The Number of Rearrangements Performed via Extended Connection Chains for Rearrangeable Clos Networks

SATORU OHTA
Department of Information Systems Engineering, Faculty of Engineering
Toyama Prefectural University
5180 Kurokawa, Imizu-shi, Toyama 939-0398,
JAPAN
https://www.researchgate.net/profile/Satoru_Ohta

Abstract: Various switching networks have been investigated because of their practical importance and theoretical interests. Among these networks, this study focuses on the Clos network. A Clos network is constructed by placing switches in three stages. In the first and third stages, r ($r > 1$) switches are aligned, whereas m ($m > 1$) switches are aligned in the second stage. There are n inputs and m outputs in the first stage. Symmetrically, the third stage switch has m inputs and n outputs. For this configuration, if $n \le m \le 2n-2$, the network is rearrangeable. Though existing connections in a rearrangeable network may block a newly requested connection, the blocking is always removed by rerouting existing connections. An interesting problem arose during this process is how many existing connections must be rearranged: the number of rearrangements. Although the problem has been studied for a long time, the number of rearrangements is not completely clarified for arbitrary combinations of parameters m, n, and r. This study presents a new upper bound on the number of rearrangements for $2 < n \le m < 2n-2$. This bound is derived from the extended connection chain concept proposed in a previous study. Using this concept, the paper first derives from the case where a parameter, s, represents the load on a second-stage switch. Then, the paper presents another new upper bound, which is independent of parameter s. The study shows that the presented upper bound is smaller than the previously known bounds for a certain range of m.

Key Words: switching network, rearrangeable network, edge coloring, algorithm

Received: November 12, 2020. Revised: May 13, 2021. Accepted: June 1, 2021. Published: June 17, 2021.

1 Introduction

A switch is an electric device with multiple inputs/outputs and can establish a connection between an input and an output. A switching network is a technique that constructs a large-scale switch with a smaller amount of hardware by interconnecting multiple small-sized switches. Various studies have been conducted on switching networks because of mathematical interests [1, 2], and important computer and communication system applications.

Switching networks are categorized into blocking and non-blocking networks. Non-blocking networks can establish arbitrary connections between inputs and outputs, whereas blocking networks cannot establish connections between some pairs of input and output. Non-blocking networks are further categorized into several classes [1, 2]. Among them, this study focuses on a rearrangeable network. For a rearrangeable switching network, all connections can be established between inputs and outputs. In spite of this fact, when some connections are already set up on the network, a new connection request may be blocked by existing connections. However, the blocked request can always be unblocked by rearranging some existing connections to alternate routes.

This study investigates the Clos network among switching networks, which was first presented by Clos [3] in 1953. A Clos network is constructed by placing switches in three stages. In the first stage, r ($r > 1$) switches are aligned, whereas the same number of switches are aligned in the third stage. First- and third-stage switches are linked to m ($m > 1$) second-stage switches. Each first-stage switch has n ($n > 1$) inputs, while each third-stage switch has n outputs. For this configuration, if $n \le m \le 2n-2$, then the Clos network is rearrangeable [1, 4]. A rearrangeable Clos network is significant because it can establish connections between inputs and outputs with less hardware than other non-blocking Clos networks.

Rearrangements of existing connections disturb data streams brought by the rerouted connections [5, 6]. To assess the degree of this disturbance, it is meaningful to evaluate how many times existing connections must be rearranged for unblocking. The analysis of the number of rearrangements is also a theoretically interesting challenge.

For a rearrangeable Clos network, several bounds have been reported on the number of rearrangements. The maximum number of rearrangements required for unblocking is often denoted as $\varphi(n, m, r)$. In 1962, Paull [7] studied $n = m = r$ and showed that $\varphi(n,n,n) = n-1$. In a book published in 1965, Beneš [1] showed that $\varphi(n,m,r) \le r-1$. The case of $m = 2n-2$ was also investigated in [8, 9]. Some studies analyzed $\varphi(n, m, r)$ for the case where the load on a second-stage switch is limited to s ($s > 0$) [10, 11]. Here, the notation $\varphi(n, m, r; s)$ is used to emphasize that s influences the number of rearrangements. Ohta [12] recently revealed a previously unknown upper bound on $\varphi(n, m, r)$ for $3n/2 \le m \le 2n-2$. In spite of these efforts, $\varphi(n, m, r)$ have not been completely known for an arbitrary combination of m, n, and r.

To unblock the blocked connection request in a rearrangeable Clos network, it is necessary to identify the existing connections that should be rearranged. The connection chain concept [13] is a powerful tool for this purpose. The concept is also efficient in analyzing the properties of rearrangements [12]. In [14], an extended connection chain, a modified version of the connection chain, was proposed. It was shown that the number of rearrangements was decreased by employing the extended connection chain compared to employing the original connection chain. The effectiveness of an extended connection chain was confirmed through computer simulation for the average number of rearrangements [14]. However, no theoretical analysis of how the extended connection chain affects $\varphi(n, m, r)$ has been conducted.

This paper first presents an upper bound on $\varphi(n, m, r; s)$ by analyzing the property of the extended connection chain. Although a similar bound has been reported in literature [10, 11], the presented bound is obtained by assuming a different rearrangement algorithm from previous studies. The paper then derived a new upper bound on $\varphi(n, m, r)$ for $2 < n \le m < 2n-2$. This bound is independent of s and is not found in any past studies. The bound is also significant as theoretical evidence for the efficacy of the extended connection chain, which has only been empirically evaluated using computer simulation [14]. Finally, the new bound is compared with the previously known bounds. The result shows that the new bound is smaller than the known bounds for a certain range of m, n, and r.

The remainder of the paper is organized as follows. Section 2 presents definitions and fundamentals to understand the contents of the paper. Section 3 reviews related past studies. In Section 4, the extended connection chain concept is explored. The new upper bound on $\varphi(n, m, r)$ is derived in Section 5. Section 6 compares the bounds of this paper with the previously known bounds. Finally, Section 7 concludes the paper.

