國 立 交 通 大 學

應用數學系

博 士 論 文

兩種連接網路：三環式網路及 **$\mathrm{Log}_2(N, m, p)$** 交換網路之研究

On Two Interconnection Networks: Triple-loop Networks and Switching $\mathrm{Log}_2(N, m, p)$ Networks

研 究 生：林琲琪

指導教授：黃光明教授

中 華 民 國 九 十 四 年 六 月

兩種連接網路：三環式網路及 Log$_2$(N, m, p) 交換網路之研究

On Two Interconnection Networks: Triple-loop Networks and Switching Log$_2$(N, m, p) Networks

研 究 生：林琲琪　　Student：Bey-Chi Lin

指導教授：黃光明教授　　Advisor：Professor Frank K. Hwang

國立交通大學

應用數學系

博 士 論 文

A Dissertation

Submitted to Department of Applied Mathematics

College of Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Applied Mathematics

June 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年六月

兩種連接網路：三環式網路及 $\mathbf{Log_2(N, m, p)}$ 交換網路之研究

學生：林玤琪　　　　　　　**指導教授：黃光明**

國 立 交 通 大 學 應 用 數 學 系

摘　要

本篇論文主要討論二種類型的網路：一是電腦網路(computer networks)；另一是應用在通訊上的交換網路(switching networks)。對於前者，我們主要針對三環式網路做研究；對於後者，我們則針對$\mathrm{Log}_2(N, m, p)$網路做研究。首先，我們先介紹三環式網路：

一個記為$\mathrm{ML}(N; s_1, \ldots, s_l)$的多環式網路，若以一具$N$個點$(0, 1, \ldots, N-1)$，$lN$條邊的有向圖來表示，其有向邊的連接方式為：$i \to i+s_1, i \to i+s_2, \ldots, i \to i+s_l$, $(\mathrm{mod}\ N)$, $i = 0,1, \ldots, N-1$。其中$s_1, \ldots, s_l$這l個整數被稱做是多環式網路的“步”。當l的值確定時，我們也可稱此多環式網路為l-環式網路。尤其當$l = 2$ 時，此多環式網路又被稱為雙環式網路，記為$\mathrm{DL}(N; s_1, s_2)$；當$l = 3$ 時，此多環式網路則又被稱為三環式網路，記為$\mathrm{TL}(N; s_1, s_2, s_3)$。

近期，雖然有許多的三環式網路已被提出，並且它們的效能也被研究，但是真實存在的此類網路，就數量來說仍是非常的稀少，因此，在此篇論文中，我們將會把已有的三環式網路推廣以增加其類型的數量，同時，我們也會提出一個啟發式(heuristic)的方式來最佳化我們所提出的三環式網路所需的參數，以增進其效能。

在本篇論文中，我們將研究三個特定三環式網路的 3-直徑(3-diameter)，其中，我們用建構的方式來做此研究，亦即，我們在任意二點間建立三條點互斥(node-disjoint)，且長度不超過直徑加 2 的最短路徑。

接下來，我們將介紹$\mathrm{Log}_d(N, m, p)$網路：

Lea 和 Shyy [32] 首先提出含有$N = 2^n$條進線(inputs)和出線(outputs)的$\mathrm{Log}_2(N, m, p)$網路，其建構方式為將p個$\mathrm{BY}^{-1}(n, m)$ 的複製網路垂直堆疊在某一進線層(input stage)和出線層(output stage)中，其中$0 \le m \le n-1$，並且每一進(出)線層含有N個$1 \times p$ (或 $p \times 1$)的閂(crossbar)。之後，Hwang [24]將$\mathrm{Log}_2(N, m, p)$網路中，每個2×2的閂由$d \times d$的閂取代，於是把它推廣為$\mathrm{Log}_d(N, m, p)$網路。

一個網路若目前送來的訊息，必須在所有的訊息皆依某一給定的演算法連接傳送，才可以保證被連接傳送時，這種不阻塞的程度稱為 *wide-sense nonblocking*。網路的交流量被分類為點對點(*point-to-point*)，例如傳統電話連接；另一為廣播式(*broadcast*)，亦即點對所有(one to all)。假如每一訊息的最多接收者有所限制，那麼廣播式亦稱為多重傳播(*multicast*)，亦即點對多(one to many)；如果接收者被限制為f，則稱為f-cast。

Tscha和Lee [44] 對於多重傳播(*multicast*) $\mathrm{Log}_2(N, 0, p)$網路提出了fixed-size window演算法，並表明期望可以將此演算法推廣至$\mathrm{Log}_2(N, m, p)$網路。之後，Kabacinski和Danilewicz [29] 將fixed-size window演算法推廣至variable size window演算法。在這篇論文中，我們更進一步地把variable size window演算法的結果，由$\mathrm{Log}_2(N, 0, p)$網路推廣至$\mathrm{Log}_2(N, m, p)$網路。

On Two Interconnection Networks: Triple-loop Networks and Switching $Log_2(N, m, p)$ Networks

Student: Bey-Chi Lin **Advisor: Frank K. Hwang**

Department of Applied Mathematics

National Chiao Tung University

HsinChu 30050, Taiwan, Republic of China

Abstract

This thesis is divided into two types of networks: computer networks and switching networks used in communication. In particular, we will study a class of computer networks called the triple-loop network, and a class of switching networks called $Log_2(N, m, p)$. We first introduce the former.

A multi-loop network, denoted by $ML(N; s_1, \ldots, s_l)$, can be represented by a digraph on N nodes, 0, 1, …, $N - 1$ and lN links of l types: $i \rightarrow i + s_1$, $i \rightarrow i + s_2$, …, $i \rightarrow i + s_l$, (mod N), $i = 0,1, \ldots, N - 1$. The integers $s_1, \ldots, s_l$ are called the steps of the multi-loop network. When l is specified, we can also call it an l-loop network. In particular, when $l = 2$, the multi-loop network is usually called the double-loop network and is denoted by $DL(N; s_1, s_2)$. When $l = 3$, the multi-loop network is usually called the triple-loop network and is denoted by $TL(N; s_1, s_2, s_3)$.

Several triple-loop networks have been recently proposed and their efficiency studied. However, the number of cases for which one of these networks exist is sparse. In this thesis, we extend these networks to larger classes to enhance their realizability. We also give a heuristic method to optimize the network parameters to increase their efficiency.

In this thesis, we study the k-diameters of three specific triple-loop networks. In particular, we construct three node-disjoint shortest paths no longer than the diameter

plus 2 for any pair of nodes.

Next we introduce the $\text{Log}_2(N, m, p)$ network.

Lea and Shyy [32] first proposed the $\text{Log}_2(N, m, p)$ network with $N = 2^n$ inputs (outputs), which consists of a vertical stacking of p copies of $\text{BY}^{-1}(n, m)$, $0 \le m \le n-1$, sandwiched between and connected to an input stage and an output stage, each with N $1 \times p$ (or $p \times 1$) crossbars. Later, Hwang [24] extended the $\text{Log}_2(N, m, p)$ network to $\text{Log}_d(N, m, p)$ network by replacing the 2×2 crossbars with $d \times d$ crossbars.

A network is *wide-sense nonblocking* (WSNB) if the connection of the current request is assured only when all connections are routed according to a given algorithm. Traffic can be classified as *point-to-point*, like 2-party phone calls, or *broadcast*, which is one to all. If there is a restriction on the maximum number of receivers per request, then broadcast is called *multicast* (one to many), or f-cast, if that number is specified to be f.

Tscha and Lee [44] proposed a fixed-size window algorithm for the multicast $\text{Log}_2(N, 0, p)$ network and expressed a desire to see its extension to the $\text{Log}_2(N, m, p)$ network. Later, Kabacinski and Danilewicz [29] generalized the fixed-size window to variable size to improve the results. In this thesis, we further extend the variable-size results from the $\text{Log}_2(N, 0, p)$ network to $\text{Log}_2(N, m, p)$.

誌　謝

這本論文對我來說不僅僅只是呈現研究上的些許結果，更是蘊涵著這五年來，自己用心走過的每一個足跡。論文的完成，過程中有著許多人的支持和鼓勵，心裡著實有許多的感謝對每一個曾經參與過我生命的人，雖然只能用隻字片語去表達，但希望當中滿滿的感激你們都能明白。

首先，最要感謝我的指導教授—黃光明老師，是老師深厚的學術實力，帶領我接觸各樣的領域，讓我得以站在他的肩膀上，看到更寬廣的世界，卻也同時深深地體會學海無涯；是老師的循循善誘、提攜後進，讓我發現如此渺小的自己，原來也可以解決國際性的問題，體驗過挑戰未知的興奮，才逐漸明白推動一個研究者不斷前進的動力來源；是老師對學術的認真投入，讓我看到一位學者的風範，當老師用教育的傳承來回饋這片孕育他成長的土地時，我更明白在研究這條路上，自己可以堅持些什麼，以及可以回報這個社會些什麼，雖然未來會有很多的困難等著我去面對，但老師亦教導我，看到問題，等同於看到了希望！

即使最初是很意外地踏入組合數學這個領域，但很謝謝陳秋媛、傅恆霖、黃大原、張鎮華、翁志文等老師們的教導，和許多生活上的幫忙，讓我這個不是學習表現最好的學生，經過這些年從您們身上的學習，也能覺得是滿載而歸。

謝謝林文鏈學長在我剛接觸連接網路這塊陌生的領域時，給予很多的指導和幫忙，讓我有勇氣踏入這個領域探索更多。謝謝君逸和飛黃這二位天才學弟，還有李珠矽老師和惠蘭共同在研究上的相互砌磋，你們的存在，讓我知道走在這條路上，我並不孤單，甚至可以享受研究的樂趣。

謝謝莊慧如老師的舞蹈教學，讓我在學校生活中，找到另一片自己的小天地，得以悠遊其中，我承認對這每個禮拜僅僅一小時練舞時間的重視，不亞於對系上的必修課程，甚至對於有機會站上耗資數十萬元的華麗舞台表演，享受當藝人的快感，更是畢生難忘！並深感榮幸！

謝謝阿珮、Alice 這二個影響我至深的好朋友，在每一個階段，用愛澆灌著我，不間斷地支持、鼓勵和教誨，陪伴我走過許多的風雨。如果今天我學得一絲一毫如何去愛身邊的人，明白絲毫愛的真諦，都是她們的功勞！共同經歷過的每一個過程，都讓我刻骨銘心！

謝謝幼婷、豆豆、秀琴一路來的陪伴，每一份關係從衝突到溝通，到現在的

彼此欣賞，這些漫長的過程或許當下很煎熬，但我很珍惜每一份浴火重生的友誼。

謝謝陳依、雅卿、美玉、昇達、柏盛、人星、世學、Alan 這群一同在校園團契奮鬥的戰友，在你們身上我得到的遠比我能給予的多。

謝謝建廷和我曾經一同經營男女朋友關係，即使關係回到原點，但這半年的點滴，依舊讓我感激在心。

謝謝新竹教會這個大家庭裡的每一個弟兄姊妹。在這裡，我學到最多的是認識神，以及對生命的堅持；在這裡，我看到愛如何真真實實地活在我們當中。新竹，是我屬靈的故鄉，是我新生命的起點！

謝謝同在一研究室的吟衡，一起經歷準備資格考的熬煉，體驗十年寒窗的艱辛，一同歡呼通過資格考的狂喜。在這間小小的研究室裡，有我們的交心、彼此打氣、天南地北的聊天，當然也有互吐苦水的真實片段。SA331 是我的另外一個家，裡頭有著這些年在新竹滿滿的回憶。榮譽室友－班榮超學長的加入，更讓這間研究室增色不少！

謝謝過去在台中師院的同窗好友－玉仙和慧如，即使大學畢業後我們各奔東西，但是你們不間斷地為我打氣，所給的支持，是我最感窩心的。

謝謝曾經在女二舍的歷屆室友：秀貞、姿瑩、曾翊、淑娟、敏慈、彥君，雖然只是短暫的交集，卻豐富了我的生活許多，不論是期待聽趣聞的心情、或是躺在床上閉著眼睛聊到三更半夜的片段、或是生平第一次的寢聚，都讓我對女二這棟冰冷的建築物，添進了許多的情感。

最後，謝謝我的家人，他們總是在背後默默地支持我，尤其是我的父母，含辛茹苦地建立這個家，在我的成長過程中，給我絕對的信任，全力的支持以及廣大的自由度去做任何的嚐試。我知道當我可以無後顧之憂地在這個世界闖蕩時，是因為他們願意放手讓我去飛；我知道當我失意、遭遇挫折，可以隨時回家好好休息時，是因為有他們對我無條件的關愛！

何其有幸，伴隨我成長的老師和朋友很多很多，感謝的話，怎麼說都說不完，紙短情長，僅將這本論文獻給我所摯愛的你們！

結束了二十多年來的學生生涯，畢業，是我人生另一個階段的開始。期許自己不斷抱持著被磨練的態度去學習，懷抱著回饋這個社會的使命去努力，讓未來的每一步都能走得踏實。

Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Motivation

This thesis is divided into two types of networks: computer networks and switching networks used in communication. In particular, we will study a class of computer networks called the triple-loop network, and a class of switching networks called $\mathrm{Log}_2(N, m, p)$. We first introduce the former.

A fundamental limitation of high-performance computer systems is the low rate at which data can be accessed and restored in the high-speed memory. To overcome this limitation, it is current practice to increase the parallelism of operation of the high-speed memory by incorporating several independent memory modules into the memory system. In [45], Stone describes a particular organization of a multimodule memory, designed to facilitate parallel block transfers. A device called the memory circulator is utilized. It consists of a bank of interconnected register, one for each memory, and control circuitry. Each register is connected to l other registers, and the connection pattern has cyclical symmetry. A pattern is completely determined by the selection of l different links. The problem is to select a set of links that will minimize the maximum and/or average number of register-to-register transfers required to achieve an arbitrary circulation. One can assume that one of the l links always connects the original register to an adjacent register. (See [41].)

A multi-loop network, denoted by $\mathrm{ML}(N; s_1, \ldots, s_l)$, can be represented by a digraph on N nodes, $0, 1, \ldots, N - 1$ and lN links of l types: $i \to i + s_1$, $i \to i + s_2$, $\ldots$, $i \to i + s_l$, (mod N), $i = 0,1, \ldots, N-1$. The integers $s_1, \ldots, s_l$ are called the steps of the multi-loop network. When l is specified, we can also call it an l-loop network. In particular, when $l = 2$, the network is usually called the double-loop network. When $l = 3$, the multi-loop network is usually called the triple-loop network. The double-loop network has been extensively studied in the literature (see [25] for a recent survey) as an interconnecting network for either processors or memories in parallel computing [20], or as a local area computer networks [38], or as a large area communication network like SONET [39].

It is known that if gcd(N, s_1, …, s_l) = 1, then an l-loop network is l-connected, hence (l − 1)-fault tolerant, has relatively short diameter and other desirable properties (to be described in chapter 2). Several triple-loop networks have been recently proposed and their efficiency studied. However, they exist only under very restrictive conditions on network parameters. In this thesis, we extend these networks to larger classes to enhance their realizability. We also give a heuristic method to optimize the network parameters to increase their efficiency.

Traditionally, connectivity and diameter were studied separately. Then various approaches have been proposed to study these two parameters together. One such approach led to the notion of k-diameter which was formalized and popularized in Hsu [21] and Hsu and Luczak [22]. The *k-diameter* of a digraph is the minimum length l such that there exist k node-disjoint paths no longer than l. In this thesis, we will study the k-diameters of these networks. In particular, we construct three node-disjoint shortest paths no longer than the diameter plus 2 for any pair of nodes.

Next we introduce the $\text{Log}_2(N, m, p)$ network.

In an s-stage network, crossbars are lined up into s columns, each called a stage. Switching networks composed of $\log_2 N$ stages are of great interest in both high-speed electronics and photonic switching. Define the states of a network as the set of all possible routings of all legitimate frames, legitimate means the load generated by each input and output terminal does not exceed its capacity; a frame means all requests are in a given session. A set of requests is routable if there exists a set of link-disjoint paths connecting the requests. A state is blocking if there exists a legitimate new request not routable in the current state, and is *nonblocking* otherwise. To obtain nonblocking characteristics, two methods have been proposed: horizontal cascading (HC) and vertical stacking (VS) [5, 31].

The HC method results in greater number of stages between each inlet–outlet pair. More stages in a switching network induce greater signal attenuation in the case of photonic switching or greater delay in the case of electronic switching. For the VS method the question is how many copies of Log_2N switching networks are to be connected in parallel to obtain nonblocking operation of the whole switching network. The number of copies needed in the case of space-division switching networks and point-to-point connections was given in [32, 40].

