy}$ =[3].

From the above, three middle-stage nodes should be needed for the case $n = 2$. In addition, $\lfloor \frac{(15)(2)}{8} \rfloor = 3$ concludes that the theorem is also true for $n = 2$. Therefore, the theorem is true for all n. □

7 Summary and directions of further study

The Beneš theorem [1] identified a family of WSN networks that are not strictly nonblocking. It also raised the question on the existence of any other such networks in the form of $[n{\times}m,\ r{\times}r,\ m{\times}n]$, $r > 2$. Theorem 4 answers this question affirmatively with a nonblocking Alogrithm 1 over the network $[6{\times}10,\ 3{\times}3,\ 10{\times}6]$. The smallest value of $WS(n,\ r)$ previously undetermined was $WS(5,\ 3)$. Theorem 6 finds $WS(5,\ 3) = 9$. Theorem 7 establishes the necessity of $m \geq \lfloor \frac{15n}{8} \rfloor$ for $[n{\times}m,\ 3{\times}3,\ m{\times}n]$ to be *WSN by packing*. Thus the example of $[6{\times}10,\ 3{\times}3,\ 10{\times}6]$ shows that a WSN network is not necessarily WSN by packing. Table 2 shows the modified Table 1 with our results.

<table>
<tr><th>r
n</th><th>2</th><th>3</th><th>4</th><th>5
-6</th><th>7-
5544</th><th>5545-
24024</th><th>24025-
102960</th><th>102961
-
437580</th><th>437581+</th></tr>
<tr><td>2</td><td colspan="9">3</td></tr>
<tr><td>3</td><td>4</td><td colspan="8">5</td></tr>
<tr><td>4</td><td>6</td><td colspan="8">7</td></tr>
<tr><td>5</td><td>7</td><td>9</td><td colspan="7">9</td></tr>
<tr><td>6</td><td>9</td><td colspan="2">10</td><td colspan="6">11</td></tr>
<tr><td>7</td><td>10</td><td colspan="4">12 or 13</td><td colspan="4">13</td></tr>
<tr><td>8</td><td>12</td><td colspan="5">14 or 15</td><td colspan="3">15</td></tr>
<tr><td>9</td><td>13</td><td>15-17</td><td colspan="5">16 or 17</td><td colspan="2">17</td></tr>
<tr><td>10</td><td>15</td><td colspan="3">17-19</td><td colspan="4">18 or 19</td><td>19</td></tr>
</table>

Table 2: Known values and bounds of $WS(n,\ r)$ for $n \leq 10$.

For further research, we conjecture that the network $[n{\times}\lfloor \frac{7n}{4} \rfloor,\ 3{\times}3,\ \lfloor \frac{7n}{4} \rfloor{\times}n]$,

$n \geq 6$, is WSN through some generalization of Algorithm 1. Algorithm 2 is one of the possible generalization of Algorithm 1. On the other hand, it could be interesting to find a new proof of Theorem 4 without the aid of computer. In addition, the SP still needs some improvements. One of the suggestions is using the parallel computing technique to shorten the running time with a multi-CPU super computer.

References

[1] V. E. Beneš, *Mathematical Theory of Connecting Networks and Telephone Traffic*, New York, Academic Press, 1965.

[2] Charles Clos, *A study of non-blocking switching networks*, Manuscript received October 30, 1952.

[3] D. Z. Du, P. Fishburn, B. Gao, and F. K. Hwang, *On 1-rate wide-sense nonblocking for 3-stage Clos networks*, Preprint.

[4] J. Friedman, *A lower bound on strictly non-blocking networks*, Combinatorica 8(2), 1988.

[5] Frank K. Hwang, *The Mathematical Theory of Nonblocking Switching Networks*, Volume 11.

[6] Yu-Ngai Joseph Hui, *Switching and traffic theory for integrated broadband networks*, 1990.

[7] R. P. Kurshan and V. E. Benes, *Wide-sense nonblocking network made of square switches*, Elec. Letters, vol. 17, pp. 697-700, 1980.

[8] Chin-Tau Lea, *Tradeoff of horizontal decomposition versus vertical stacking in rearrangeable nonblocking networks*, IEEE Transactions on Communications, vol. 39, no. 6, June 1991.

[9] Shuo-Yen Robert Li, *Algebraic Switching Theory and Broadband Applications*, Academic Press, 2000.

[10] D. G. Smith, *Lower bound on the size of a 3-stage wide-sense nonblocking network*, Electronics Letters, vol. 13, No. 7, 31st March, 1977.

[11] Yuanyuan Yang, *Wide-sense nonblocking Clos networks under packing strategy*, IEEE Transactions on Computers, vol. 48, no.3, March 1999.