2 Preliminaries

Fig. 1 shows an example of a Clos network. A Clos network is a three-stage switching network configured by placing r ($r > 1$) switches in the first stage, m ($m > 1$) switches in the second stage, and r switches in the third stage. A switch in the first or second stage is connected to each switch's next stage through a single link. Therefore, each first-stage switch has m outputs for this link configuration, whereas each third stage has m inputs. Also, a second-stage switch has r inputs and r outputs. Each first-stage switch has n ($n > 1$) inputs. Symmetrically, each third-stage switch has n outputs. Parameters n, m, and r determine the scale of a Clos network. Using these parameters, the network is often denoted as $C(n, m, r)$.

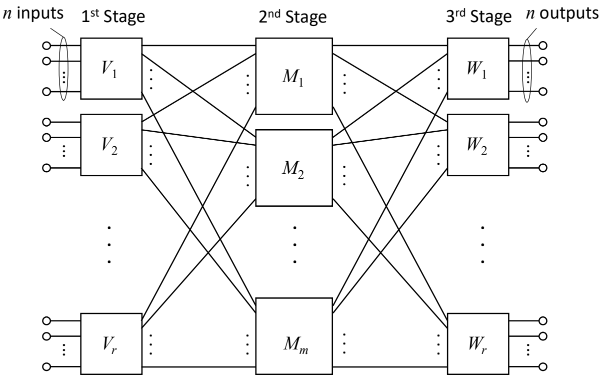

Fig. 1 Clos network C(n, m, r).

This paper assumes a classical switching environment, where every link is used at most one connection, and every connection is unicast. Additionally, it is also assumed that connection requests come one by one. This way of connection request arrivals is called *dynamic* traffic [1]. The paper does not treat *scheduled* traffic, where all requests are simultaneously provided. For the classical switching environment if $n \le m \le 2n-2$, the Clos network is rearrangeable. This means that existing connections may block a newly requested connection, but the blocking is always removed by adequately rearranging some existing connections.

Consequently, arbitrary permutations are achievable between inputs and outputs.

In the following, second-stage switches are denoted as M_1, M_2, …, M_m. Additionally, the first-stage switches are denoted as V_1, V_2, …, V_r, whereas the third-stage switches are denoted as W_1, W_2, …, W_r. Fig. 2 shows an example of blocking and unblocking through rearrangements. In Fig. 2 (a), suppose that a new connection is requested between Input 1 and Output 1. Then, this request was blocked. However, the request was unblocked by two rearrangements. To do this, the connection set up from Input 2 to Output 4 via M_1 is rearranged to go through M_2. Next, the connection set up from Input 5 to Output 5 via M_2 is rearranged to M_1. The result is shown in Fig. 2 (b). As shown in the figure, it has become possible to connect Input 1 to Output 1 through M_1 through rearrangements.

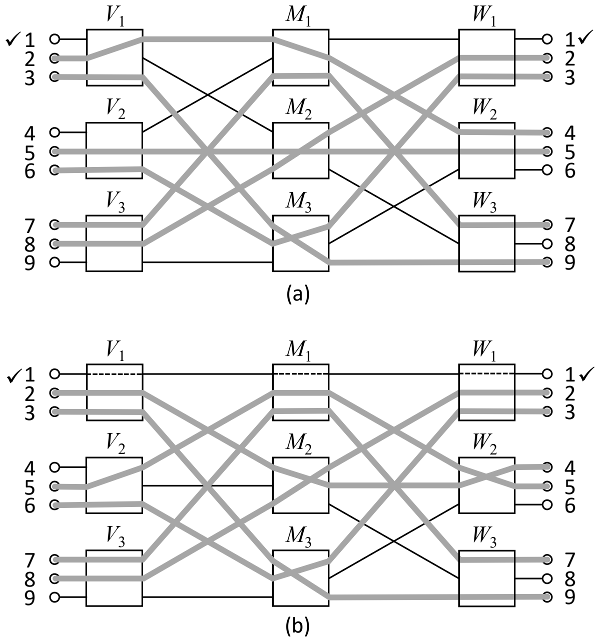

Fig. 2 Rearrangements to unblock a blocked connection request: (a) blocking for a new request between Input 1 and Output 1; (b) request unblocked via rearrangements.

This study investigates how many times existing connections must be rearranged to unblock the blocked request: the number of rearrangements. The number of rearrangements is a function of n, m, and r. Thus, it is denoted as $\varphi(n, m, r)$.

It is known that the state of connections set up in a Clos network is modeled as a bipartite multigraph. Let $\mathcal{V}$ denote the set of first-stage switches V_1, V_2, …, V_r. Let $\mathcal{W}$ denote the set of third-stage switches W_1, W_2, …, W_r. In the multigraph model, $\mathcal{V}$ and $\mathcal{W}$ are sets of vertices. Then, a connection established between a pair of first- and third-stage switches are represented by an edge set up between elements of $\mathcal{V}$ and $\mathcal{W}$. Let $\mathcal{E}$ denote the set of these edges. Since two or more connections may exist between a pair of first- and third stage switches, $\mathcal{V}$, $\mathcal{W}$, and $\mathcal{E}$ constitutes a multigraph $G(\mathcal{V}, \mathcal{W}, \mathcal{E})$. Obviously, from the definition of an edge, $G(\mathcal{V}, \mathcal{W}, \mathcal{E})$ is bipartite. In $G(\mathcal{V}, \mathcal{W}, \mathcal{E})$, the second-stage switch, through which a connection passes, is represented by coloring its associated edge. For example, if the connection goes through the second-stage switch M_c ($1 \le c \le m$), the edge is colored c. Since a single connection can exclusively link between first- and second-stage switches, every edge that stems from the element $\mathcal{V}$ has a distinct color.

Similarly, every edge that stems from the element $\mathcal{W}$ is distinctly colored. Thus, the problem of simultaneously determining the routes of all requested connections is equivalent to the edge-coloring problem of a bipartite multigraph. For example, Fig. 3 shows the relation between the connections in a Clos network (a) and the bipartite multigraph (b).