Lea and Shyy [32] first proposed the $\text{Log}_2(N, m, p)$ network (when $m = 0$, we denote it as a multi-Log_2N network) with $N = 2^n$ inputs (outputs), which consists of a vertical stacking of p copies of $\text{BY}^{-1}(n, m)$, $0 \le m \le n-1$, sandwiched between and connected to an input stage and an output stage, each with N $1 \times p$ (or $p \times 1$) crossbars.

Apart from point-to-point connections, many services, for instance video-conference, video-distribution, multi-party communications, etc., will require connections from one input to many or even all outputs [35, 33, 23]. Nonblocking multicast multi-Log_2N networks were first considered in [43]. Later, this result was improved in [44], where nonblocking operation of multi-Log_2N switching networks was given, provided a special control algorithm, called a window algorithm, is used.

Tscha and Lee [44] stated in conclusion that whether their approach could be extended to $\text{Log}_2(N, m, p)$ (to be defined in chapter 4) was unclear. Kabacinski and Danilewicz [29] generalized the window algorithm from fixed size to variable sizes. Danilewicz and Kabacinski [13, 14] also made an attempt to extend their results to $\text{Log}_2(N, m, p)$, but encountered some difficulties. In this thesis, we will give such an extension for the variable window-size algorithm by adopting a channel graph blockage analysis first used by Shyy and Lea [40] on a single-cast network. We also determine the optimal window size for given m, and then compare the performance among different m.

1.2 Overview of the thesis

In chapter 2, we will give the architecture of multi-loop networks. Some most studied topics of multi-loop networks: minimum distance diagram (MDD) and the tesselatibility of MDD shapes are also introduced. Later, we present some known classical results of existence conditions between L-shape (hyper-L) tile and double-loop (triple-loop) networks, respectively.

In chapter 3, we first generalize the three classes of triple-loop networks studied in the literature to larger classes. Later, we construct the wide-diameters for each of these enlarged classes.

In chapter 4, we first give the architecture of $\mathrm{Log}_d(N, m, p)$ networks. Then the blockingness and channel graph are introduced. Next, we present the classical WSNB results for multicast $\mathrm{Log}_d(N, m, p)$ networks. Later, we provide a new result using window algorithm which was first proposed by Tscha and Lee [44]. At last, we determine the optimal window size and the optimal number of extra stages.

Chapter 2 Preliminaries and Classical Results of Multi-loop Networks

2.1 Architecture

Multi-loop networks were first proposed by Wong and Coppersmith [47] for organizing multimodule memory services. Fiol et al. [20] slightly extended its definition in their study of the data alignment problem in SIMD processors. Nowadays, it is used for both local area computer networks [36, 38] and large area communication networks like SONET [15, 39]. Multi-loop network architectures present an attractive topology for local networks [18, 36, 37], since they require simple control software and interfaces. They permit effective operation at higher data rates and over larger distances than broadcast busses since they do not suffer from carrier sense limitations.

In a unidirectional single loop network with N nodes, (see Fig. 2.1.1) the host computers are connected to the networks via loop interface hardware. Each node i is connected to node $i + 1$ (mod N) to form a completer loop, and messages are passed from node to node along unidirectional links. There are no routing decisions to be made and there is thus no need for central control. A node simply transmits its message to the next node in the loop, and the message circulates around the network until it reaches the destination node. The interface hardware must be able to identify messages intended for its host.

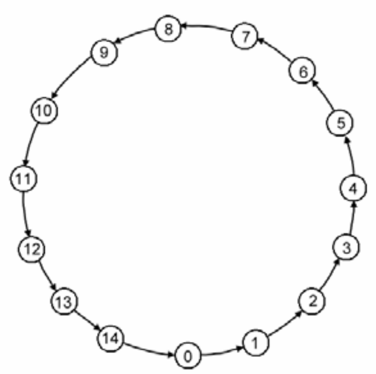


Fig. 2.1.1 Single Loop Network.

An important issue in loop networks is the control mechanism used for message transmission. This mechanism can be centralized or distributed. A distributed control mechanism seems to be more advantageous in terms of performance and reliability as there is no single central node responsible for networks operation. Newhall loop [18] and Pierce loop [37] are two access control mechanisms in common use for loop networks, and the delay insertion register mechanism [36, 45, 46] combines the best features of the first two schemes.

There are several important issues to be studied in the design and analysis of loop networks architectures. The important characteristics of loop networks include the maximum delay for any message, the average delay, reliability, node processing overhead, and the saturation throughput. These performance measures are all interdependent and are related to the network topology. In particular, the three performance measures: reliability, delay, and nodal processing limitation, are affected by network size. There are two approaches to improve reliability. One is to bring all the interfaces to a central point. The other is to introduce link redundancy, i.e. there exist several alternate paths for communication between a pair of nodes.

Raghavendra and Silvester [38] studied various loop networks architectures. Here, we take two architectures for 2-loop and 3-loop networks, respectively, for example. Distributed Double Loop Computer Network (DDLCN) was proposed by Liu [36], and is the topology of the SONET ring (see Fig. 2.1.2). In this network with N nodes, each node i is connected to $i + 1$ (mod N) and $i - 1$ (mod N) nodes. With these redundant links, the network can sustain single interface failures.

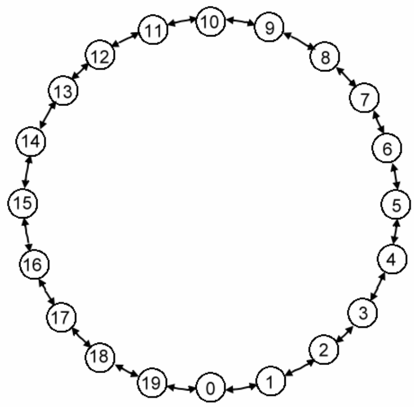


Fig. 2.1.2 Distributed Double Loop Computer Network-DDLCN.

In terms of mathematical form, a multi-loop network, denoted by ML(N; s_1, …, s_l), can be represented by a digraph on N nodes, 0, 1, …, $N - 1$ and lN links of l types: $i \to i + s_1$, $i \to i + s_2$, …, $i \to i + s_l$, (mod N), $i = 0,1, \ldots, N - 1$. The integers s_1, …, s_l are called the steps of the multi-loop network. When l is specified, we can also call it an l-loop network. In particular, when $l = 2$, the multi-loop network is usually called the double-loop network and is denoted by DL(N; s_1, s_2). Thus, DDLCN is denoted by DL(N; 1, $N - 1$). When $l = 3$, the multi-loop network is usually called the triple-loop network and is denoted by TL(N; s_1, s_2, s_3).

2.2 Minimum Distance Diagram

A minimum distance diagram MDD(v) for DL(N; s_1, s_2) is a two-dimensional array which gives the shortest paths from node v to every other node. Since DL(N; s_1, s_2) is node-symmetric, we need only study MDD(0), or simply, MDD. Let node 0 occupies cell (0, 0) in an MDD. Then node v occupies cell (i, j) (i is the column index and j the row index) if and only if $is_1 + js_2 \equiv v \pmod N$ and $i + j$ is the minimum among all (i', j') satisfying the congruence, equality is broken by minimizing i. Namely, a shortest path from 0 to v is through taking i s_1-steps and j s_2-steps (in any order). Fig. 2.2.1 gives the MDD of DL(16; 1, 7).

Wong and Coppersmith [47] gave an O(N) time construction of MDD by sequentially adding nodes to the diagram which can be reached from node 0 in k steps for k = 0, 1, ..., until every node appears exactly once. They also proved that an MDD for a double-loop network is an L-shape which includes the degenerate form of a rectangle. It can be characterized by six parameters l, h, m, n, p, q (4 of them independent) (see Fig. 2.2.2). Thus, we denote it by L(l, h, n, p). This L-shape plays a crucial role in proving many desirable properties for DL(N; s_1, s_2).

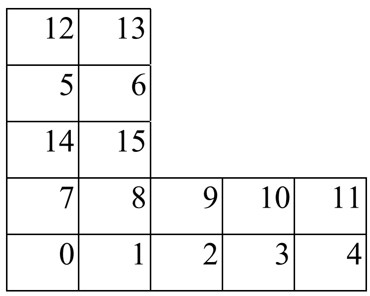


Fig. 2.2.1 An MDD(0) of DL(16; 1, 7).

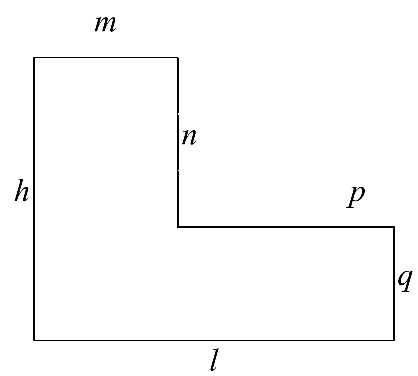


Fig. 2.2.2 An L-shape.

The MDD for a triple-loop network is a three-dimensional array with each step in the x_i-axis signifying an s_i-step. Unfortunately, the MDD does not have a uniform nice shape like the L-shape (see Fig. 2.2.4, Fig. 2.2.6, Fig. 2.2.8) and this fact has hampered the study of triple-loop networks. Aguiló et al. [3] overcame this difficulty by skipping the triple-loop network and going directly to a nice three-dimensional shape which they called hyper-L tile. Later, Aguilo-Gost [4] identified two other shapes which she named H_1 and H_2 (see Fig. 2.2.5 and Fig. 2.2.7). For convenience, we use H_0 (see Fig. 2.2.3) to denote the hyper-L shape.

Note that H_0 is characterized by three parameters l, m, n, and is highly structured and symmetrical, where l, m, n are integers, $m \geq n \geq 0$ and $l > m + n$. H_1 and H_2 are characterized by three parameters $\{h, m, n\}$ and $\{l, m, n\}$, respectively, where l, h, m, n are positive integers. Thus, we also use $H_0(l, m, n)$, $H_1(h, m, n)$ and $H_2(l, m, n)$ to denote H_0, H_1 and H_2, respectively.

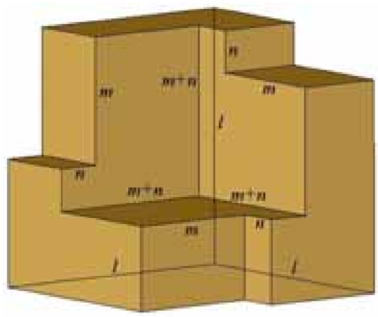


Fig. 2.2.3 $H_0(l, m, n)$.

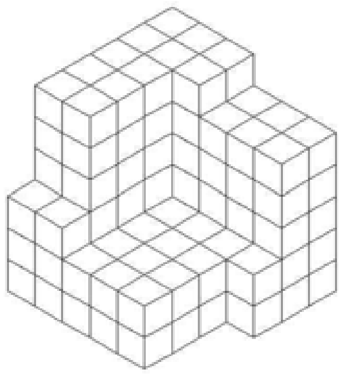


Fig. 2.2.4 MDD of TL(134; 33, 15, 19).

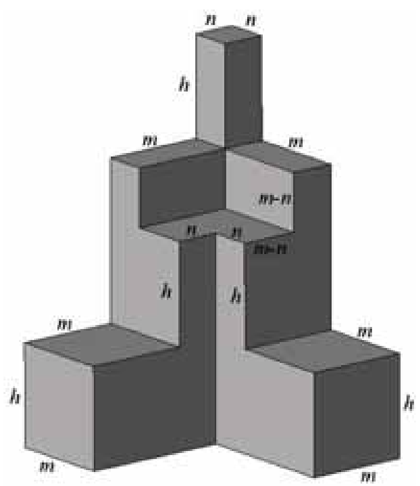


Fig. 2.2.5 $H_1(h, m, n)$.

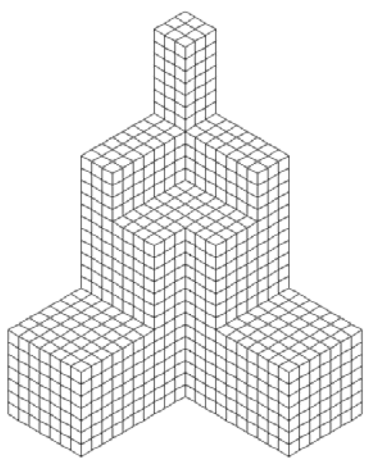
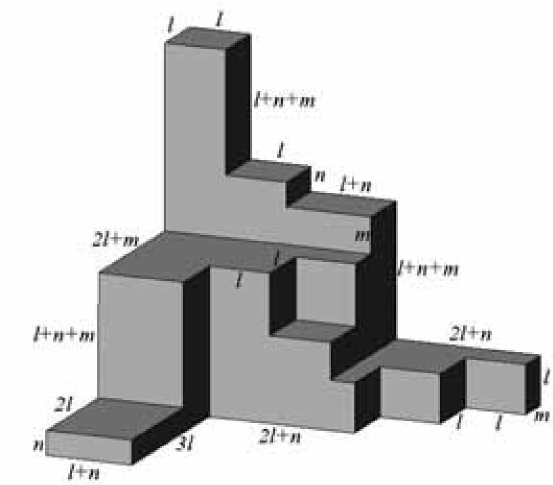


Fig. 2.2.6 MDD of TL(2277; 12, −250, 51).



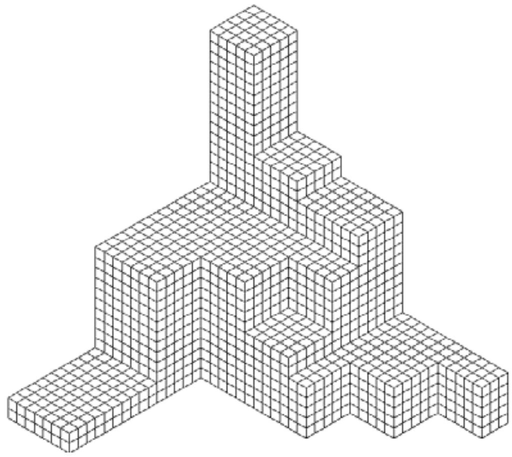


Fig. 2.2.7 $H_2(l, m, n)$. Fig. 2.2.8 MDD of TL(4097; −59, −110, 256).

Besides, suppose that $\Re^d$ is divided into unit hypercubes and a shape is a connected set of hypercubes. A shape is said to tessellate $\Re^d$ if any number of it can be fitted together with neither gaps nor overlapping (rotation or reflection not allowed). Fiol et al. [20] observed that an L-shape always tessellates the plane (see Fig. 2.2.9) regardless of the L-shape is degenerate or not. Agulió-Gost [4] showed the 3D tessellation of hyper-L (see Fig. 2.2.10).

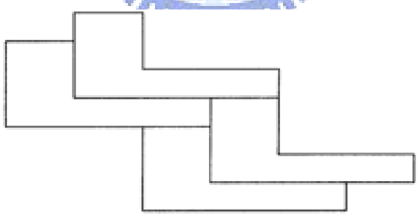


Fig. 2.2.9 L-shape tessellates the plane.

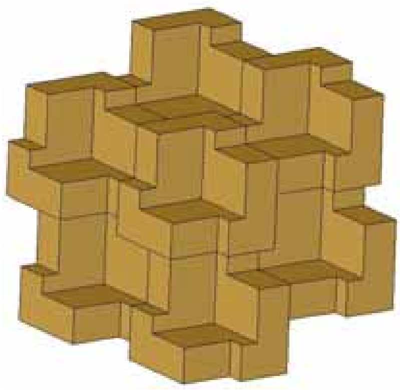


Fig. 2.2.10 Generical 3D tessellation of H_0.

Chen et al. [11] gave a sufficient condition for a shape to tessellate. The following result follows as a special case.

Theorem 2.2.1 Every MDD tessellates $\Re^d$.

For $H_0(l, m, n)$, Aguiló et al. [3] used the tesselatibility of the MDD shape to yield an 3 × 3 matrix which characterizes the interrelation among the locations of the same node (say, node 0) in several adjacent copies of the MDD. We use M_0 to denote this characterizing matrix.

$$M_0 = \begin{pmatrix} l & -m & -n \\ -n & l & -m \\ -m & -n & l \end{pmatrix}$$

Namely, each row vector represents the steps to go from one node 0 to another. For example, the first row represents that after we use l s_1-steps, $-m$ s_2-steps (− denotes the opposite direction) and $-n$ s_3-steps, we can go from one node 0 to another.

By the same way, we define the characterizing matrices of $H_1(h, m, n)$ and $H_2(l, m, n)$ as follows:

$$M_1 = \begin{pmatrix} n & n & 2h \\ -m & n+m & h \\ -m & -m & h+m-n \end{pmatrix}, \; M_2 = \begin{pmatrix} 2l+n & l+m & l+n \\ 3l+n & -2l & l \\ -2l-n & l & l+m+2n \end{pmatrix}.$$

The diameter of a triple-loop network is the maximum distance among pairs of nodes in the network. Let $N(D)$ denote the maximum number of nodes in a triple-loop network with diameter D. Hyper-L tiles were proven to be an effective tool to obtain lower bounds for $N(D)$. In particular, Aguiló et al. [3] used the H_0 to obtain

$$N(D) \geq \tfrac{2}{27}(D+3)^3 \approx 0.074D^3.$$

Agulió-Gost [4] used the H_1 to obtain

$$N(D) \geq \tfrac{1485}{27^3} D^3 \approx 0.075D^3,$$

and used the H_2 to obtain

$$N(D) \geq \tfrac{860}{22^3} D^3 \approx 0.08D^3.$$

For convenience of comparison, the efficiency of a triple-loop network TL is defined [4] as

$$\mathrm{E}(\mathrm{TL}) = \frac{N}{D^3}.$$

2.3 Existence Conditions

Unfortunately, not every L-shape (hyper-L) tile has a double-loop (triple-loop) network realizing it; see [10] for examples. Thus it becomes important to determine when a L-shape (hyper-L) tile has a double-loop (triple-loop) network realizing it. Fiol et al. [20] (also see Chen and Hwang [9]) proved

Theorem 2.3.1 Necessary and sufficient conditions that L(l, h, n, p) can be implemented is that $l > n$, $h \geq p$ and gcd(l, h, n, p) = 1.

By noting the locations of cells containing node 0 (as specified by M), they obtained the following equations:

$$ls_1 - ns_2 \equiv 0 \pmod N,\ -ps_1 + hs_2 \equiv 0 \pmod N. \tag{2.3.1}$$

Note that (2.3.1) can also be written as

$$\begin{pmatrix} l & -n \\ -p & h \end{pmatrix}\begin{pmatrix} s_1 \\ s_2 \end{pmatrix} = N\begin{pmatrix} \alpha \\ \beta \end{pmatrix}, \text{ or } \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} = \begin{pmatrix} h & n \\ p & l \end{pmatrix}\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

for some integers α, β. Fiol et al. [2, 17] proposed the Smith normalization method to solve for s_1 and s_2. They proved:

Theorem 2.3.2 There exists unimodular, integral 2 × 2 matrices L and R such that

$$L\begin{pmatrix} l & -p \\ -n & h \end{pmatrix}R = S = \begin{pmatrix} 1 & 0 \\ 0 & N \end{pmatrix} \text{ (the Smith normal form).}$$

Furthermore, let

$$L = \begin{pmatrix} w & x \\ y & z \end{pmatrix}$$

Then DL(N, y, z) implements L(l, h, n, p) and (y, z) is unique up to isomorphism.

The computation of L and R involves solving for q_1, q_2 in $q_1u - q_2v = 1$ for various pairs of (u, v).

For general L(l, h, n, p), Chen and Hwang [9] gave the following method to find s_1 and s_2.

For $k = 0, 1, \ldots$, defines

$$s_{1_k} = h + kn, \qquad s_{2_k} = p + kl.$$

Let F_k denote the set of prime factors of $\gcd(s_{1_k}, s_{2_k})$ and F the set of prime factors of N. They used the sieve method in number theory to show the existence of a k such that $f \notin F_k$ for all $f \in F$. Then (s_{1_k}, s_{2_k}) is a solution of (2.3.1). For L(6, 4, 3, 2), we easily find the solution $s_1 = h = 4$ and $s_2 = p = 3$.

Next, we discuss the existence conditions for some triple-loop networks. A triple-loop network with a hyper-L shape is called a hyper-L triple-loop. Fiol [19] proposed two necessary conditions for the existence of an $H_0(l, m, n)$ triple-loop:

(i) gcd (N, l, m, n) = 1, and

(ii) gcd (2 × 2 minors of M_0) = 1.

Chen et al. [10] showed that (ii) implies (i) for H_0 and gave a necessary and sufficient condition.

Theorem 2.3.3 A necessary and sufficient condition for the existence of an $H_0(l, m, n)$ triple-loop network is $\gcd(l^2 - mn, m^2 + ln, n^2 + lm) = 1$.

Furthermore, for a TL(N; s_1, s_2, s_3) with $H_0(l, m, n)$ shape, if it satisfies the conditions of Theorem 2.3.3, then the solution of (s_1, s_2, s_3) is $(l^2 - mn, m^2 + ln, n^2 + lm)$ unique up to the equivalence defined by a permutation of (s_1, s_2, s_3) or a multiplication of (s_1, s_2, s_3) by a scalar.

Let M be a 3 × 3 integral matrix with $|\det(M)| = N > 0$. Fiol [19] defined G(M) as the Cayley diagraph of the group Z^3/MZ^3 with the generator set $\{e_1, e_2, e_3\}$, where $e_1 = (1, 0, 0)^T$, $e_2 = (0, 1, 0)^T$, $e_3 = (0, 0, 1)^T$. Chen and Hung [8] used Cayley diagraph to derive the necessary and sufficient conditions for the existence of $H_1(h, m, n)$ and $H_2(l, m, n)$ triple-loops as follows.

Lemma 2.3.4 G(M) is isomorphic to a triple-loop network TL(N; s_1, s_2, s_3) with

$$\mathrm{M}\begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix} = 0 \pmod{N}$$

if and only if gcd(all the 2 × 2 minors of M) = 1.

Apply Lemma 2.3.4 to H_1 and H_2, they obtained

Theorem 2.3.5 A necessary and sufficient condition for the existence of an $H_1(h, m, n)$ triple-loop is $\gcd(m, n) = 1$ and $3 \nmid m - n$.

Theorem 2.3.6 A necessary and sufficient condition for the existence of an $H_2(l, m, n)$ triple-loop is $\gcd(l, m, n) = 1$.

Chapter 3 Further Research on Triple-loop Networks

3.1 Generalizing and Fine Tuning H_1 and H_2

Aguilo, Fiol and Garcia [3] used the computer search to find some good MDDs for *l*-loop networks. Of course, the computer search works only for very small *N*. Then they looked at those good MDDs and tried to identify their shapes to grow it to larger *N* but keeping the shape. The method of growing is to use the tesselatibility of the MDD shape to yield an $l \times l$ matrix M which characterizes the interrelations of the locations of the same node in several adjacent copies of the MDD. For a given shape S, we define F(S) as a family of all shapes obtained from S by varying the parameters of S.

Such an approach encounters three problems. The first is that although the original shape is derived from a triple-loop network, there is no guarantee a member of F(S) also corresponds to a triple loop. Thus one has to check the existence of such a triple-loop. Necessary and sufficient conditions for existence were given in section 2.4 in principle.

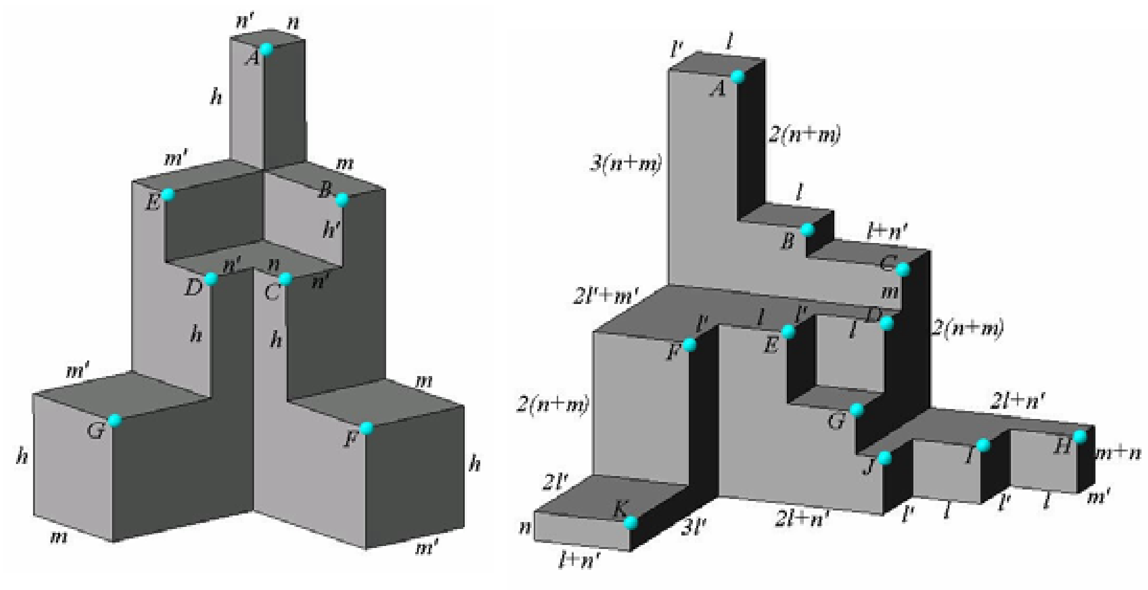
The second problem is that there are not many known good shapes to work with, and the existence of a given shape is sparse.

The third problem is that there is no systematic way to optimize the parameters of a given shape.

In this section, we [34] propose ways to alleviate problems 2 and 3. We will represent H_1 and H_2 each by a 6-parameter family, thus significantly enhancing the chance of finding H_1 or H_2 in the neighborhood of a given *N*. We also propose a method for sub-optimal selection of parameters. The price we pay is that the necessary and sufficiency condition for the existence of a corresponding triple-loop network becomes messy.

We generalize H_1 and H_2 to H_1' and H_2' by allowing some line segments which have the same length to have different lengths. We mark the new parameters in Fig.

3.1.1. Note that all parameters of H_1' and H_2' are larger than or equal to 1. For H_1', $m \geq n$ and $m' \geq n'$.



H₁′ H₂′

Fig. 3.1.1 H_1', H_2'.

It can be verified that H_1' tessellates R^3 with

$$M_1' = \begin{pmatrix} n & n' & 2h \\ -m & n'+m' & h \\ -m & -m' & h+h' \end{pmatrix}.$$

H_1 is the special case of H_1' by setting $m' = m$, $n' = n = h'$.

We apply the necessary and sufficient conditions given in [8] for the existence of a triple-loop network to H_1':

gcd (determinants of the nine minors of M_1')

$= \gcd((n' + 2m')h + (n' + m')h', (n' + 2m')h + n'h', (n' + 2m')h, mh', (n' + 2m')h + nh', (n + 2m)h, (n' + 2m')m, nm' - n'm, (n' + m')n + mn')$

$= \gcd(m'h', n'h', (n' + 2m')h, mh', nh', (n + 2m)h, (n' + 2m')m, nm' - n'm, (n' + 2m')n)$ (3.1.1)

$= 1$

$\Rightarrow \gcd(m', n', m, n) = 1,$ (3.1.2)

$$\gcd(h, h', m, n) = 1, \tag{3.1.3}$$

(3.1.1) is reduced to

$\gcd(h', (n' + 2m')h, (n + 2m)h, (n' + 2m')m, nm' - n'm, (n' + 2m')n)$ by (2)

$= \gcd(h', n' + 2m', (n + 2m)h, nm' - n'm) = 1$ by (3.1.3) (3.1.4)

The farthest nodes from the base node of H_1' must be at one of the circled node. Their distances are:

$d(\mathrm{A}) = n + n' + 3h + h'$,

$d(\mathrm{B}) = n + m + n' + 2h + h'$,

$d(\mathrm{C}) = n + m + m' + 2h$,

$d(\mathrm{D}) = m + n' + m' + 2h$,

$d(\mathrm{E}) = n + n' + m' + 2h + h'$,

$d(\mathrm{F}) = 2m + n + m' + h$,

$d(\mathrm{G}) = m + 2m' + n' + h$.

Our heuristic method sets all these distances equal. Thus

$d(\mathrm{A}) = d(\mathrm{B}) \Rightarrow h = m$,

$d(\mathrm{B}) = d(\mathrm{C}) \Rightarrow h' = m' - n'$,

$d(\mathrm{C}) = d(\mathrm{D}) \Rightarrow n = n'$,

$d(\mathrm{D}) = d(\mathrm{E}) \Rightarrow h' = m - n$.

Summarizing, we have

$$h = m = m',\ n = n' \text{ and } h' = m - n.$$

Therefore in the suboptimal setting $\tilde{H}_1$, there are only two independent parameters m and n, and the diameter is $4m + n$.

Note that for this suboptimal version, necessary and sufficient conditions for the existence of a corresponding triple-loop network is induced from (3.1.2), (3.1.3), (3.1.4) to $\gcd(m, n) = 1$.

Efficiency of $\tilde{H}_1$ is

$$\mathrm{E}(\tilde{\mathrm{H}}_1) = \frac{N}{D^3} = \frac{4m^3 + 6m^2 n - n^3}{(4m+n)^3}.$$

Setting $m = kn$, then n can be canceled out and

$$\frac{N}{D^3} = \frac{4k^3 + 6k^2 - 1}{(4k+1)^3}.$$

$$\frac{d}{dk}\left(\frac{N}{D^3}\right) = \frac{12k(k+1)}{(4k+1)^3} - \frac{(4k^3 + 6k^2 - 1)\cdot 12}{(4k+1)^4} = 0$$

$$\Rightarrow k(k+1)(4k+1) = 4k^3 + 6k^2 - 1$$

$$\Rightarrow k^0 = \frac{1+\sqrt{5}}{2} \approx 1.5$$

Hence we choose $n = 2$ and $m = 3$ for integrality,

$$\mathrm{E}(\tilde{\mathrm{H}}_1) = \frac{26}{343} > 0.07580.$$

Setting $n = 3$ and $m = 5$ yields a slightly better efficiency $923/23^3 \approx 0.07856$. Recall that the efficiency of H_1 is $1485/27^3 \approx 0.075$.

It can be verified that H_2' tessellates R^3 with

$$\mathrm{M}_2' = \begin{pmatrix} 2l+n' & l'+m' & m+2n \\ 3l+n' & -2l' & m+n \\ -2l-n' & l' & 2m+3n \end{pmatrix}.$$

H_2 is the special case of H_2' by setting $m' = m$, $n' = n$, and $l = l' = m' + n'$.

Again, we apply the necessary and sufficient conditions given in [8] for the existence of a triple-loop network to H_2':

gcd (determinants of the nine minors of M_2')

$= \gcd(-(8m' + 11n')l' - (3m' + 4n')n,\ (2l' + n)(3m' + 5n'),\ m'l' + 4n'l' + nn',\ -(5m' + 7n')l,\ -(n' + m')l - (2m' + 3n')m,\ (3m' + 5n')l + (n' + m')m,\ (n + l')l,\ (n + 2l')(m + 2l),\ 7ll' + 3nl + 3ml' + nm)$

$= \gcd((8m' + 11n')l' + (3m' + 4n')n,\ (2l' + n)(3m' + 5n'),\ m'l' + 4n'l' + nn',\ (5m' + 7n')l,\ (n' + m')l + (2m' + 3n')m,\ (3m' + 5n')l + (n' + m')m,\ (n + l')l,\ (n +$

$2l')(m + 2l), 7ll' + 3nl + 3ml' + nm)$

$= \gcd(3nm' - 5n'l' + 4m'l', 3nm' - 14n'l' - nn', m'l' + 4n'l' + nn', 7n'l + 5m'l, 4n'l + 2m'l - mm' - 2mn', 4n'l + 5mm' + 5mn', nl + ll', nl + 3ll' + ml', 4ll' + 3ml' + mn)$

$= \gcd(3nm' - 5n'l' + 4m'l', 3nm' + m'l' - 10n'l', m'l' + 4n'l' + nn', 7n'l + 5m'l, 6n'l - 5mm' - 10mn', 4n'l + 5mm' + 5mn', nl + ll', 2ll' + ml', 3nl + 5ll' - mn)$

$= \gcd((5n' + 3m')l', 3nm' + m'l' - 10n'l', m'l' + 4n'l' + nn', (7n' + 5m')l, 10n'l - 5mn', 4n'l + 5mm' + 5mn', (n + l')l, (2l + m)l', 2nl + mn)$

$= \gcd((5n' + 3m')l', (3n + 7l')m', (7l' + 3n)n', (7n' + 5m')l, 5(2l - m)n', 5(7n' + 5m')m, (n + l')l, (2l + m)l', (2l + m)n) \quad (3.1.5)$

$= 1$

The farthest nodes from the base node of H_2' must be at one of the circled node. Their distances are:

$d(A) = l' + l + 6n + 5m,$

$d(B) = l' + 2l + 3n + 3m,$

$d(C) = l' + 3l + n' + 3n + 3m,$

$d(D) = l' + 3l + n' + 3n + m' + 2m,$

$d(E) = 2l' + 2l + n' + 3n + m' + 2m,$

$d(F) = 3l' + l + n' + 3n + m' + 2m,$

$d(G) = 2l' + 3l + n' + 2n + m' + m,$

$d(H) = 5l + 2n' + n + m' + m,$

$d(I) = l' + 4l + 2n' + n + m' + m,$

$d(J) = 2l' + 3l + 2n' + n + m' + m,$

$d(K) = 5l' + l + n' + n + m'.$

Our heuristic method sets all these distances equal except $d(B)$. Thus

$d(A) = d(C) \Rightarrow 3n + 2m = 2l + n',$

$d(C) = d(D) \Rightarrow m' = m',$

$d(D) = d(E) \Rightarrow l' = l,$

$d(\mathrm{F}) = d(\mathrm{G}) \Rightarrow l = n + m$.