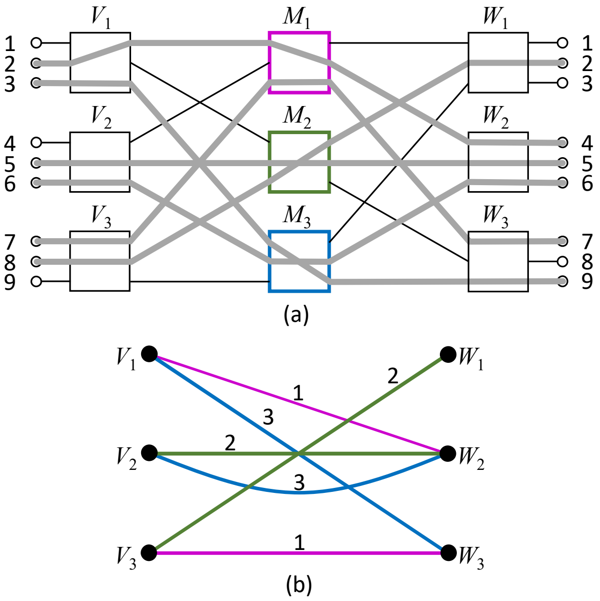

Fig. 3 Example of (a) a connection state and (b) its edge-colored bipartite multigraph model.

Throughout this paper, $\lfloor \bullet \rfloor$ means the largest integer that is smaller than or equal to $\bullet$. Similarly, $\lceil \bullet \rceil$ means the smallest integer that is larger than or equal to $\bullet$. Additionally, for a certain discrete set S, $|S|$ means the number of elements in S.

3 Related Work

In [3], Clos investigated switching networks, later termed *Clos networks*, and analyzed their non-blocking condition as well as hardware amount. A Clos network is configured in three stages, as shown in Fig. 1. Based on the three-stage configuration, the number of stages can be enlarged to any odd number greater than three. The nonblocking condition shown in [3] is $m \geq 2n-1$. This condition assures that an arbitrary connection request is always connectable, not depending on existing connections without using a particular routing algorithm. The condition is called strictly non-blocking (SNB). For a Clos network, if the SNB condition does not hold, blocking will occur. However, blocking can be removed by appropriately rerouting existing connections in the network if $n \leq m \leq 2n-2$. This class of switching networks is called rearrangeable. Since the number of second-stage switches is smaller, the hardware amount is smaller for a rearrangeable Clos network than for an SNB Clos network with the same capacity.

Beneš also comprehensively studied switching networks [1]. In [1], another class of non-blocking Clos networks was also revealed. For this class, a connection request is not blocked if a certain algorithm adequately determines the connection route. This class is called wide sense non-blocking (WSNB). The characteristics of WSNB Clos networks were comprehensively analyzed in [15-17]. These studies suggest that WSNB Clos networks' advantage is slight compared to SNB Clos networks. Another class of non-blocking Clos networks is reported in [18]. For this class, the route for a newly requested connection is determined by a certain algorithm. Additionally, some existing connections may be rerouted when a connection is completed. This class of Clos networks is called repackable networks.

Among the classes of non-blocking Clos networks, a rearrangeable network is important because it can satisfy any connection request without blocking and requires a smaller hardware number. However, rearranging existing connections disrupts data transmission. From the viewpoint of assessing the disturbance degree, it is worthwhile to estimate how many times connections must be rearranged for unblocking. It is also theoretically interesting to estimate the number of rearrangements.

References [1, 2, 7] reported that studies on the number of rearrangements started as early as in the 1950s. An early study on a rearrangeable Clos network is conducted by Paull [7]. Paull investigated the case of $m = n = r$ and showed a rearrangement algorithm as well as $\varphi(n, n, n)$ is $n - 1$. A few years later, Beneš [1] showed that

$$\varphi(n,m,r) \leq r-1 . \tag{1}$$

Gotoh [13] investigated the time-division T-S-T (T: Time, S: Space) switching network, which is equivalent to the Clos network. In [13], the "connection chain" concept is presented. The concept is helpful in concretely representing the existing connections that should be rearranged. It is also effective in analyzing the characteristics of rearrangements. Gotoh derived the same bounds as Beneš's work through the connection chain concept for the number of rearrangements. Although Gotoh's work does not improve the bound over Beneš' result, the bound derivation becomes easier to be understood by employing the connection chain concept.

Nakamura [8] and Bassalygo et al. [9] tackled the number of rearrangements of the special case of $m = 2n - 2$. The latter derived a stronger result,

$$\varphi(n,2n-2,r) = \left\lfloor \log_{n-1} \frac{r(n-2)+1}{2n-3} \right\rfloor + 1 . \tag{2}$$

The number of rearrangements was studied from a slightly different viewpoint in Bassalygo [10] and Hwang and Lin [11]. These works consider the case where s connections are set up through the lightly loaded second stage switch. For this situation, $\varphi(n, m, r)$ depends on load s; thus, is denoted by $\varphi(n, m, r; s)$ to clarify the influence of s. References [10] and [11] showed the upper and lower bounds on $\varphi(n, n, r; s)$.

Recently, Ohta [12] derived a bound on $\varphi(m, n, r)$ for $3n/2 \leq m \leq 2n-2$ as follows.

$$\varphi(n,m,r) \leq \left\lceil \frac{n^2(r-1)}{4(m-n+1)^2} \right\rceil \tag{3}$$

The bound of (3) is derived by assuming a rearrangement algorithm based on Gotoh's connection chain concept. Thus, if a different rearrangement algorithm is assumed, a further smaller bound may be obtained.

Gotoh's connection chain concept was extended by Ohta and Ueda [14]. For the case of employing an extended connection chain to identify which connections should be rerouted, the system may be unblocked through a fewer rearrangements than for employing Gotoh's original connection chain. In [14], computer simulation was performed to estimate the number of rearrangements for employing extended and original connection chains. The simulation result shows that the number of rearrangements reduces by approximately 40% for certain conditions of m, n, and the number of existing connections for using an extended connection chain. However, it was not shown in [14] how the theoretical bound on $\varphi(n, m, r)$ is influenced by the extended connection chain. To further clarify the effectiveness of employing an extended connection chain, a theoretical analysis should be provided along with the estimation by computer simulation.

4 Connection Chain

As described in the previous section, the connection chain concept is an efficient tool to tackle the problems involved in rearrangements. The bipartite multigraph model is useful to describe the concept.

Assume that the state of existing connections is represented by bipartite multigraph $G(\mathcal{V}, \mathcal{W}, \mathcal{E})$. An edge is colored c ($1 \le c \le m$) if the connection goes through the second stage switch M_c. For this situation, if a newly requested connection is blocked between v_1 ($v_1 \in \mathcal{V}$) and v_1^* ($v_1^* \in \mathcal{W}$), let us define two sets of colors as follows.

$$\mathcal{A} \triangleq \{c \mid \text{an edge of color } c \text{ stems from } v_1 \text{ and no edges of color } c \text{ stem from } v_1^*\}, \quad (4)$$

$$\mathcal{B} \triangleq \{c \mid \text{an edge of color } c \text{ stems from } v_1^* \text{ and no edges of color } c \text{ stem from } v_1\}. \quad (5)$$

Trivially from the above definition, $\mathcal{A} \cap \mathcal{B} = \phi$. The third stage switch represented by v_1^* has n outputs, and at least one output, which a newly requested connection occupies, is idle. Therefore, the degree of v_1^* is at most $n - 1$. Meanwhile, there exist m colors and $m \ge n$. Consequently, no edges stem from v_1^* for at least one color, c. Meanwhile, v_1 is the endpoint of an edge with this color because the connection is blocked. Otherwise, it is possible to establish the connection through the second stage switch M_c. Therefore, $\mathcal{A}$ is not empty. Similarly, $\mathcal{B}$ is not empty, either.

Clearly from the above observation,

$$|\mathcal{A}| \ge m - n + 1, \quad (6)$$

$$|\mathcal{B}| \ge m - n + 1. \quad (7)$$

For the simplicity of notation, the following terminologies are defined.

Definition 1: A c-edge is the edge colored c.
Definition 2: The “length” of a path is defined as the number of included edges.

A connection chain is defined as a path in $G(\mathcal{V}, \mathcal{W}, \mathcal{E})$. Let a and b be colors such that $a \in \mathcal{A}$ and $b \in \mathcal{B}$. Then, the path starts from v_1 or v_1^* and is constructed by tracing a-edges and b-edges alternately. Let P and P^* denote paths that start from v_1 and v_1^*, respectively. Let v_1, v_2, v_3,… denote the vertices contained in P. Although multiple edges may exist between v_i and v_{i+1} (i: positive integer), the edge included in P is uniquely identified by its color, denoted by c_i. Thus, path P is represented as

$$P: v_1, c_1, v_2, c_2, v_3, c_3, v_4, \dots, c_K, v_{K+1}$$

where $K \ge 1$. Path P is a connection chain if

$$c_i = \begin{cases} a, & i = 2t-1 \\ b, & i = 2t \end{cases}, \quad (8)$$

where t is a positive integer. Also, path P^* is,

$$P^*: v_1^*, c_1^*, v_2^*, c_2^*, v_3^*, c_3^*, v_4^*, \dots, c_L^*, v_{L+1}^*$$

where $L \ge 1$. Path P^* is a connection chain if

$$c_i^* = \begin{cases} b, & i = 2t-1 \\ a, & i = 2t \end{cases}. \quad (9)$$

Connection chain P terminates at v_{K+1} if no a-edge or b-edge stems from v_{K+1}. Similarly, connection chain P^* terminates at v_{L+1}^* if no a-edge or b-edge stems from v_{L+1}^*.

In the connection chain, unblocking is performed by rearranging the connections modeled by a-edges in P (or P^*) to M_b and connections modeled by b-edges in P (or P^*) to M_a [13]. When rearrangements are executed in P (or P^*), the number of the rearrangements is the length of P (or P^*). It is known that the shorter of P and P^* is no longer than $r - 1$ [12, 13]. This immediately proves that $\varphi(n,m,r) \le r-1$.

An extended connection chain [14] is also defined as a path, which starts from v_1 or v_1^*, and is constructed by tracing a-edges and b-edges alternately. The difference is the endpoint of the path. For extended connection chain P (P^*), the endpoint v_{K+1} (v_{L+1}^*) satisfies either Condition 1 or 2.

Condition 1: no a- or b-edge stems from v_{K+1} (v_{L+1}^*).
Condition 2: no edges with a certain color c stem from v_K and v_{K+1} (v_L^* and v_{L+1}^*).

Of these, Condition 1 is the same as that for the original connection chain concept. Thus, an extended connection chain is constructed by adding a termination condition, Condition 2, to the original connection chain. If the extended connection chain terminates by Condition 2, unblocking is performed as follows. First, the connection modeled by the last *a*- or *b*-edge, set between v_K and v_{K+1} (v_L^* and v_{L+1}^*), is rearranged to M_c. Then, for other $K - 1$ ($L - 1$) edges, the connections modeled by *a*-edges are moved to M_b, whereas those modeled by *b*-edges are moved to M_a. Thus, it is necessary to know which color *c* is used to rearrange the last edge between v_K and v_{K+1} (v_L^* and v_{L+1}^*). When the chain terminates by Condition 1, the rearrangements are executed in the same way as in the original connection chain. Since Condition 2 is added for terminating the path, the length of an extended connection chain is not greater than that of the original connection chain for a certain blocking state. Namely, the number of rearrangements may decrease by employing an extended connection chain.

Property 1: For extended connection chains *P* and P^*, created by colors *a* and *b*,

$$v_i \neq v_j,\ 1 \le i < j \le K+1, \tag{10}$$

$$v_i^* \neq v_j^*,\ 1 \le i < j \le L+1, \tag{11}$$

$$v_i \neq v_j^*,\ 1 \le i \le K+1,\ 1 \le j \le L+1\,. \tag{12}$$

Proof: It is known that (10), (11), and (12) hold for Gotoh's original connection chains [12, 13]. The same relations trivially hold for extended connection chains because every vertex of *P* and P^* is also included in the original connection chains. □

An extended connection chain is found by computational procedure *find_ex_chain*() shown in Fig. 4. The procedure returns the sequence of vertices and edge colors that specify *P* or P^* and color *z* ($z \neq c_1, c_2$) or *c* ($c = c_1$ or c_2). The last element, *z* or *c*, is essential to see which edge color should be used to rearrange the connection modeled by the edge between v_K and v_{K+1}.

```
Procedure find_ex_chain(G, v1, c1, c2)
// Arguments
//     G: bipartite multigraph that models the connections
//     v1: start point of the chain
//     c1: a if v1 ∈ V, b otherwise
//     c2: b if v1 ∈ V, a otherwise
1. K:= 0;
2. c:= c1;
3. while e_{K+1} with color c stemming from v_{K+1} exists do
4.     K:= K + 1;
5.     v_{K+1}:= e_K's endpoint, which is not v_K;
6.     if neither of v_K nor v_{K+1} is the end point of an edge
       with a certain color z
       then return (v1, c1, v2, c2, v3, c1, v4,…, v_{K+1}), z;
7.     if K is odd then c:= c1 else c:= c2;
8. end while
9. return (v1, c1, v2, c2, v3, c1, v4,…,v_{K+1}), c;
```

Fig. 4 Procedure to compute extended connection chain.