Summarizing, we have

$$l = l' = m + n,\ n = n' \text{ and } m = m'.$$

Therefore in the suboptimal setting $\tilde{\mathrm{H}}_2$, there are only two independent parameters m and n, and the diameter is $8n + 7m$.

Note that for this suboptimal version, necessary and sufficient conditions for the existence of a corresponding triple-loop network is induced from (3.1.5) to gcd $(m, n) = 1$.

Efficiency of $\tilde{\mathrm{H}}_2$ is

$$\mathrm{E}(\tilde{\mathrm{H}}_2) = \frac{N}{D^3} = \frac{40n^3 + 110n^2m + 96nm^2 + 27m^3}{(8n+7m)^3}.$$

Setting $m = kn$, then n can be canceled out and

$$\frac{N}{D^3} = \frac{27k^3 + 96k^2 + 110k + 40}{(7k+8)^3}.$$

$$\frac{d}{dk}\left(\frac{N}{D^3}\right) = \frac{(81k^2 + 192k + 110)}{(7k+8)^3} - \frac{(27k^3 + 96k^2 + 110k + 40)\cdot 21}{(7k+8)^4} = 0$$

$$\Rightarrow 6k^2 + k - 10 = 0$$

$$\Rightarrow k^0 = \frac{-1+\sqrt{241}}{12} \approx 1.2$$

Hence we choose $n = 5$ and $m = 6$ for integrality,

$$\mathrm{E}(\tilde{\mathrm{H}}_2) = \frac{44612}{551368} > 0.08091.$$

Recall that the efficiency of H_2 is $860/22^3 \approx 0.08$.

3.2 Wide-Diameter of H_0

Traditionally, connectivity and diameter were studied separately. Then various approaches have been proposed to study these two parameters together. One such approach led to the notion of k-diameter which was formalized and popularized in Hsu [21] and Hsu and Luczak [22]. The *k-diameter* of a digraph is the minimum length l such that there exist k node-disjoint paths non longer than l. Clearly, the 1-diameter is just the usual diameter D. Note that the k-diameters give a complete description of the interplay between the connectivity and the diameter. It also automatically provides the information if f faults occur for $1 \le f < k$, then the diameter of the surviving graph, the fault-tolerant diameter, does not exceed the k-diameter.

In this section, we [27] will prove that H_0 is 3-connected by constructing 3 node-disjoint paths from any node i to any other node j. A set P of k node-disjoint paths from i to j with lengths $l_1 \le l_2 \le \ldots \le l_k$ is called a *minimum-k-routing* if for any such set of paths with lengths $l_1' \le l_2' \le \ldots \le l_k'$ we have $l_i \le l_i'$ for $i = 1, \ldots, k$. P is called a *weak minimum-k-routing* if $(l_1, l_2, \ldots, l_k)$ is lexicographically shorter than $(l_1', l_2', \ldots, l_k')$. Further, P is *oblivious* if the routing from i to j depends only on i and j. In this paper we give an oblivious weak minimum-3-routing for an arbitrary pair (i, j) and show that a minimum-3-routing does not exist. From the weak minimum-k-routing, we derive an upper bound of the k-diameter. In particular, the 3-diameter is at most $D + 2$.

For convenient, let $H_0(N; s_1, s_2, s_3)$ denote the $TL(N; s_1, s_2, s_3)$ with $H_0(l, m, n)$ shape. Let $H_0(0)$ denote the MDD(0) of $H_0(N; s_1, s_2, s_3)$. By Theorem 2.3.1, we have known that every MDD(0) of triple-loop networks always tessellates $\Re^3$. One consequence is that there exists another shape $H_0^*(0)$ with base 0 located at cell $(l - m - n, l - m - n, l - m - n)$, which is adjacent to $H_0(0)$ in the tessellation (see Fig. 3.2.1).

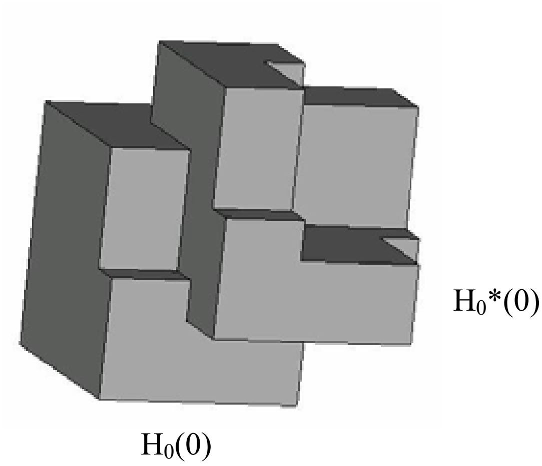


Fig. 3.2.1 $H_0(0)$ and $H_0{*}(0)$.

A *dimension routing* from node u to node v means first taking all steps in one dimension (same s_i), then all steps in a second dimension, then all steps in a third dimension. For example, a dimension routing from node 0 to a node at (x_1, x_2, x_3) with the dimension order (3, 1, 2) takes the $x_3\,s_3$-steps first, then the $x_1\,s_1$-steps and finally the $x_2\,s_2$-steps. Note that a dimension routing always yields a shortest path.

Since TL(N; s_1, s_2, s_3) is node-transitive, it suffices to consider paths from node 0 to an arbitrary node v with coordinates (v_1, v_2, v_3) in $H_0(0)$.

Theorem 3.2.1 There exists an oblivious weak minimum-3-routing from node 0 to an arbitrary node v in H_0.

Proof. Suppose v occupies cell (v_1, v_2, v_3) in $H_0(0)$. We consider three cases:

(i) $v_i > 0$ for $i = 1, 2, 3$. We use dimension routing. The dimension order for path 1 is (1, 2, 3), for path 2 is (2, 3, 1) and for path 3 is (3, 1, 2) (see Fig. 3.2.2). Then clearly, the three paths are node-disjoint and each has length $v_1 + v_2 + v_3$ which is the distance from 0 to v.

Since the lengths of these 3 paths are equal to the distance from 0 to v, it's obvious that the paths we construct constitute a minimum-3-routing.

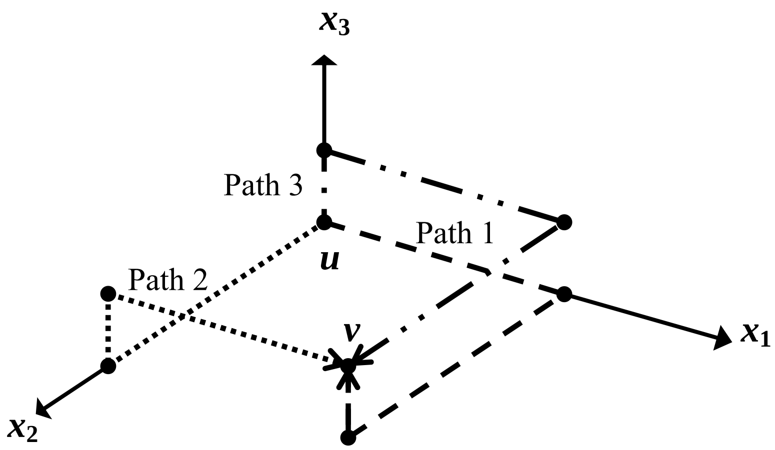


Fig. 3.2.2 Dimension routing for $v_1 > 0$, $v_2 > 0$, $v_3 > 0$.

(ii) Exactly one $v_i = 0$ (say v_3). We use dimension routing in the $x_3 = 0$ plane (where v lines) with orders (1, 2) and (2, 1), respectively, to obtain two node-disjoint paths to v. The third path will be routed through the node $u \equiv v - s_3 \pmod{N}$ as a penultimate node. Suppose u is not in the $x_3 = 0$ plane. Then path 3 is obtained by a dimension routing from node 0 to u starting with s_3-steps. Since path 3 uses only nodes not in the $x_3 = 0$ plane in $H_0(0)$, it is node-disjoint from paths 1 and 2.

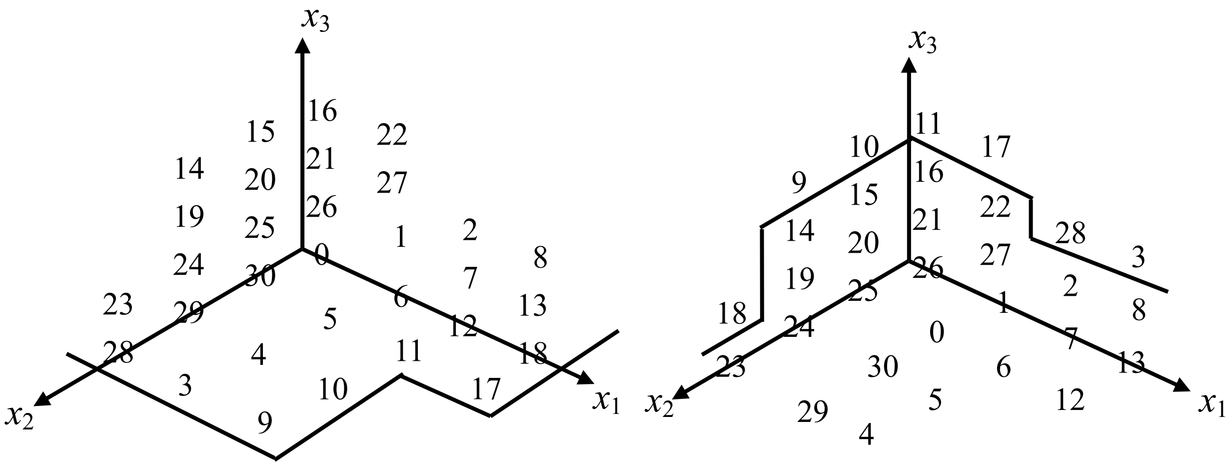
Call a node x occupying cell (x_1, x_2, x_3) in $H_0(0)$ *1-maximal* if cell (x_1+1, x_2, x_3) is not in $H_0(0)$. Similarly we can define *2-maximal* and *3-maximal*. Then u must be 3-maximal in $H_0(0)$ or v would lie in a plane $x_3 = k > 0$ in $H_0(0)$, contradicting our assumption that $v_3 = 0$.

Suppose u is in the $x_3 = 0$ plane. From the fat that u is 3-maximal, necessarily, $l - m - n = 1$. Hence v occupies cell $(v_1 + 1, v_2 + 1, v_3 + 1)$ in $H_0{}^*(0)$.

Path 3 starts with an s_3-steps and enter cell (0, 0, 1), which can be treated as the base of $H_0(s_3)$. It is easily verified that $H_0(s_3)$ can be obtained from $H_0(0)$ by moving nodes on the boundary of the $x_1 = 0$ and $x_2 = 0$ planes (see Fig. 3.2.3).

It u is not in the $x_3 = 1$ plane (the floor plane in Fig. 3.2.3 (b)), implying

u is a boundary node of the $x_3 = 0$ plane, then path 3 uses only nodes not in the $x_3 = 0$ plane, except u, which is not on paths 1 or 2. Hence path 3 is node-disjoint from paths 1 and 2.



(a) (b)

Fig. 3.2.3 (a) and (b) are $H_0(0)$ and $H_0(26)$, respectively, for $l - m - n = 1$, where $N = 31$, $s_1 = 6$, $s_2 = -1$, $s_3 = -5$, $l = 4$, $m = 2$, $n = 1$.

Suppose u is in the $x_3 = 1$ plane. Since v occupies cell $(v_1 + 1, v_2 + 1, v_3 + 1)$, u must occupy cell $(v_1 + 1, v_2 + 1, v_3)$ in $H_0(0)$ and hence also cell $(v_1 + 2, v_2 + 2, v_3 + 1)$ in $H_0{}^*(0)$, which is also in $H_0(s_3)$. Note that paths 1 and 2 enclose a rectangle $1 \le x_1 \le v_1+1$, $1 \le x_2 \le v_2 + 1$ in $H_0(s_3)$, and u is outside of it. Hence a path from s_3 to u using either the (1, 2) or the (2, 1) dimension routing bypasses the rectangle and consequently is node-disjoint with paths 1 and 2. Path 3 is completed by adding the steps from 0 to s_3 and from u to v.

Since the lengths of paths 1 and 2 are equal to the distance from 0 to v, these two paths are shortest. Further, all shortest paths must start and end either with an s_1-step or an s_2-step (any combination allowed). Therefore a third disjoint path must start and end with an s_3-step, i.e., the second node of the path is s_3 and the penultimate node is u. Since our proposed third path uses dimension routing from s_3 to u, it is shortest among the set of third disjoint paths given that the first second paths are shortest. Hence the proposed routing is a weak minimum-3-routing.

(iii) Exactly two $v_i = 0$ (say, $v_3 = v_2 = 0$). Path 1 is the unique shortest path from node 0 to v along the x_1-axis. Let $u \equiv v - s_3 \pmod N$ and $w \equiv v - s_2 \pmod N$. We will show that in $H_0(0)$ one of u and w has $x_2 > 0$ and the other $x_3 > 0$. Then we let path 2 go from 0 to s_2, followed by a dimension routing to the node in $\{u, w\}$ with $x_2 > 0$ (in fact, the dimension routing starts with dimension 2, hence is also a dimension routing from 0). Similarly, path 3 goes from 0 to s_3 followed by a dimension routing (starting from dimension 3) to the other node in $\{u, w\}$. Let L_i, $i \in \{2, 3\}$ denote the set of paths whose last step is a s_i-step. Then a weak minimum-3-routing must have one path from L_2 and one from L_3. But our proposed paths constitute a shortest pair from L_2 and L_3 since they use dimension routing. This proves weak minimum-3-routing.

To prove the existence of the desirable u and w, we first prove a lemma which locates u and w in $H_0(0)$. Among the six permutations of (s_1, s_2, s_3) mentioned in Theorem 3.2.1, call $(s_1 = a, s_2 = b, s_3 = c)$, $(s_1 = b, s_2 = c, s_3 = a)$, $(s_1 = c, s_2 = a, s_3 = b)$ type 1 and the other three permutations type 2, where $a = l^2 - mn$, $b = m^2 + ln$, $c = n^2 + lm$.

Lemma 3.2.2 Let $v = (v_1, 0, 0)$.

(i) Suppose $0 \le v_1 < m + n$. Then $u = (v_1 + l - m - n, l - m - n, l - m - n - 1)$, $w = (v_1 + l - m - n, l - m - n - 1, l - m - n)$.

(ii) Suppose $m + n \le v_1 < l$ and $n > 0$. Then $u = (v_1 - m - n, l - n, l - m - 1)$ and $w = (v_1 - m - n, l - n - 1, l - m)$ if (s_1, s_2, s_3) is of the first type. Otherwise, $u = (v_1 - m - n, l - m, l - n - 1)$ and $w = (v_1 - m - n, l - m - 1, l - n)$

(iii) Suppose $m + n \le v_1 < l$, $n = 0$ and $l - m - n = 1$. Then $u = (0, 0, l - 1)$ and $w = (0, l - 1, 1)$ if (s_1, s_2, s_3) is of type 1. Otherwise, $u = (0, 1, l - 1)$ and $w = (0, l - 1, 0)$.

Proof.

(i) v also occupies $(v_1 + l - m - n, l - m - n, l - m - n)$ in $H_0^*(0)$. So u occupies $(v_1 + l - m - n, l - m - n, l - m - n - 1)$ and w occupies $(v_1 + l - m - n, l - m - n - 1, l - m - n)$. Since $v_1 < m + n$, the above two locations of u and w are in $H_0(0)$.

(ii) We first check $v \equiv v_1 s_1 \pmod N$ also occupies $(v_1 - m - n, l - n, l - m)$ if (s_1, s_2, s_3) is of type 1.

$(-m - n)(l^2 - mn) + (l - n)(m^2 + ln) + (l - m)(n^2 + lm) = 0$,

$(-m - n)(m^2 + ln) + (l - n)(n^2 + lm) + (l - m)(l^2 - mn)$

$= l^3 - m^3 - n^3 - 3lmn \equiv 0 \pmod N$,

$(-m - n)(n^2 + lm) + (l - n)(l^2 - mn) + (l - m)(m^2 + ln)$

$= l^3 - m^3 - n^3 - 3lmn \equiv 0 \pmod N$.