Property 2: Procedure *find_ex_chain*() returns an extended connection chain in a finite time.

Proof: Steps 2, 3, and 7 of the procedure select edges e_1, e_2, e_2,… to satisfy (8) if $v_1 \in \mathcal{V}$. Also, the selected edges satisfy (9) if $v_1 \in \mathcal{W}$. Thus, the sequence of vertices and edges satisfies the requirement for the original connection chain. It is known that $v_i \neq v_j$ $(1 \le i < j \le K+1)$ for the original connection chain [9]. Since the number of vertices is $2r$, Step 5 can be executed no more than $2r - 1$ times. This means that the while loop (Steps 4–7) is not executed more than $2r - 1$ times, and thus the procedure is completed in a finite time. If the while loop stops because of the condition in Step 3, Condition 1 holds. If the procedure returns the output because of Step 6, Condition 2 holds. Thus, the returned value is a pair of an extended connection chain and the color used for repainting the last edge. □

5 Bound on $\varphi(n, m, r)$

First, let us focus on a characteristic of extended connection chain *P*, built using colors $a \in \mathcal{A}$ and $b \in \mathcal{B}$. Assume that the length of *P* (P^*) is not shorter than K ($K \ge 1$).

Definition 3: Let K_z denote the number of vertices, each of which is one of the first *K* vertices v_1, v_2, …., v_K in *P* and an endpoint of a *z*-edge ($z \neq a$, $z \neq b$, $1 \le z \le m$). Similarly, let K_z^* denote the number of vertices, each of which is one of the first *K* vertices in P^* and an endpoint of a *z*-edge.

Namely, these K_z vertices are used for *z*-edges. For $1 \le i < K$, at least one of v_i and v_{i+1} must be an endpoint of a *z*-edge. Otherwise, since Condition 2 holds, the length of *P* becomes $K - 1$ or shorter. This contradicts the assumption. Since one of two adjacent vertices is an endpoint of a *z*-edge, K_z will be at least nearly half of *K*. This intuition is concretely confirmed by observing that the state of v_1 and v_K falls into one of the following four cases.

Case 1: Both v_1 and v_K are the endpoints of *z*-edges.
Case 2: v_1 is the endpoint of a *z*-edge, but v_K is not the endpoint of a *z*-edge.
Case 3: Neither v_1 nor v_K is the endpoint of a *z*-edge.
Case 4: v_1 is not the endpoint of a *z*-edge, but v_K is the endpoint of a *z*-edge.

Then, the following property is obtained for each case.

Property 3: K_z is bounded as follows.

$$K_z \geq \begin{cases} (K+1)/2, & \text{Case 1} \\ K/2, & \text{Cases 2 and 4} \\ (K-1)/2, & \text{Case 3} \end{cases}. \qquad (13)$$

Proof: The property is proved by mathematical induction. If K = 1, either Case 1 or 3 occurs. Trivially, K_z is 1 and 0 for Cases 1 and 3, respectively. Thus, (13) is correct for Cases 1 and 3. Cases 2 and 4 occur when $K > 1$. If $K = 2$, K_z is 1 for these cases. Therefore, (13) is also valid for Cases 2 and 4 when $K = 2$.

Next, assume that (1) is correct for $K = k$. Then, let us consider the case of $K = k + 1$.

If P is in Case 1, v_1 is an endpoint of a z-edge. Thus, the portion of v_1, v_2, …, v_k is in Case 1 or Case 2. Thus, at least $k/2$ endpoints of z-edges are contained in this portion. By Property 1, v_{k+1} is different from v_1, v_2, …, v_k, and a z-edge stems from v_{k+1}. Thus,

$$K_z \geq k/2+1=(K+1)/2\,. \qquad (14)$$

If P is in Case 2, a z-edge stem from v_k. Otherwise, Condition 2 is satisfied between v_k and v_{k+1}, and the length of P becomes $K - 1$. This contradicts the assumption. Thus, the portion of v_1, v_2, …, v_k is in Case 1. Thus, the number of endpoints of z-edges is not smaller than $(k + 1)/2$ for this portion. Since v_{k+1} is not an endpoint of a z-edge,

$$K_z \geq (k+1)/2 = K/2\,. \qquad (15)$$

Similarly, a z-edge necessarily stems from v_k if P is in Case 3. For this case, the portion of v_1, v_2, …, v_k is in Case 4 because no z-edge stems from v_1. Thus, this portion contains $k/2$ endpoints of z-edges. Since no z-edge stems from v_{k+1},

$$K_z \geq k/2 = (K-1)/2\,. \qquad (16)$$

if P is in Case 4, the portion of v_1, v_2, …, v_k is in Case 3 or 4. Thus, at least $(k - 1)/2$ endpoints of z-edges are contained in this portion. By Property 1, v_{k+1} is different from v_1, v_2, …, v_k, and a z-edge stems from v_{k+1}. Thus,

$$K_z \geq (k-1)/2+1 = K/2\,. \qquad (17)$$

Therefore, (13) is valid for all cases, and any value of K. □

It is easy to show that (13) also holds for K_z^*. Suppose that s connections pass a second-stage switch M_z. There exist $2s$ endpoints for z-edges that model these connections. From Property 1, each K_z vertices in P is different from any of K_z^* vertices in P^*. Therefore, $K_z + K_z^*$ endpoints of z-edges appear in P and P^*. Thus, $K_z + K_z^*$ is upper bounded by $2s$. Meanwhile, Property 3 asserts that K_z (K_z^*) must be larger if the length of P (P^*) is longer. An upper bound on $\varphi(n, m, r; s)$ is derived from this observation.

Theorem 1: If rearrangements are executed via the shorter of extended connection chains P and P^*, $\varphi(n, m, r; s)$ is upper bounded as follows.

$$\varphi(n,m,r;s) \leq 2s\,. \qquad (18)$$

Proof: First, consider the case where the load on M_a is s. This means that there are s a-edges. In P, a single b-edge always appears after every a-edge. Therefore, the length of P does not excess $2s$. Thus, (18) is correct for this case. Similarly, (18) is also valid when the load on M_b is s.

Next, let us focus on a middle switch M_z, for which $z \neq a$ and $z \neq b$. Suppose that the load on M_z is s. Let K ($K > 0$) denote the shorter extended connection chains P and P^*. By Property 1, there are no duplications among K_z and K_z^* endpoints of z-edges. Additionally, because the blocking occurs between v_1 and v_1^*, at least one of v_1 and v_1^* is the endpoint of a z-edge. Thus, at least one of P or P^* is in Case 1 or 2, whereas the other may be in Case 3. Then, from Property 3,

$$K_z + K_z^* \geq K - 1/2\,. \qquad (19)$$

Since K, K_z, and K_z^* are integers,

$$K_z + K_z^* \geq K\,. \qquad (20)$$

As the sum of K_z and K_z^* does not excess the total number of endpoints of z-edges,

$$2s \geq K_z + K_z^* \geq K\,. \qquad (21)$$

Therefore, the length of P and P^* is not larger than $2s$. This means the number of rearrangements is not larger than $2s$. □

If $m = n$, Theorem 1 eventually coincides with a result shown in [10] and [11]: $\varphi(n,n,r;s) \leq 2s$. However, [10] and [11] derive the result by assuming the rearrangements of different connections. The connection sequence considered in [10] and [11] is also represented as a path in the multigraph. However, the path includes edges colored a, b, and a third color c in the algorithm of [10] and [11]. For the algorithm of these studies, the path starts with a- and b-edges but turns into an alternate sequence of a-and c-edges (or b- and c-edges) in its halfway. By contrast, the extended connection chain considered in this paper consists of the edges strictly with two colors, a and b. Thus, the condition for deriving Theorem 1 is different from that assumed in the study of [10] or [11].

Next, another bound on the number of rearrangements is derived in the form that is independent of s. The new bound on $\varphi(n, m, r)$ is obtained by utilizing the fact that an extended connection chain consists of only a- and b-edges. The following intuition leads to the bound. The number of rearrangements is the length K of P or P^*. However, if K is large for any combinations of the elements in $\mathcal{A}$ and $\mathcal{B}$, the number of a- and b-edges included in P and P^* must also be large. Meanwhile, the number of edges with colors other than a and b is determined by subtracting the number of a- and b-edges from the total number of edges. Here the total number of edges is upper bound by the number of inputs (or outputs). Thus, the number of edges with other colors will decrease if K increases.

Moreover, a decreased number of edges with other colors leads to a smaller value of s for a certain color, not a or b. Here, Theorem 1 asserts that K is upper bounded by $2s$. Consequently, K will not be able to exceed a certain value because of this constraint by s.

The new bound is obtained by considering algorithm *Ex_Unblock*, shown in Fig. 5.

```
Algorithm Ex_Unblock
// Given: $v_1$, $v_1^*$, $\mathcal{A}$, $\mathcal{B}$, the existing connections
// Variables:  $P_{\min}$: the shortest extended connection chain
//             $k_{\min}$: the length of $P_{\min}$
//             $a_{\min}$, $b_{\min}$: elements of $\mathcal{A}$, $\mathcal{B}$ used for $P_{\min}$
1.  $G$:= bipartite multigraph that models the connections;
2.  $k_{\min}$:= $\infty$;
3.  $\mathcal{A}_{\text{sub}}$:= {$m - n + 1$ elements selected from $\mathcal{A}$};
4.  $\mathcal{B}_{\text{sub}}$:= {$m - n + 1$ elements selected from $\mathcal{B}$};
3.  for each $a$ in $\mathcal{A}_{\text{sub}}$ do
4.      for each $b$ in $\mathcal{B}_{\text{sub}}$ do
5.          $(P, z)$ := find_ex_chain($G$, $v_1$, $a$, $b$);
6.          if the length of $P < k_{\min}$ then
7.              $k_{\min}$:= the length of $P$;
8.              $P_{\min}$:= $P$, $a_{\min}$:= $a$, $b_{\min}$:= $b$, $z_{\min} = z$;
9.          end if
10          $(P^*, z)$:= find_ex_chain($G$, $v_1^*$, $b$, $a$);
11.         if the number of edges in $P^* < k_{\min}$ then
12.             $k_{\min}$:= the length of $P^*$;
13.             $P_{\min}$:= $P$, $a_{\min}$:= $a$, $b_{\min}$:= $b$, $z_{\min} = z$;
14.         end if
15.     end for
16. end for
17. In $P_{\min}$, rearrange the connections by $a_{\min}$, $b_{\min}$, and $z_{\min}$;
```

Fig. 5 Algorithm Ex_Unblock that is used to derive a new bound on the number of rearrangements.

Property 4: Algorithm *Ex_Unblock* unblock the blocked request within a finite time $n \le m$. The number of rearrangements is at most $r - 1$.

Proof: As each $\mathcal{A}_{\text{sub}}$ and $\mathcal{B}_{\text{sub}}$ has just $m - n + 1$ elements, the for-loops of Steps 3–16 complete in a finite time. From (6), (7), and $n \le m$, $\mathcal{A}_{\text{sub}}$ and $\mathcal{B}_{\text{sub}}$ are not empty. Thus, Steps 5–14 are executed for at least a pair of a and b ($a \in \mathcal{A}_{\text{sub}}$, $b \in \mathcal{B}_{\text{sub}}$). For this pair, Step 5 finds P because *find_ex_chain*() is executed with feeding v_1 as the start point of the path, and it correctly returns an extended connection chain as assured by Property 2. Similarly, Step 10 finds P^*. Then, the shortest chain is selected by Steps 8 and 13. The system is unblocked by Step 17. When the original connection chain concept is used for a pair of a and b, the number of rearrangements is at most $r - 1$. For the extended connection chain, the length is not greater than that for the original connection chain. Thus, the property is proved. □

Property 5: Let K denote the length of the shorter of P and P^*. Then, at least K a-edges and K b-edges are contained in P and P^*.