It is easily checked that $u = (v_1 - m - n, l - n, l - m - 1)$ and $w = (v_1 - m - n, l - n - 1, l - m)$ are in $H_0(0)$. The proof is similar if (s_1, s_2, s_3) is of type 2.

(iii) By the given conditions, we have $v_1 = m = l - 1$. Therefore $a = l^2$, $b = (l - 1)^2$, $c = l(l - 1)$. Note that

$$(l - 1)a \equiv lc \equiv lb + c \pmod N,$$

$$(l - 1)b \equiv la \equiv lc + a \pmod N,$$

$$(l - 1)c \equiv lb \equiv la + b \pmod N.$$

If (s_1, s_2, s_3) is of type 1, then v, which occupies cell $(l - 1, 0, 0)$ in $H_0(0)$, also occupies $(0, 0, l)$ and $(0, l, 1)$. Therefore u occupies $(0, 0, l - 1)$

and w occupies $(0, l-1, 1)$ in $H_0(0)$. The proof is similar if (s_1, s_2, s_3) is of type 2. □

We now prove that paths 2 and 3 are node-disjoint (their disjointness from path 1 is obvious). We consider three cases:

1. $0 \le v_1 < m + n$ or $m + n \le v_1 < l$ and $n > 0$. The locations of u and w in $H_0(0)$ are given in Lemma 3.2.2. Since $x_2 > 0$ for u and $x_3 > 0$ for w, a (2, 1, 3) dimension routing exists from 0 to u and a (3, 1, 2) from 0 to w. Node-disjointness is easily verified.
2. $m + n \le v_1 < l$, $n = 0$, $l - m - n > 1$. Since $l - m - n > 1$, u is 3-maximal and w 2-maximal in $H_0(0)$. Hence $x_2 > 0$ for w and $x_3 > 0$ for u. Use the (2, 1, 3) dimension routing from 0 to w, and the (3, 1, 2) dimension routing from 0 to u. Node-disjointness holds just as the previous two cases.
3. $m + n \le v_1 < l$, $n = 0$, $l - m - n = 1$. Suppose (s_1, s_2, s_3) is of type 1. By Lemma 3.3.2, $u = (0, 0, l-1)$ and $w = (0, l-1, 1)$ in $H_0(0)$. Since $x_3 > 0$ for u and $x_2 > 0$ for w, a (3, 1, 2) dimension routing (which degenerates into a dimension routing of (3)) exists from 0 to u, and a (2, 1, 3) dimension routing (which degenerates into a dimension routing of (2, 3)) exists from 0 to w. It is easily seen that the two paths are node-disjoint. Suppose (s_1, s_2, s_3) is of type 2. Then we switch he dimension routings between u and w.

Obliviousness is clear from the construction. □

We give an example that a minimum-3-routing does not exist. For $H_0(31; 9, 8, 14)$ and $v = 26$, the proposed routing yields length (3, 3, 7) while the routing: P_1': 0-9-17-26, P_2': 0-8-16-25-3-12-26, P_3': 0-14-23-1-10-18-26 yields length (3, 6, 6). Since $l_3 > l_3'$, (P_1, P_2, P_3) is not a minimum-3-routing. On the other hand, it is easily seen that if a minimum-3-routing exists, then (P_1, P_2, P_3), a weak minimum-3-routing, must be it.

Corollary 3.2.3 The connectivity of H_0 is 3.

Theorem 3.2.4 The k-diameter of H_0 is at most $D + k - 1$ for $k = 1, 2, 3$.

Proof. That the k-diameter for $k = 1, 2, 3$ does not exceed $D + k - 1$ is easily verified by our construction. It is also easily checked that the 1-diameter is indeed D since only dimension routing is used for path 1. For $k = 2$, the worst case is case (iii) in which a path may take $D + 1$ steps. We take $H_0(7; 2, 1, 4)$ (see Fig. 3.2.4) with $v = 2$ for example to show that $D + 1$ is realizable. Here path 2 is (0, 4, 5, 2) of length 3 = $D + 1$. For $k = 3$, the worst case is case (ii) in which a path may take $D + 2$ steps. We take $H_0(31; 6, 30, 26)$ (see Fig. 3.2.3) with $v = 4$ for example to show that $D + 2$ is realizable. Here path 3 is (0, 26, 21, 16, 11, 10, 9, 4) of length 7 = $D + 2$.

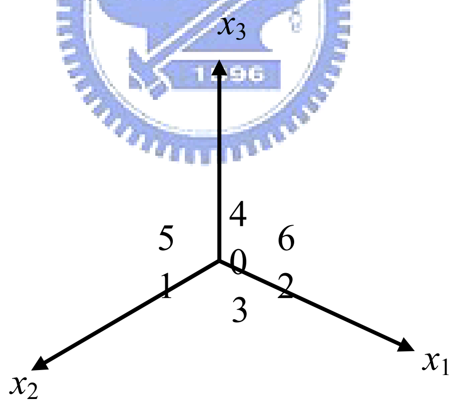


Fig. 3.2.4 $H_0(7; 2, 1, 4)$ with $v = 2$, where $u = 5$, $w = 1$.

Corollary 3.2.5 The 3-diameter of H_0 is at most $D + 2$.

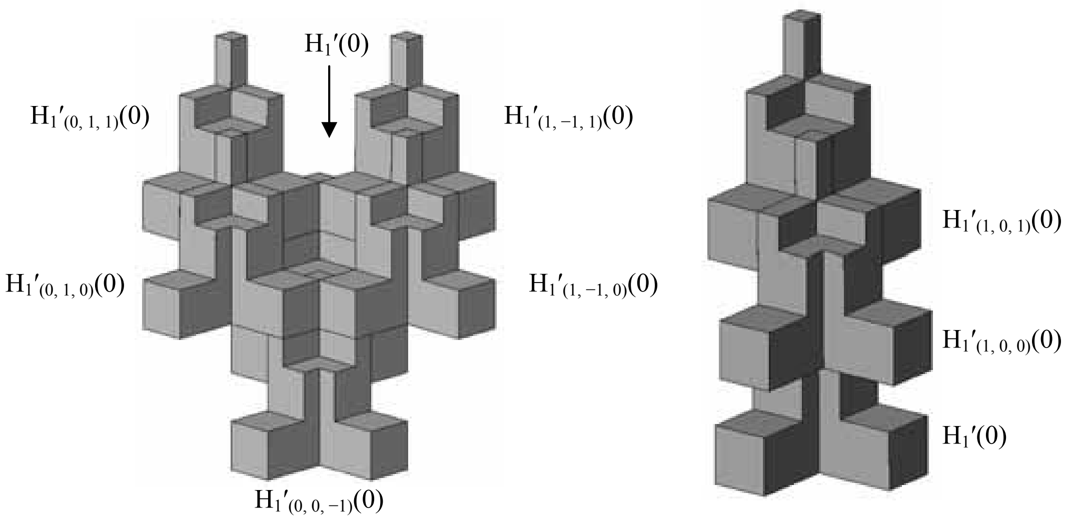
Corollary 3.2.6 The diameter of H_0 is at most $D + 2$ after two arbitrary failures (nodes or links).

3.3 Wide-Diameter of H_1'

In section 3.1, we have generalized H_1 and H_2 to H_1' and H_2' by allowing some line segments which have the same length to have different lengths. In this section, we also use oblivious weak minimum-3-routing to prove that H_1' is 3-connected by constructing 3 node-disjoint paths from any node i to any other node j. For 3-diameter of H_2', we will prove it in next section 3.4 by similar method.

For convenient, let $H_1'(0)$ ($H_2'(0)$) denote the MDD(0) of $H_1'(H_2')$. We define $H_1'_{(a,\,b,\,c)}(0)$ as the copy of $H_1'(0)$, which is obtained by adding the $a(n, n', 2h) + b(-m, n' + m', h) + c(-m, -m', h + h')$ vector, to each nodes of $H_1'(0)$, where $a, b, c \in \mathbb{Z}$.(See Fig. 3.3.1) Similarly, we define $H_2'_{(a,\,b,\,c)}(0)$ as the copy of $H_2'(0)$, which is obtained by adding the $a(2l + n', l' + m', m + 2n) + b(3l + n', -2l', m + n) + c(-2l - n', l', 2m + 3n)$ vector, to each nodes of H_2', where $a, b, c \in \mathbb{Z}$.(See Fig. 3.4.1) We call a node x occupying cell (x_1, x_2, x_3) in $H_1'(0)$ or $H_2'(0)$ *1-maximal* if cell $(x_1{+}1, x_2, x_3)$ is not in $H_1'(0)$ or $H_2'(0)$. Similarly we can define *2-maximal* and *3-maximal*.

Besides, we define that $t \equiv v - s_1 \pmod N$, $w \equiv v - s_2 \pmod N$, $u \equiv v - s_3 \pmod N$, $t' \equiv t - s_1 \pmod N$, $w' \equiv w - s_2 \pmod N$, and $u' \equiv u - s_3 \pmod N$.



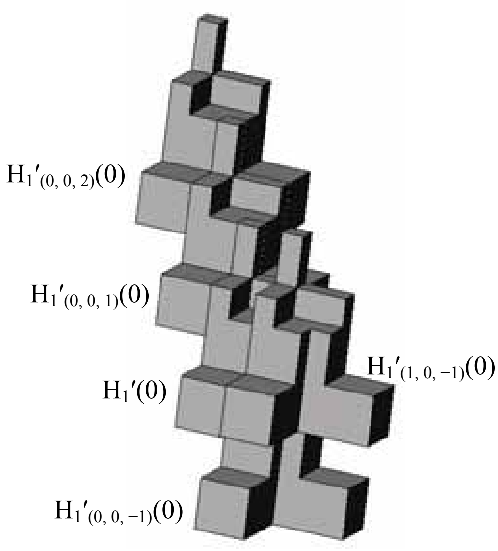


Fig. 3.3.1 $H_1'(0)$ and its copies.

Theorem 3.3.1 There exists an oblivious weak minimum-3-routing from node 0 to an arbitrary node v in H_1'. Suppose v occupies cell (v_1, v_2, v_3) in $H_1'(0)$. Let l_1, l_2, l_3 be the distances from 0 to t, w, u in $H_1'(0)$, respectively. The lengths of the three paths are

(i) $v_1 + v_2 + v_3$, $v_1 + v_2 + v_3$ and $v_1 + v_2 + v_3$, when $v_i > 0$ for $i = 1, 2, 3$.

(ii) $v_j + v_k$, $v_j + v_k$ and $l_i + 2$, when exactly one $v_i = 0$ for $i \in \{1, 2, 3\}$, where $j, k \in \{v_1, v_2, v_3\} / \{v_i\}$ and $j \neq k$.

(iii) v_k, $l_i + 1$ and $l_i + 1$, when $v_i = v_j = 0$ for $i, j \in \{1, 2, 3\}$, and $i \neq j$, where $k = \{v_1, v_2, v_3\} / \{v_i, v_j\}$.

Proof. We consider three cases:

(i) $v_i > 0$ for $i = 1, 2, 3$. We use dimension routing. The dimension order for path 1 is (1, 2, 3), for path 2 is (2, 3, 1) and for path 3 is (3, 1, 2). Then clearly, the three paths are node-disjoint and each has length $v_1 + v_2 + v_3$ which is the distance from 0 to v.

Since the lengths of these 3 paths are equal to the distance from 0 to v, it's obvious that the paths we construct constitute a minimum-3-routing.

(ii) Exactly one $v_i = 0$. We consider three cases:

a. $v_1 = 0$. We use dimension routing in the $x_1 = 0$ plane (where v lies) with orders (2, 3) and (3, 2), respectively, to obtain two node-disjoint paths to v. The third path will be routed through node t as a penultimate node. Suppose t is not in the $x_1 = 0$ plane. Then path 3 is obtained by a dimension routing from node 0 to t starting with s_1-steps. Since path 3 uses only nodes not in the $x_1 = 0$ plane in $H_1'(0)$, it is node-disjoint from paths 1 and 2.

Besides, we know that t is 1-maximal in $H_1'(0)$ or v would lie in a plane $x_1 = k > 0$ in $H_1'(0)$, contradicting our assumption that $v_1 = 0$.

Suppose t is in the $x_1 = 0$ plane. From the fact that t is 1-maximal, necessarily, $n = 1$ or $m = 1$. For $n = 1$, we only need to consider the condition that t is located in the following two regions R_1 and R_2:

1. R_1: $x_1 = 0$, $0 \le x_2 < n'$, $2h + h' \le x_3 < 3h + h'$.

   It occurs when $v_1 = 0$, $2m' \le v_2 < 2m' + n'$, $0 \le v_3 < h$ for v also occupies cell $(n, v_2 - 2m', v_3 + 2h + h')$ in $H_1'_{(1, -1, 1)}(0)$. Thus we have that t occupies cell $(0, v_2 - 2m', v_3 + 2h + h')$ in $H_1'(0)$. Since t also occupies cell $(m, v_2 - m', v_3 + h)$ in $H_1'_{(0, 0, -1)}(0)$. Therefore, t' occupies cell $(m - 1, v_2 - m', v_3 + h)$ in $H_1'(0)$, and v occupies cell $(m + 1, v_2 - m', v_3 + h)$ in $H_1'_{(1, -1, 0)}(0)$. Path 3 starts with an s_1-step and enter cell (1, 0, 0), followed by a dimension routing to t' in $H_1'(0)$, and then add an s_1-step to t in $H_1'_{(0, 0, -1)}(0)$. Path 3 is completed by an s_1-step to v in $H_1'_{(1, -1, 0)}(0)$.

2. R_1: $x_1 = 0$, $n' \le x_2 < n' + m'$, $2h \le x_3 < 2h + h'$.

   It occurs when $v_1 = 0$, $0 \le v_2 < m'$, $0 \le v_3 < h'$ for v also occupies cell $(n, v_2 + n', v_3 + 2h)$ in $H_1'_{(1, 0, 0)}(0)$. Thus we have that t occupies cell $(0, v_2 + n', v_3 + 2h)$ in $H_1'(0)$. Since t also occupies cell $(m, v_2 + n' + m', v_3 + h - h')$ in $H_1'_{(0, 0, -1)}(0)$. Therefore, t' occupies cell $(m - 1, v_2 + n' + m', v_3 + h - h')$ in $H_1'(0)$, and v occupies cell $(m + 1, v_2 + n' + m', v_3 + h - h')$ in $H_1'_{(1, 0, -1)}(0)$. Path 3 starts with an s_1-step and enter cell (1, 0, 0), followed by a dimension routing to t' in $H_1'(0)$, and then add an s_1-step to t in $H_1'_{(0, 0, -1)}(0)$. Path 3 is completed by an s_1-step to v in $H_1'_{(1, 0, -1)}(0)$.

Hence, path 3 is node-disjoint from paths 1 and 2, and it has length at most D + 2.

For $m = 1$, we only need to consider the condition that t is located in the following four regions R_1, R_2, R_3 and R_4:

1. R_1: $x_1 = 0$, $m' \le x_2 < m' + n'$, $h + h' \le x_3 < 2h$. (if $h' < h$)

   It occurs when $v_1 = 0$, $0 \le v_2 < n'$, $2h + 2h' \le v_3 < 3h + 2h'$ for v also occupies cell $(m, v_2 + m', v_3 - h - h')$ in $H_1'_{(0, 0, -1)}(0)$. Thus we have that t occupies cell $(0, v_2 + m', v_3 - h - h')$ in $H_1'(0)$. Since t also occupies cell $(m, v_2 + 2m', v_3 - 2h - 2h')$ in $H_1'_{(0, 0, -1)}(0)$. Therefore v occupies cell $(m + 1, v_2 + 2m', v_3 - 2h - 2h')$ in $H_1'_{(0, 0, -2)}(0)$. Path 3 starts with an s_1-step and enter cell (1, 0, 0), followed by a dimension routing to t in $H_1'_{(0, 0, -1)}(0)$. Path 3 is completed by an s_1-step to v in $H_1'_{(0, 0, -2)}(0)$.
2. R_2: $x_1 = 0$, $m' \le x_2 < m' + n'$, $h \le x_3 < h + h'$.

   It occurs when $v_1 = 0$, $0 \le v_2 < n'$, $2h + h' \le v_3 < 2h + 2h'$, because of the same reason for R_1. Since t also occupies cell $(n + m, v_2, v_3 - h')$ in $H_1'_{(1, -1, 0)}(0)$. Therefore v occupies cell $(n + m + 1, v_2, v_3 - h')$ in $H_1'_{(1, -1, -1)}(0)$. Path 3 starts with an s_1-step and enter cell (1, 0, 0), followed by a dimension routing to t in $H_1'_{(1, -1, 0)}(0)$. Path 3 is completed by an s_1-step to v in $H_1'_{(1, -1, -1)}(0)$.
3. R_3: $x_1 = 0$, $m' \le x_2 < 2m'$, $0 \le x_3 < h$.