Proof: from the definition of extended connection chain, P starts with an a-edge, and then b- and an a-edges appear alternately. Thus, P contains at least $\lceil K/2 \rceil$ a-edges and $\lfloor K/2 \rfloor$ b-edges. Similarly, P^* contains at least $\lceil K/2 \rceil$ b-edges and $\lfloor K/2 \rfloor$ a-edges. As the sum of $\lceil K/2 \rceil$ and $\lfloor K/2 \rfloor$ is K, the property is proved. □

By using the algorithm *Ex_Unblock*, the following theorem is obtained.

Theorem 2: For $2 < n \le m \le 2n - 3$, if algorithm *Ex_Unblock* is used,

$$\varphi(n,m,r) \le \min\left(\left\lfloor \frac{2(nr-1)}{3m-2n+2} \right\rfloor, r-1 \right). \qquad (22)$$

Proof: Let K denote the length of the connection chain that is the shortest among the chains examined by algorithm *Ex_Unblock*. Subset $\mathcal{A}_{\text{sub}}$ obtained in Step 2 has $m - n + 1$ elements. Similarly, $\mathcal{B}_{\text{sub}}$ contains $m - n + 1$ elements. Let us define $\mathcal{Z}$ it as a set of colors not included in $\mathcal{A}_{\text{sub}}$ or $\mathcal{B}_{\text{sub}}$. Namely,

$$\mathcal{Z} \triangleq \{1,2,\ldots,m\} - \mathcal{A}_{\text{sub}} - \mathcal{B}_{\text{sub}}. \qquad (23)$$

From $\mathcal{A} \cap \mathcal{B} = \phi$ and the definitions of $\mathcal{A}_{\text{sub}}$ or $\mathcal{B}_{\text{sub}}$,

$$|\mathcal{Z}| = m - |\mathcal{A}_{\text{sub}}| - |\mathcal{B}_{\text{sub}}| = 2n - m - 2. \qquad (24)$$

Since $m \le 2n - 3$ and $2 < n$, $|\mathcal{Z}| > 0$. By Property 5, the number of a-edges for a single element of $\mathcal{A}_{\text{sub}}$ is not smaller than K. Thus, the total number of a-edges for all elements of $\mathcal{A}_{\text{sub}}$ is at least $(m - n + 1)K$. Similarly, the total number of b-edges included in all elements of $\mathcal{B}_{\text{sub}}$ is at least $(m-n+1)K$. Meanwhile, the total number of edges in G is at most $nr-1$ because the number of the input (or output) ports is nr, and at least one port is idle. Let $N_{\mathcal{Z}}$ denote the total number of the edges colored by elements of $\mathcal{Z}$. From the above observation,

$$N_{\mathcal{Z}} \le nr - 1 - 2(m-n+1)K\,. \tag{25}$$

Let us focus on z-edge, whose color z is included in $\mathcal{Z}$. The average number of z-edges is $N_{\mathcal{Z}} / |\mathcal{Z}|$ for a certain z. Let s denote the minimum of the number of z-edges among all z's in Z. Then, since s is not greater than the average,

$$s \le \frac{nr-1-2(m-n+1)K}{|Z|}\,. \tag{26}$$

From (24) and (26),

$$s \le \frac{nr-1-2(m-n+1)K}{2n-m-2}\,. \tag{27}$$

By Theorem 1, K is not greater than $2s$. Therefore,

$$K \le 2 \cdot \frac{nr-1-2(m-n+1)K}{2n-m-2}\,. \tag{28}$$

By solving (28) for K,

$$K \le \frac{2(nr-1)}{3m-2n+2}\,. \tag{29}$$

Algorithm *Ex_Unblock* unblocks the system by the shortest connection chain. The number of rearrangements is the length of the connection chain and thus an integer. Consequently,

$$\varphi(n,m,r) \le \left\lfloor \frac{2(nr-1)}{3m-2n+2} \right\rfloor. \tag{30}$$

From Property 4 and (30), (22) is obtained. □

6 Comparison with Previous Results

The bound proposed in this paper is smaller than the previously known bounds for a certain range of m, n, and r. This section compares the bound with two previously known bounds. The first one is a classical bound (1) reported in 1965 [1]. As the second one, the recently found bound [12] shown in (3) is also considered. It is not difficult to see that the right side of (22) is smaller than (1) for a certain range of m, n, and r. If $r \ge n$ and $m \ge 4n/3$, it is possible to show that the right side of (22) is smaller than the bound of (1). The right side of (22) is also smaller than (3) for some n, m, and r. For example, assume that $m = 3n/2$ and n is extremely large. For $m = 3n / 2$, the right side of (3) is lower bounded by $n^2(r-1) / (n+2)^2$, thus it is approximately $r-1$ for a large value of n. Meanwhile, the right side of (22) is upper bounded by $4r/5$ and is smaller than that of (3) if $r > 5$.

Figs. 6 and 7 compare the result of this paper with the previous bounds. In these figures, the x-axis is m, the number of second-stage switches, while the y-axis is the upper bound on $\varphi(n, m, r)$, the number of rearrangements. Fig. 6 shows the case of $n = 100$ and $r = 100$ while Fig. 7 shows the case of $n = 10$ and $r = 100$. Fig. 6 shows that the upper bound derived by this paper is smaller than the previously known bounds $134 \le m \le 157$. For $134 \le m \le 149$, the proposed bound is lower than the classical upper bound $r-1$. The proposed bound is also lower than that reported in [12] if $150 \le m \le 157$. The bound of [12] is derived from the original connection chain concept. It is also possible to obtain the bound $r-1$ via the original connection chain [12, 13]. Any smaller bounds have not been found via the original connection chain. Thus, Fig. 6 provides theoretical evidence for the advantage of the extended connection chain over the original connection chain in the number of rearrangements.

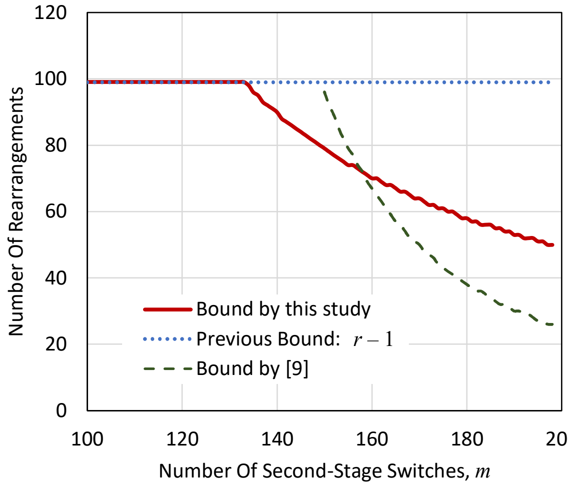

Fig. 6 Comparison between the bound derived in this study and previously known bounds for n = 100 and r = 100.

Fig. 7, which plots a smaller n value case, shows that the advantage of the proposed is obtained for $m = 13$ and 14. For this case, the proposed bound is not advantageous over the bound of [12]. This result suggests that the advantage of the proposed bound over the bound of [12] is obtainable if the value of n is large. In comparison with the classical bound $r-1$, the proposed bound is more advantageous if $m \ge 13$. From these results, it is concluded that the number of rearrangements is upper bounded by a smaller number than the previously known bound $r-1$ for a certain range of m if the extended connection chain is employed. Furthermore, for a certain range of m and a larger value of n, it is also observed that the system is unblocked via a smaller number of rearrangements than the bound derived in [12].

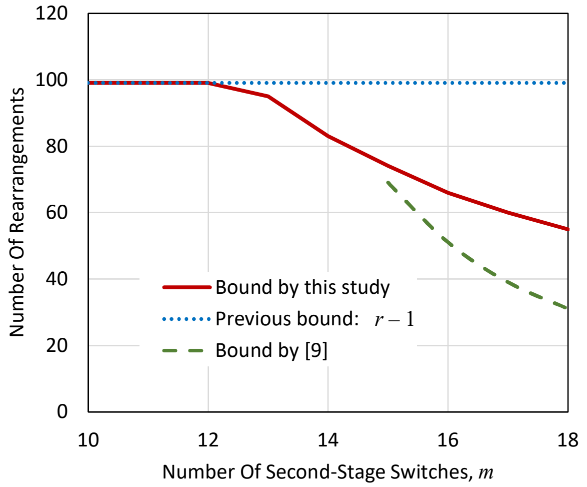

Fig. 7 Comparison between the bound derived in this study and previously known bounds for n = 10 and r = 100.

7 Conclusions

This paper investigated a three-stage rearrangeable Clos network. An unsolved problem of rearrangeable Clos network is to clarify how many existing connections must be rearranged to connect a blocked new connection request. For this problem, the paper presented a new upper bound on the number of rearrangements. The presented bound derived by assuming a rearrangement algorithm that uses the extended connection chain concept proposed in [14]. The paper compares the new bound with previously known bounds. The result shows that the upper bound is tighter than the previous bounds for a certain range of parameters.

As stated above, the result of this paper is obtained by assuming a particular rearrangement algorithm. Thus, by assuming a different algorithm, another improved upper bound may be found on the number of rearrangements. However, it is still an open problem to find the tightest bound for this problem.

References:

[1] V. E. Beneš, *Mathematical theory of connecting networks and telephone traffic*, New York and London, Academic Press, 1965.

[2] F. K. Hwang, *The mathematical theory of non-blocking switching networks*, *2nd Edition*, Singapore, World Scientific, 2004.

[3] C. Clos, A study of non-blocking switching networks, *Bell System Technical Journal*, Vol. 32, No. 2, 1953, pp. 406–424.

[4] A. Jajszczyk, Nonblocking, repackable, and rearrangeable Clos networks: Fifty years of theory evolution, *IEEE Communications Magazine*, Vol. 41, No. 10, 2003, pp. 28–33.

[5] N. Fujii, Application of a rearrangement algorithm for digital cross-connect system control, *in proc. IEEE INFOCOM '89,* Otawa, Canada, 1989, pp. 228–233.

[6] F. K. Hwang, W. D. Lin, V. Lioubimov, On noninterruptive rearrangeable networks, *IEEE/ACM Transactions on Networking*, Vol. 14, No. 5, 2006, pp. 1141–1149.

[7] M. C. Paull, Reswitching of connection networks, *Bell System Technical Journal*, Vol. 41, No. 3, 1962, pp. 833–855.

[8] G. Nakamura, The number of reswitching for a certain 3-stage connecting network, *Electronics and Communications in Japan*, Vol. 50, 1967, pp. 56–63.

[9] L. A. Bassalygo, I. Grushko, V. L. Neiman, The number of reswitching for a 3-stage connecting network, *in proc. ITC6*, Munich, Germany, 1970, pp. 241/1–241/5.

[10] L. A. Bassalygo, On a number of reswitching in a three-stage connecting network, *in proc. ITC7*, Stockholm, Sweden, 1973, pp. 231/1–231/4.

[11] F. K. Hwang, W. -D. Lin, The number of rearrangements in a 3-stage Clos network using an auxiliary switch, In *Switching Networks: Recent Advances* D. -Z. Du, H. Q. Ngo (Eds.), Dordrecht: Kluwer Academic Publishers, 2001, pp. 179–189.

[12] S. Ohta, The number of rearrangements for Clos networks–new results, *Theoretical Computer Science*, Vol. 814, 2020, pp. 106–119.

[13] K. Gotoh, Rearrangement characteristics on time division T–S–T networks, *Transactions of IEICE (B)*, Vol. J61-B, No. 3, 1979, pp. 182–189 (in Japanese).

[14] S. Ohta, H. Ueda, A rearrangement algorithm for three-stage switching networks, *Electronics and Communications in Japan*, Vol. 70, No. 9, 1987, pp. 68–77.

[15] D. G. Smith, Lower bound on the size of a 3-stage wide-sense non-blocking network, *Electronics Letters*, Vol. 13, No. 7, 1977, pp. 215–216.

[16] K. H. Tsai, D. W. Wang, F. Hwang, Lower bounds for wide-sense non-blocking Clos network, *Theoretical Computer Science*, Vol. 261, 2001, pp. 323–328.

[17] F. H. Chang, J. Y. Guo, F. K. Hwang, C. K. Lin, Wide-sense non-blocking for symmetric and asymmetric 3-stage Clos networks under various routing strategies, *Theoretical Computer Science*, Vol. 314, 2004, pp. 375–386.

[18] A. Jajszczyk, G. Jekel, A new concept — repackable networks, IEEE Transactions on Communications, Vol. 41, No. 8, 1993, pp. 1232–1237.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

Satoru Ohta conducted the whole study.

Sources of funding for research presented in a scientific article or scientific article itself

This work was supported by JSPS KAKENHI Grant Number JP19K11928.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)