   It is the same as the proof for R_2, except that it occurs when $v_1 = 0$, $n' \le v_2 < m'$ (if $n' < m'$), $h + h' \le v_3 < 2h + h'$.
4. R_4: $x_1 = 0$, $2m' + n' \le x_2 < 2m' + n'$, $0 \le x_3 < h$.

   It occurs when $v_1 = 0$, $m' \le v_2 < n' + m'$, $h + h' \le v_3 < 2h + h'$, because of the same reason for R_1. Since t also occupies cell $(n, v_2 - m', v_3 + h)$ in $H_1'_{(1, -1, 1)}(0)$. Therefore v occupies cell $(n + 1, v_2 - m', v_3 + h)$ in $H_1'_{(1, -1, 0)}(0)$. Path 3 starts with an s_1-step and enter cell (1, 0, 0), followed by a dimension routing to t in $H_1'_{(1, -1, 1)}(0)$. Path 3 is completed by an s_1-step to v in $H_1'_{(1, -1, 0)}(0)$.

Hence, path 3 is node-disjoint from paths 1 and 2, and it has length at most D + 2.

Since the lengths of paths 1 and 2 are equal to the distance from 0 to v, these two paths are shortest. Further, all shortest paths must start and end either with an s_2-step or an s_3-step (any combination allowed). Therefore a third disjoint path must start and end with an s_1-step, i.e., the second node of the path is s_1 and the penultimate node is t. Since our proposed third path uses dimension routing from s_1 to t, it is shortest among the set of third disjoint paths given that the first and second paths are shortest. Hence the proposed routing is a weak minimum-3-routing.

Since the proofs of the two cases, $v_2 = 0$ and $v_3 = 0$, are analogous to $v_1 = 0$, we only consider the conditions different from $v_1 = 0$.

b. $v_2 = 0$. Suppose w is in the $x_2 = 0$ plane. From the fact that w is 2-maximal, necessarily, $n' = 1$ or $m' = 1$. For $n' = 1$, we only need to consider the condition that w is located in the following two regions R_1 and R_2:

1. R_1: $0 \le x_1 < n, x_2 = 0, 2h + h' \le x_3 < 3h + h'$.

   It occurs when $2m \le v_1 < 2m + n$, $v_2 = 0$, $0 \le v_3 < h$ for v also occupies cell $(v_1 - 2m, n', v_3 + 2h + h')$ in $H_1'_{(0, 1, 1)}(0)$. Thus we have that w occupies cell $(v_1 - 2m, 0, v_3 + 2h + h')$ in $H_1'(0)$. Since w also occupies cell $(v_1 - m, m', v_3 + h)$ in $H_1'_{(0, 0, -1)}(0)$. Therefore, w' occupies cell $(v_1 - m, m' - 1, v_3 + h)$ in $H_1'(0)$, and v occupies cell $(v_1 - m, m' + 1, v_3 + h)$ in $H_1'_{(0, 1,}$

$_{0)}(0)$.

2. R_2: $n \le x_1 < n + m$, $x_2 = 0$, $2h \le x_3 < 2h + h'$.

   It occurs when $0 \le v_1 < m$, $v_2 = 0$, $0 \le v_3 < h'$ for v also occupies cell $(v_1 + n, n', v_3 + 2h)$ in $H_1'_{(1, 0, 0)}(0)$. Thus we have that w occupies cell $(v_1 + n, 0, v_3 + 2h)$ in $H_1'(0)$. Since w also occupies cell $(v_1 + n + m, m', v_3 + h - h')$ in $H_1'_{(0, 0, -1)}(0)$. Therefore, w' occupies cell $(v_1 + n + m, m' - 1, v_3 + h - h')$ in $H_1'(0)$, and v occupies cell $(v_1 + n + m, m' + 1, v_3 + h - h')$ in $H_1'_{(1, 0, -1)}(0)$.

For $m' = 1$, we only need to consider the condition that w is located in the following four regions R_1, R_2, R_3 and R_4:

1. R_1: $m \le x_1 < m + n$, $x_2 = 0$, $h + h' \le x_3 < 2h$. (if $h' < h$)

   It occurs when $0 \le v_1 < n$, $v_2 = 0$, $2h + 2h' \le v_3 < 3h + 2h'$ for v also occupies cell $(v_1 + m, m', v_3 - h - h')$ in $H_1'_{(0, 0, -1)}(0)$. Thus we have that w occupies cell $(v_1 + m, 0, v_3 - h - h')$ in $H_1'(0)$. Since w also occupies cell $(v_1 + 2m, m', v_3 - 2h - 2h')$ in $H_1'_{(0, 0, -1)}(0)$. Therefore v occupies cell $(v_1 + 2m, m' + 1, v_3 - 2h - 2h')$ in $H_1'_{(0, 0, -2)}(0)$.

2. R_2: $m \le x_1 < m + n$, $x_2 = 0$, $h \le x_3 < h + h'$.

   It occurs when $0 \le v_1 < n$, $v_2 = 0$, $2h + h' \le v_3 < 2h + 2h'$, because of the same reason for R_1. Since w also occupies cell $(v_1, n' + m', v_3 - h')$ in $H_1'_{(0, 1, 0)}(0)$. Therefore v occupies cell $(v_1, n' + m' + 1, v_3 - h')$ in $H_1'_{(0, 1, -1)}(0)$.

3. R_3: $m \le x_1 < 2m$, $x_2 = 0$, $0 \le x_3 < h$.

   It is the same as the proof for R_2, except that it occurs when $n \le v_1 < m$ (if $n < m$), $v_2 = 0$, $h + h' \le v_3 < 2h + h'$.

4. R_4: $2m + n \le x_1 < 2m + n$, $x_2 = 0$, $0 \le x_3 < h$.

   It occurs when $m \le v_1 < n + m$, $v_2 = 0$, $h + h' \le v_3 < 2h + h'$, because of the same reason for R_1. Since w also occupies cell $(v_1 - m, n', v_3 + h)$ in $H_1'_{(0, 1, 1)}(0)$. Therefore v occupies cell $(v_1 - m, n' + 1, v_3 + h)$ in $H_1'_{(0, 1,}$

$_{0)}(0)$.

c. $v_3 = 0$. Suppose u is in the $x_3 = 0$ plane. From the fact that u is 3-maximal, necessarily, $h = 1$. Hence we only need to consider the condition that u is located in the following four regions R_1, R_2, R_3 and R_4:

1. R_1: $0 \le x_1 < m$, $n' + m' \le x_2 < 2m'$, $x_3 = 0$.

   It occurs when $m \le v_1 < 2m$, $0 \le v_2 < m' - n'$, $v_3 = 0$ for v also occupies cell $(v_1 - m, v_2 + n' + m', h)$ in $H_1'_{(0, 1, 0)}(0)$. Thus we have that u occupies cell $(v_1 - m, v_2 + n' + m', 0)$ in $H_1'(0)$. Since u also occupies cell $(v_1 + n, v_2 + n', h)$ in $H_1'_{(1, -1, 0)}(0)$. Therefore, u' occupies cell $(v_1 + n, v_2 + n', h - 1)$ in $H_1'(0)$, and v occupies cell $(v_1 + n, v_2 + n', h + 1)$ in $H_1'_{(1, 0, 0)}(0)$.

2. R_2: $0 \le x_1 < m$, $2m' \le x_2 < n' + 2m'$, $x_3 = 0$.

   It occurs when $m \le v_1 < 2m$, $m' - n' \le v_2 < m'$, $v_3 = 0$. That's the same reason for R_1. Since u also occupies cell $(v_1 + n, v_2 - 2m', 2h + h')$ in $H_1'_{(1, -1, 1)}(0)$. Therefore, u' occupies cell $(v_1 + n, v_2 - 2m', 2h + h' - 1)$ in $H_1'(0)$, and v occupies cell $(v_1 + n, v_2 - 2m', 2h + h' + 1)$ in $H_1'_{(1, 0, 1)}(0)$.

3. R_3: $n + m \le x_1 < 2m$, $0 \le x_2 < m'$, $x_3 = 0$.

   It occurs when $0 \le v_1 < m - n$, $m' \le v_2 < 2m'$, $v_3 = 0$ for v also occupies cell $(v_1 + n + m, v_2 - m', h)$ in $H_1'_{(1, -1, 0)}(0)$. Thus we have that u occupies cell $(v_1 + n + m, v_2 - m', 0)$ in $H_1'(0)$. Since u also occupies cell $(v_1 + n, v_2 + n', h)$ in $H_1'_{(0, 1, 0)}(0)$. Therefore, u' occupies cell $(v_1 + n, v_2 + n', h - 1)$ in $H_1'(0)$, and v occupies cell $(v_1 + n, v_2 + n', h + 1)$ in $H_1'_{(1, 0, 0)}(0)$.

4. R_4: $2m \le x_1 < n + 2m$, $0 \le x_2 < m'$, $x_3 = 0$.

   It occurs when $m - n \le v_1 < m$, $m' \le v_2 < 2m'$, $v_3 = 0$. That's the same reason for R_3. Since u also occupies cell $(v_1 - 2m, v_2 + n', 2h + h')$ in $H_1'_{(0, 1, 1)}(0)$. Therefore, u' occupies cell $(v_1 - 2m, v_2 + n', 2h + h' - 1)$ in $H_1'(0)$, and v occupies cell $(v_1 - 2m, v_2 + n', 2h + h' + 1)$ in $H_1'_{(1, 0, 1)}(0)$. Path 3 starts with an s_3-step and enter cell (0, 0, 1), followed by a dimension routing to u' in $H_1'(0)$, and then add an s_3-step to u in $H_1'_{(0,1,1)}(0)$.

(iii) Exactly two $v_i = 0$. We consider three cases:

a. $v_2 = v_3 = 0$. Path 1 is the unique shortest path from node 0 to v along the x_1-axis. We will show that in $H_1'(0)$ one of u and w has $x_2 > 0$ and the other $x_3 > 0$. Then we let path 2 go from 0 to s_2, followed by a dimension routing to the node in $\{u, w\}$ with $x_2 > 0$ (in fact, the dimension routing starts with dimension 2, hence is also a dimension routing from 0). Similarly, path 3 goes from 0 to s_3 followed by a dimension routing (starting from dimension 3) to the other node in $\{u, w\}$. Then a weak minimum-3-routing must have one path starting from s_2 and one from s_3. But our proposed paths constitute a shortest pair from s_2 and s_3 since they use dimension routing. This proves weak minimum-3-routing.

To prove the existence of the desirable u and w, we first prove a lemma which is located u and w in $H_1'(0)$.

Lemma 3.3.2 Let $v = (v_1, 0, 0)$.

(i) Suppose $0 \le v_1 < m$. Then $u = (v_1 + n, n', 2h - 1)$ and $w = (v_1 + n, n' - 1, 2h)$.

(ii) Suppose $m \le v_1 < 2m$. Then $u = (v_1 - m, n' + m', h - 1)$ and $w = (v_1 - m, n' + m' - 1, h)$.

(iii) Suppose $2m \le v_1 < 2m + n$. Then $u = (v_1 - 2m, n', 2h + h' - 1)$ and $w = (v_1 - 2m, n' - 1, 2h + h')$.

Proof.

(i) Since v also occupies $(v_1 + n, n', 2h)$ in $H_1'_{(1,0,0)}(0)$, u occupies $(v_1 + n, n', 2h - 1)$ and w occupies $(v_1 + n, n' - 1, 2h)$. Since $v_1 < m$, the above two locations of u and w are in $H_1'(0)$.

(ii) Since v also occupies $(v_1 - m, n' + m', h)$ in $H_1'_{(0,1,0)}(0)$, u occupies $(v_1 - m, n' + m', h - 1)$ and w occupies $(v_1 - m, n' + m' - 1, h)$. Since $m \le v_1 < 2m$, the above two locations of u and w are in $H_1'(0)$.

(iii) Since v also occupies $(v_1 - 2m, n', 2h + h')$ in $H_1'_{(0, 1, 1)}(0)$, u occupies $(v_1 - 2m, n', 2h + h' - 1)$ and w occupies $(v_1 - 2m, n' - 1, 2h + h')$. Since $2m \le v_1 < 2m + n$, the above two locations of u and w are in $H_1'(0)$. □

By lemma 3.3.2, we get that $x_2 > 0$ for u and $x_3 > 0$ for w. Thus a (2, 1, 3) dimension routing exists from 0 to u and a (3, 1, 2) from 0 to w. Hence, paths 2 and 3 are node-disjoint (their disjointness from path 1 is obvious), and they have lengths at most D + 1.

Obliviousness is clear from the construction.

Since the proofs of two cases, $v_1 = v_3 = 0$ and $v_1 = v_2 = 0$, are analogous to $v_2 = v_3 = 0$, we only consider the conditions different from $v_2 = v_3 = 0$.

b. $v_1 = v_3 = 0$. We will show that in $H_1'(0)$ one of t and u has $x_1 > 0$ and the other $x_3 > 0$.

Lemma 3.3.3 Let $v = (0, v_2, 0)$.

(i) Suppose $0 \le v_2 < m'$. Then $t = (n - 1, v_2 + n', 2h)$ and $u = (n, v_2 + n', 2h - 1)$.

(ii) Suppose $m' \le v_2 < 2m'$. Then $t = (n + m - 1, v_2 - m', h)$ and $u = (n + m, v_2 - m', h - 1)$.

(iii) Suppose $2m' \le v_2 < 2m' + n'$. Then $t = (n - 1, v_2 - 2m', 2h + h')$ and $u = (n, v_2 - 2m', 2h + h' - 1)$.

Proof.

(i) Since v also occupies $(n, v_2 + n', 2h)$ in $H_1'_{(1, 0, 0)}(0)$, t occupies $(n - 1, v_2 + n', 2h)$ and u occupies $(n, v_2 + n', 2h - 1)$.

(ii) Since v also occupies $(n + m, v_2 - m', h)$ in $H_1'_{(1, -1, 0)}(0)$, t occupies $(n + m - 1, v_2 - m', h)$ and u occupies $(n + m, v_2 - m', h - 1)$.

(iii) Since v also occupies $(n, v_2 - 2m', 2h + h')$ in $H_1'_{(1, -1, 1)}(0)$, t occupies $(n - 1, v_2 - 2m', 2h + h')$ and u occupies $(n, v_2 - 2m', 2h + h' - 1)$. □

c. $v_1 = v_2 = 0$. We will show that in $H_1'(0)$ one of t and w has $x_1 > 0$ and the other $x_2 > 0$.

Lemma 3.3.4 Let $v = (0, 0, v_3)$.

(i) Suppose $0 \le v_3 < h'$. Then $t = (n - 1, n', v_3 + 2h)$ and $w = (n, n' - 1, v_3 + 2h)$.

(ii) Suppose $h' \le v_3 < h + h'$. Then $t = (n + 2m - 1, 0, v_3 - h')$ and $w = (0, n' + 2m' - 1, v_3 - h')$.

(iii) Suppose $h + h' \le v_3 < 3h + h'$. Then $t = (m - 1, m', v_3 - h - h')$ and $w = (m, m' - 1, v_3 - h - h')$.

Proof.

(i) Since v also occupies $(n, n', v_3 + 2h)$ in $H_1'_{(1, 0, 0)}(0)$, t occupies $(n - 1, n', v_3 + 2h)$ and w occupies $(n, n' - 1, v_3 + 2h)$.

(ii) Since v also occupies $(n + 2m, 0, v_3 - h')$ in $H_1'_{(1, -1, -1)}(0)$, t occupies $(n + 2m - 1, 0, v_3 - h')$. Since v also occupies $(0, n' + 2m', v_3 - h')$ in $H_1'_{(0, 1, -1)}(0)$, thus w occupies $(0, n' + 2m' - 1, v_3 - h')$.

(iii) Since v also occupies $(m, m', v_2 - h - h')$ in $H_1'_{(0, 0, -1)}(0)$, t occupies $(m - 1, m', v_3 - h - h')$ and w occupies $(m, m' - 1, v_3 - h - h')$. □

□

We give an example that a minimum-3-routing does not exist in H_1'. For $H_1'(161; 117, 2, 7)$ and $v = 26$ with coordinates (0, 6, 2), the proposed routing yields lengths (8, 8, 12), the proposed routing yields lengths (8, 8, 12) while the routing: P_1': 0-2-4-6-8-10-17-24-26, P_2': 0-7-14-21-28-35-42-49-56-63-70-26, P_3': 0-117-124-131-138-145-152-159-5-12-19-26 yields length (8, 11, 11). Since $l_3 > l_3'$, (P_1, P_2, P_3) is not a minimum-3-routing.

Corollary 3.3.5 The connectivity of H_1' is 3.

Theorem 3.3.6 The *k*-diameter of H_1' is at most $D + k - 1$ for $k = 1, 2, 3$.

Proof. It's the same as Theorem 3.2.4. □

Corollary 3.3.7 The 3-diameter of H_1' is at most $D + 2$.

Corollary 3.3.8 The diameter of H_1' is at most $D + 2$ after two arbitrary failures (nodes or links).

3.4 Wide-Diameter of H_2'

Similar to the previous section, we use oblivious weak minimum-3-routing to prove that H_2' is 3-connected by constructing 3 node-disjoint paths from any node i to any other node j in this section.

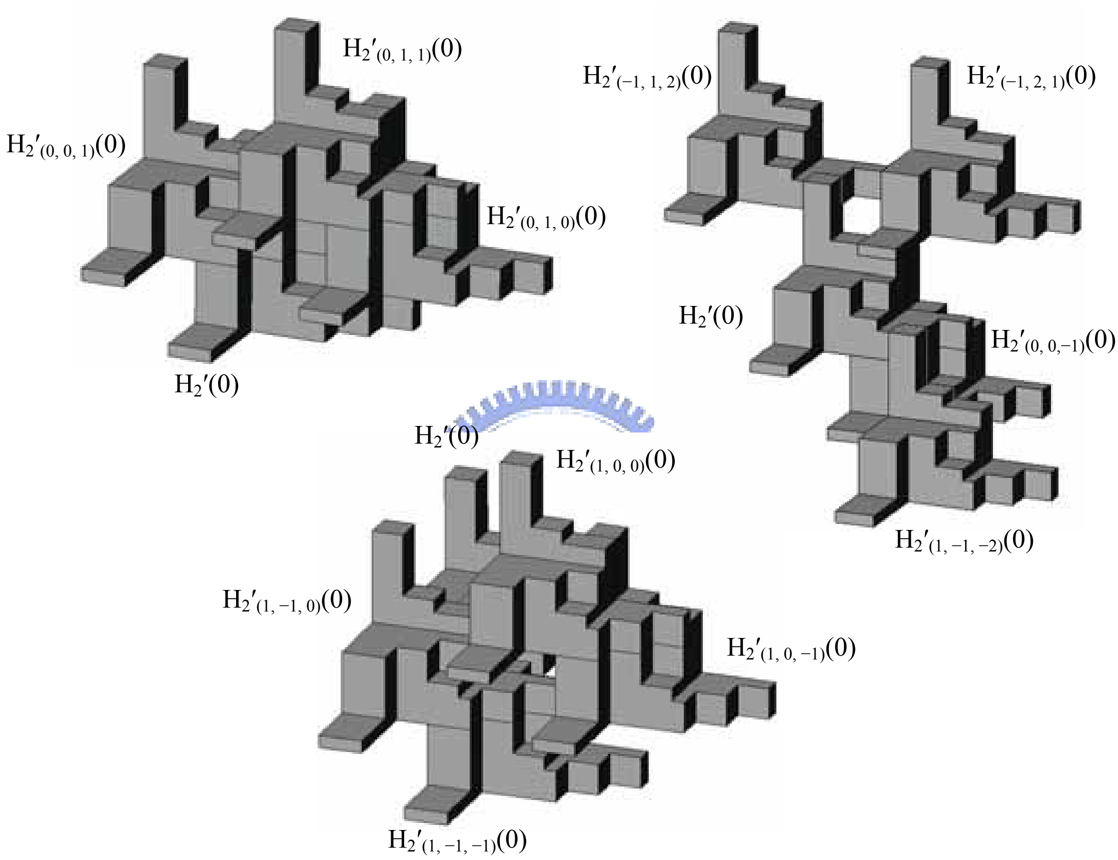


Fig. 3.4.1 $H_2'(0)$ and its copies.

Theorem 3.4.1 There exists an oblivious weak minimum-3-routing from node 0 to an arbitrary node v in H_2'. Suppose v occupies cell (v_1, v_2, v_3) in $H_2'(0)$. Let l_1, l_2, l_3 be the distances from 0 to t, w, u in $H_2'(0)$, respectively. The lengths of the three paths are

(i) $v_1 + v_2 + v_3$, $v_1 + v_2 + v_3$ and $v_1 + v_2 + v_3$, when $v_i > 0$ for $i = 1, 2, 3$.

(ii) $v_j + v_k$, $v_j + v_k$ and $l_i + 2$, when exactly one $v_i = 0$ for $i \in \{1, 2, 3\}$, where j, $k \in$

$\{v_1, v_2, v_3\}$ / $\{v_i\}$ and $j \neq k$.

(iii) v_k, $l_i + 1$ and $l_i + 1$, when $v_i = v_j = 0$ for $i, j \in \{1, 2, 3\}$, and $i \neq j$, where $k = \{v_1, v_2, v_3\}$ / $\{v_i, v_j\}$.

Proof. Since this proof is similar to Theorem 3.3.1, we only consider the following two conditions different from Theorem 3.3.1.

(ii) Exactly one $v_i = 0$. We consider three cases:

a. $v_1 = 0$. Suppose t is in the $x_1 = 0$ plane. From the fact that t is 1-maximal, necessarily, $l = 1$. Hence we only need to consider the condition that t is located in the following two regions R_1 and R_2:

1. R_1: $x_1 = 0, 0 \leq x_2 < l', 4n + 3m \leq x_3 < 5n + 4m$.

   It occurs when $v_1 = 0, l \leq v_2 < 2l, 0 \leq v_3 < n + m$ for v also occupies cell $(l, v_2 - l', v_3 + 4n + 3m)$ in $H_2'_{(0, 1, 1)}(0)$. Thus we have that t occupies cell $(0, v_2 - l', v_3 + 4n + 3m)$ in $H_2'(0)$. Since t also occupies cell $(l + n', v_2 + l' + m', v_3 + 2n + m)$ in $H_2'_{(1, -1, -1)}(0)$. Therefore, t' occupies cell $(n', v_2 + l' + m', v_3 + 2n + m)$ in $H_2'(0)$, and v occupies cell $(l + n' + 1, v_2 + l' + m', v_3 + 2n + m)$ in $H_2'_{(1, 0, 0)}(0)$.

2. R_1: $x_1 = 0, 0 \leq x_2 < l', 5n + 4m \leq x_3 < 6n + 5m$.

   It occurs when $v_1 = 0, l \leq v_2 < 2l, n + m \leq v_3 < 2n + 2m$ for the same reason of the above case. Since t also occupies cell $(3l + 2n', v_2 + m', v_3 - n - m)$ in $H_2'_{(1, -1, -2)}(0)$. Therefore, t' occupies cell $(2l + 2n', v_2 + m', v_3 - n - m)$ in $H_2'(0)$, and v occupies cell $(3l + 2n' + 1, v_2 + m', v_3 - n - m)$ in $H_2'_{(1, 0, -1)}(0)$.

b. $v_2 = 0$. Suppose w is in the $x_2 = 0$ plane. From the fact that w is 2-maximal, necessarily, $l' = 1$ or $m' = 1$. For $l' = 1$, we only need to consider the condition that w is located in the following five regions R_1, R_2, R_3, R_4 and R_5:

1. R_1: $0 \leq x_1 < l, x_2 = 0, 5n + 4m \leq x_3 < 6n + 5m$.

   It occurs when $2l + n' \leq v_1 < 3l + n', v_2 = 0, 2n + 2m \leq v_3 < 3n + 3m$

for v also occupies cell $(v_1 - 2l - n', l', v_3 + 3n + 2m)$ in $H_2'_{(0,0,1)}(0)$. Thus we have that w occupies cell $(v_1 - 2l - n', 0, v_3 + 3n + 2m)$ in $H_2'(0)$. Since w also occupies cell $(v_1 + l + n', l' + m', v_3 - 2n - 2m)$ in $H_2'_{(1,-1,-2)}(0)$. Therefore, w' occupies cell $(v_1 + l + n', m', v_3 - 2n - 2m)$ in $H_2'(0)$, and v occupies cell $(v_1 + l + n', l' + m' + 1, v_3 - 2n - 2m)$ in $H_2'_{(1,-1,-1)}(0)$.

2. R_2: $0 \le x_1 < l, x_2 = 0, 3n + 2m \le x_3 < 5n + 4m$.

   It occurs when $2l + n' \le v_1 < 3l + n', v_2 = 0, 0 \le v_3 < 2n + 2m$ for the same reason for R_1. Since w also occupies cell $(v_1 - l, 2l' + m', v_3 + n)$ in $H_2'_{(1,-1,-1)}(0)$. Therefore, w' occupies cell $(v_1 - l, l' + m', v_3 + n)$ in $H_2'(0)$, and v occupies cell $(v_1 - l, 2l' + m' + 1, v_3 + n)$ in $H_2'_{(1,-1,0)}(0)$.

3. R_3: $l \le x_1 < 2l, x_2 = 0, 3n + 2m \le x_3 < 4n + 3m$.

   It is the same as the proof for R_2, except that it occurs when $3l + n' \le v_1 < 4l + n', v_2 = 0, 0 \le v_3 < n + m$.

4. R_4: $2l \le x_1 < 2l + n', x_2 = 0, 3n + 2m \le x_3 < 3n + 3m$

   It is the same as the proof for R_2, except that it occurs when $4l + n' \le v_1 < 4l + 2n', v_2 = 0, 0 \le v_3 < m$.

5. R_5: $2l + n' \le x_1 < 3l + n', x_2 = 0, 3n + 2m \le x_3 < 3n + 3m$.

   It occurs when $4l + 2n' \le v_1 < 5l + 2n', v_2 = 0, 0 \le v_3 < m$ for the same reason for R_1. Since w also occupies cell $(v_1 - 4l - 2n', l', v_3 + 6n + 4m)$ in $H_2'_{(0,0,1)}(0)$. Therefore, w' occupies cell $(v_1 - 4l - 2n', 0, v_3 + 6n + 4m)$ in $H_2'(0)$, and v occupies cell $(v_1 - 4l - 2n', l' + 1, v_3 + 6n + 4m)$ in $H_2'_{(0,0,2)}(0)$.

For $m' = 1$, we only need to consider the condition that w is located in the following two regions R_6 and R_7:

1. R_6: $4l + 2n' \le x_1 < 5l + 2n', x_2 = 0, 0 \le v_3 < m$.

   It occurs when $0 \le v_1 < l, v_2 = 0, n + m \le x_3 < n + 2m$ for v also occupies cell $(v_1 + 2l + n', l' + m', v_3 + 2n + m)$ in $H_2'_{(1,0,0)}(0)$. Thus we have that w occupies cell $(v_1 + 2l + n', l', v_3 + 2n + m)$ in $H_2'(0)$. Since w

also occupies cell $(v_1, 2l', v_3 + 5n + 3m)$ in $H_2'_{(0, 0, 1)}(0)$. Therefore, w' occupies cell $(v_1, 2l' - 1, v_3 + 5n + 3m)$ in $H_2'(0)$, and v occupies cell $(v_1, 2l' + 1, v_3 + 5n + 3m)$ in $H_2'_{(1, 0, 0)}(0)$.

2. R_7: $4l + 2n' \leq x_1 < 5l + 2n'$, $x_2 = 0$, $m \leq x_3 < n + m$.

   It occurs when $0 \leq v_1 < l$, $v_2 = 0$, $n + 2m \leq x_3 < 2n + 2m$ for the same reason for R_6. Since w also occupies cell $(v_1 - 2l, 6l' + m', v_3 + 2n)$ in $H_2'_{(1, -2, 0)}(0)$. Therefore, w' occupies cell $(v_1 - 2l, 6l', v_3 + 2n)$ in $H_2'(0)$, and v occupies cell $(v_1 - 2l, 6l' + m' + 1, v_3 + 2n)$ in $H_2'_{(2, -2, -1)}(0)$.

c. $v_3 = 0$. Suppose u is in the $x_3 = 0$ plane. From the fact that u is 3-maximal, necessarily, $n = 1$. Hence we only need to consider the condition that u is located in the following three regions R_1 and R_2:

1. R_1: $0 \leq x_1 < n'$, $3l + m' \leq x_2 < 4l + m'$, $x_3 = 0$.

   It occurs when $l \leq v_1 < l + n'$, $0 \leq v_2 < l'$, $v_3 = 0$ for v also occupies cell $(v_1 - l, v_2 + 3l' + m', n)$ in $H_2'_{(1, -1, 0)}(0)$. Thus we have that u occupies cell $(v_1 - l, v_2 + 3l' + m', 0)$ in $H_2'(0)$. Since u also occupies cell $(v_1 + 2l + n', v_2 + l' + m', m + n)$ in $H_2'_{(0, 1, 0)}(0)$. Therefore, u' occupies cell $(v_1 + 2l + n', v_2 + l' + m', m)$ in $H_2'(0)$, and v occupies cell $(v_1 + 2l + n', v_2 + l' + m', m + n + 1)$ in $H_2'_{(1, 0, 0)}(0)$.

2. R_2: $n' \leq x_1 < l + n'$, $3l + m' \leq x_2 < 4l + m'$, $x_3 = 0$.

   It occurs when $l + n' \leq v_1 < 2l + n'$, $0 \leq v_2 < l'$, $v_3 = 0$ for the same reason for R_1. Since u also occupies cell $(v_1 - l - n', v_2, 6n + 5m)$ in $H_2'_{(-1, 2, 2)}(0)$. Therefore, u' occupies cell $(v_1 - l - n', v_2, 5n + 5m)$ in $H_2'(0)$, and v occupies cell $(v_1 - l - n', v_2, 6n + 5m + 1)$ in $H_2'_{(0, 1, 2)}(0)$.

3. R_3: $0 \leq x_1 < l + n'$, $4l + m' \leq x_2 < 5l + m'$, $x_3 = 0$.

   It occurs when $l \leq v_1 < 2l + n'$, $l' \leq v_2 < 2l'$, $v_3 = 0$ for the same reason for R_1. Since u also occupies cell $(v_1 + l, v_2 - l', 3n + 3m)$ in $H_2'_{(-1, 2, 1)}(0)$. Therefore, u' occupies cell $(v_1 + l, v_2 - l', 2n)$ in $H_2'(0)$, and v occupies cell $(v_1 + l, v_2 - l', 3n + 3m + 1)$ in $H_2'_{(0, 1, 1)}(0)$.

(iii) Exactly two $v_i = 0$. We consider three cases:

a. $v_2 = v_3 = 0$. We will show that in $H_2'(0)$ one of u and w has $x_2 > 0$ and the other $x_3 > 0$. To prove the existence of the desirable u and w, we first prove a lemma which is located u and w in $H_2'(0)$.

Lemma 3.4.2 Let $v = (v_1, 0, 0)$.

(i) Suppose $0 \le v_1 < l$. Then $w = (v_1 + 2l + n', m', m + 2n)$ and $u = (v_1 + 2l + n', l' + m', m + 2n - 1)$.

(ii) Suppose $l \le v_1 < 2l + n'$. Then $w = (v_1 - l, 2l' + m', n)$ and $u = (v_1 - l, 3l' + m', n - 1)$.

(iii) Suppose $2l + n' \le v_1 < 5l + 2n'$. Then $w = (v_1 - 2l - n', 0, 2m + 3n)$ and $u = (v_1 - 2l - n', l', 2m + 3n - 1)$.

Proof.

(i) Since v also occupies $(v_1 + 2l + n', l' + m', m + 2n)$ in $H_2'_{(1, 0, 0)}(0)$, w occupies $(v_1 + 2l + n', m', m + 2n)$ and u occupies $(v_1 + 2l + n', l' + m', m + 2n - 1)$.

(ii) Since v also occupies $(v_1 - l, 3l' + m', n)$ in $H_2'_{(0, -1, 1)}(0)$, w occupies $(v_1 - l, 2l' + m', n)$ and u occupies $(v_1 - l, 3l' + m', n - 1)$.

(iii) Since v also occupies $(v_1 - 2l - n', l', 2m + 3n)$ in $H_2'_{(0, 0, 1)}(0)$, w occupies $(v_1 - 2l - n', 0, 2m + 3n)$ and u occupies$(v_1 - 2l - n', l', 2m + 3n - 1)$. $\square$

b. $v_1 = v_3 = 0$. We will show that in $H_2'(0)$ one of t and u has $x_1 > 0$ and the other $x_3 > 0$.

Lemma 3.4.3 Let $v = (0, v_2, 0)$.

(i) Suppose $0 \le v_2 < l'$. Then $t = (2l + n' - 1, v_2 + l' + m', m + 2n)$ and $u = (2l + n', v_2 + l' + m', m + 2n - 1)$.

(ii) Suppose $l' \le v_2 < 2l'$. Then $t = (l - 1, v_2 - l', 4n + 3m)$ and $u = (l, v_2 - l', 4n + 3m - 1)$.

(iii) Suppose $2l' \le v_2 < 4l' + m'$. Then $t = (3l + n' - 1, v_2 - 2l', n + m)$ and $u = (3l + n', v_2 - 2l', n + m - 1)$.

(iv) Suppose $4l' + m' \le v_2 < 5l' + m'$. Then $t = (2l - 1, v_2 - 4l' - m', 3n + 3m)$ and $u = (2l, v_2 - 4l' - m', 3n + 3m - 1)$.

Proof.

(i) Since v also occupies $(2l + n', v_2 + l' + m', m + 2n)$ in $H_2'_{(1, 0, 0)}(0)$, t occupies $(2l + n' - 1, v_2 + l' + m', m + 2n)$ and u occupies $(2l + n', v_2 + l' + m', m + 2n - 1)$.

(ii) Since v also occupies $(l, v_2 - l', 4n + 3m)$ in $H_2'_{(0, 1, 1)}(0)$, t occupies $(l - 1, v_2 - l', 4n + 3m)$ and u occupies $(l, v_2 - l', 4n + 3m - 1)$.

(iii) Since v also occupies $(3l + n', v_2 - 2l', n + m)$ in $H_2'_{(0, 1, 0)}(0)$, t occupies $(3l + n' - 1, v_2 - 2l', n + m)$ and u occupies $(3l + n', v_2 - 2l', n + m - 1)$.

(iv) Since v also occupies $(2l, v_2 - 4l' - m', 3n + 3m)$ in $H_2'_{(-1, 2, 1)}(0)$, t occupies $(2l - 1, v_2 - 4l' - m', 3n + 3m)$ and u occupies $(2l, v_2 - 4l' - m', 3n + 3m - 1)$. □

c. $v_1 = v_2 = 0$. We will show that in $H_1'(0)$ one of t and w has $x_1 > 0$ and the other $x_2 > 0$.

Lemma 3.4.4 Let $v = (0, 0, v_3)$.

(i) Suppose $0 \le v_3 < n + m$. Then $t = (2l + n' - 1, l' + m', v_3 + m + 2n)$ and $w = (2l + n', l' + m' - 1, v_3 + m + 2n)$.

(ii) Suppose $n + m \le v_3 < 2n + 2m$. Then $t = (4l + 2n' - 1, m', v_3 - m - n)$ and $w = (4l + 2n', m' - 1, v_3 - m - n)$.

(iii) Suppose $2n + 2m \le v_3 < 5n + 4m$. Then $t = (l + n' - 1, 2l' + m', v_3 - 2m - 2n)$ and $w = (l + n', 2l' + m' - 1, v_3 - 2m - 2n)$.

(iv) Suppose $5n + 4m \le v_3 < 6n + 5m$. Then $t = (3l + 2n' - 1, l' + m', v_3 - 4m - 5n)$

and $w = (3l + 2n', l' + m' - 1, v_3 - 4m - 5n)$.

Proof.

(i) Since v also occupies $(2l + n', l' + m', v_3 + m + 2n)$ in $H_2'_{(1,0,0)}(0)$, t occupies $(2l + n' - 1, l' + m', v_3 + m + 2n)$ and w occupies $(2l + n', l' + m' - 1, v_3 + m + 2n)$.

(ii) Since v also occupies $(4l + 2n', m', v_3 - m - n)$ in $H_1'_{(1,0,-1)}(0)$, t occupies $(4l + 2n' - 1, m', v_3 - m - n)$. Since v also occupies $(0, n' + 2m', v_3 - h')$ in $H_2'_{(0,1,-1)}(0)$, thus w occupies $(4l + 2n', m' - 1, v_3 - m - n)$.

(iii) Since v also occupies $(l + n', 2l' + m', v_3 - 2m - 2n)$ in $H_2'_{(1,-1,-1)}(0)$, t occupies $(l + n' - 1, 2l' + m', v_3 - 2m - 2n)$ and w occupies $(l + n', 2l' + m' - 1, v_3 - 2m - 2n)$.

(iv) Since v also occupies $(3l + 2n', l' + m', v_3 - 4m - 5n)$ in $H_2'_{(1,-1,-2)}(0)$, t occupies $(3l + 2n' - 1, l' + m', v_3 - 4m - 5n)$ and w occupies $(3l + 2n', l' + m' - 1, v_3 - 4m - 5n)$. □

□

We give an example that a minimum-3-routing does not exist in H_2', For $H_2'(273; 255, 262, 41)$ and $v = 226$ with coordinates $(2, 1, 0)$, the proposed routing yields lengths $(3, 3, 14)$ while the routing: P_1': 0-255-244-226, P_2': 0-262-251-240-229-218-207-196-185-226, P_3': 0-41-30-19-8-270-259-248-237-226 yields lengths $(3, 9, 9)$. Since $l_3 > l_3'$, (P_1, P_2, P_3) is not a minimum-3-routing.

Corollary 3.4.5 The connectivity of H_2' is 3.

Theorem 3.4.6 The k-diameter of H_2' is at most $D + k - 1$ for $k = 1, 2, 3$.

Proof. It's the same as Theorem 3.2.4. □

Corollary 3.4.7 The 3-diameter of H_2' is at most $D + 2$.

Corollary 3.4.8 The diameter of H_2' is at most $D + 2$ after two arbitrary failures (nodes or links).

Chapter 4 WSNB on $\text{Log}_2(N, m, p)$ Networks

4.1 Architecture

For computer networks, delays more than polylog time are generally unacceptable. Therefore centralized routing algorithms which usually require $O(N \log N)$ time are out. Instead, a bunch of $\log_2 N$-stage networks with self-routing property have been invented; here, *self-routing*, first proposed by Lawrie [30] for the Omega network, means that a request can be routed by only knowing its input and output, and nothing about other requests. These networks are usually recognized as the banyan-type by the following features.

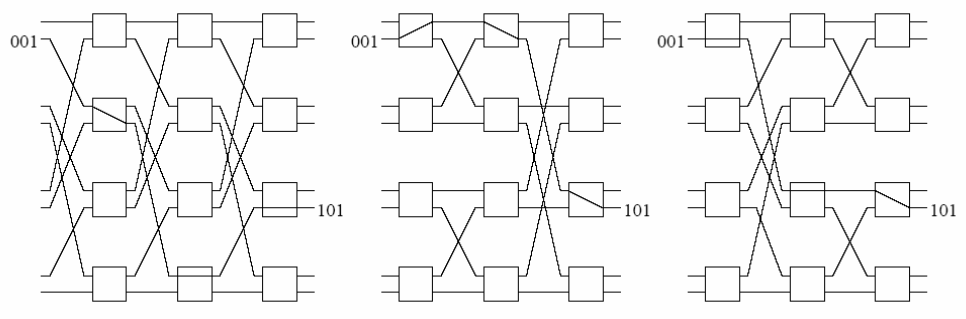
(i) The network is an n-stage binary network ($n = \log_2 N$).

(ii) Each input has a unique path to each output.

Dais and Jump [16] introduced the "buddy" notation: Let v and v' be two crossbars in stage i and let V_v and $V_{v'}$ be two sets of crossbars in stage j that v and v' can reach, respectively. Then the network is a buddy network if for any i and $j = i + 1$, either $V_v = V_{v'}$ or $V_v \cap V_{v'} = \phi$.

Agrawal [1] called a buddy network a strict buddy network if the buddy condition also holds for $j = i + 2$. Chen et al. [12] further generalize the strict buddy network to the universal buddy network by allowing j to be arbitrary.

Some well known self-routing networks which have the buddy property, are shown in Fig. 4.1.1.



(a) Omega (OM) (b) Banyan (BY) (c) Baseline (BL)

Fig. 4.1.1 Some self-routing networks.

The above class of binary networks with N inputs and M outputs can be extended to d-nary by replacing (i) with (i') $N = M = d^n$. The network consists of n stages of crossbars of size $d \times d$.

An (n + 1)-stage buddy network was first proposed by Siegal-Smith [41] for increasing the connection power and for fault tolerance. Shyy and Lea [40] considered adding m extra stages to BY^{-1} and specified that the extra m stages should be identical to the mirror image of the first m stages. Represent a m-extra-stage buddy network by B(n, m) or B(N, m). The specified way of addition has the advantage that $BY^{-1}(n, m)$ can be sequentially decomposed m times, $1 \le j \le m$, namely the subnetwork of $BY^{-1}(n, m)$ from stage $j + 1$ to stage $n + m - j$ decomposed into 2^j $BY^{-1}(n - j, m - j)$ such that each input (output) switch of the $BY^{-1}(n, m)$ has a unique path to each $BY^{-1}(n - j, m - j)$ (see Fig. 4.1.2 in which the external terminals are not drawn). Denote this way of adding extra stages by F^{-1}. Hwang [26]observed that there are three other natural ways of addition.

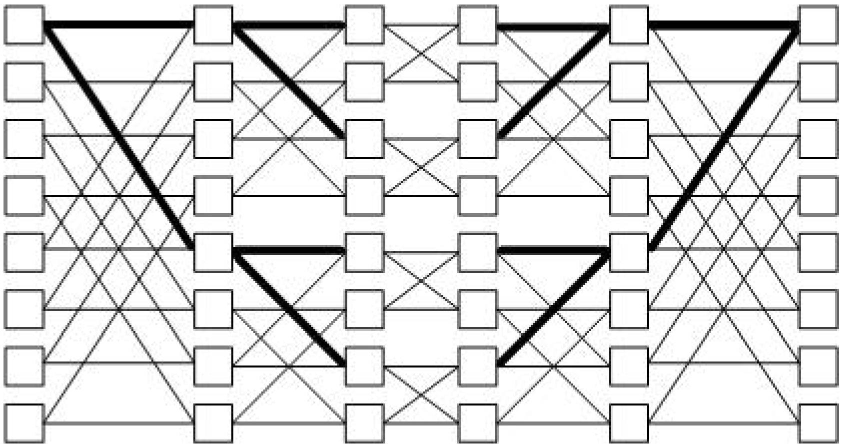


Fig. 4.1.2 Decomposition of $BY^{-1}(4, 2)$.

(i) F: The extra m stages are identical to the first m stages.

(ii) L: The extra m stages are identical to the last m stages.

(iii) L^{-1}: The extra m stages are identical to the mirror image of the last m stages.

The various ways of addition result in different networks with different connection capabilities in general. Extra-stage/Omega networks are known as shuffle exchange (SE) networks. Hwang-Liaw-Yeh determined the equivalence classes among the m-extra-stage networks SE(m), $SE^{-1}(m)$, BY(m), $BY^{-1}(m)$, BL(m), $BL^{-1}(m)$ for all m and under each of F, F^{-1}, L, L^{-1}.

A network is *strictly nonblocking* (SNB) if the current request can always be connected regardless of how previous connections were routed. While $BY^{-1}(n, m)$ itself is not an SNB network, Lea and Shyy [32] first proposed the $\mathrm{Log}_2(N, m, p)$ network with $N = 2^n$ inputs (outputs), which consists of a vertical stacking of p copies of $BY^{-1}(n, m)$, $0 \le m \le n-1$, sandwiched between and connected to an input stage and an output stage, each with N $1 \times p$ (or $p \times 1$) crossbars. As shown in Fig. 4.1.3, there are three copies of $BY^{-1}(3, 1)$ sandwiched between the input and output stages. Later, Hwang [24] extended the $\mathrm{Log}_2(N, m, p)$ network to $\mathrm{Log}_d(N, m, p)$ network by replacing the 2×2 crossbars with $d \times d$ crossbars.

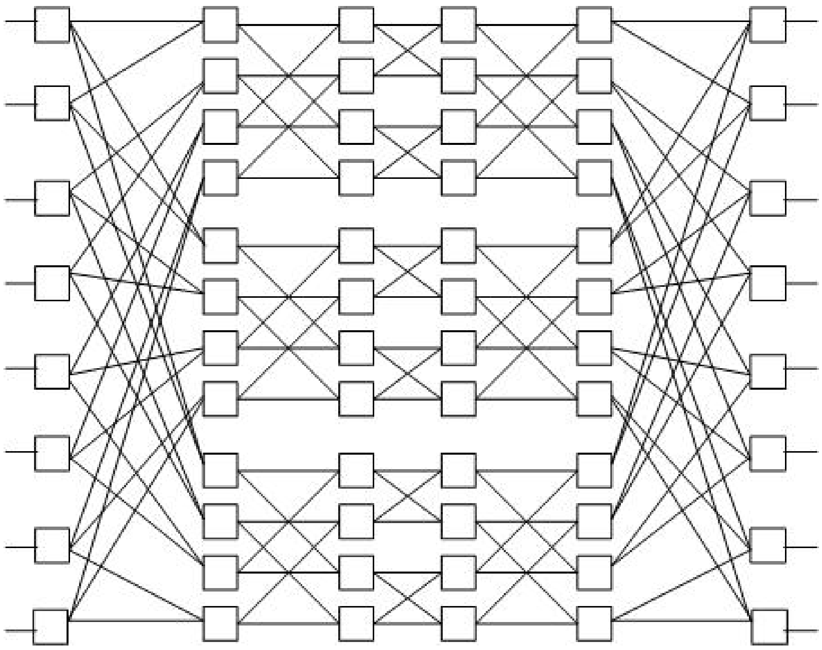


Fig. 4.1.3 $Log_2(8, 1, 3)$.

4.2 Blockingness

Traffic can be classified as *point-to-point*, like 2-party phone calls, or *broadcast*, which is one to many. If there is a restriction on the maximum number of receivers per request, then broadcast is called *multicast*, or *f*-cast, if that number is specified to be *f*. Traffic can be further divided into two types according to whether additional receivers can be added after a multicast request is already connected. We will use *open-end broadcasting* (which allows additions) and *closed-end broadcasting* (which does not allow) to differentiate the two types.

Traditionally, there are different levels of nonblockingness: strictly, wide-sense and rearrangeable. In this thesis, we only consider the wide-sense condition. A network is *wide-sense nonblocking* (WSNB) if the connection of the current request is assured only when all connections are routed according to a given algorithm.

Before providing the classical results of $\text{Log}_2(N, m, p)$ networks, we first study the concept of channel graph. The *channel graph* CG(i, o) between an input i and an output o is the union of all paths connecting them (see Fig. 4.2.1). In $\text{BY}^{-1}(n, m)$, all channel graphs are isomorphic with the following double-tree form (two binary trees with their 2^m leaves linked by paths in a one-to-one fashion). The channel graph of a multicast call is simply the union of its point-to-point channel graphs.

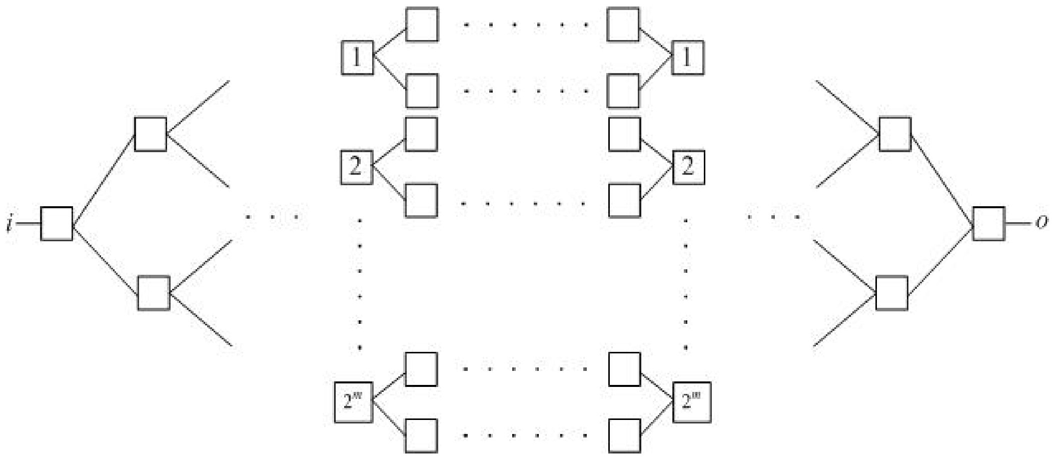


Fig. 4.2.1 A channel graph of $BY^{-1}(n, m)$.

Note that whether a request can be connected depends only on the state of the channel graph: a request is blocked if and only if every path in its channel graph contains an occupied link.

Throughout this thesis, a link connecting stage i and stage $(i + 1)$ is called a *stage-i link*. Note that the inputs (outputs) are the 0-th (n-th) link stage. We use *shell i* to denote the i-th link stage and the $(n - i)$-th link stage for $0 \leq i \leq \lceil (n - 1)/2 \rceil$. An *intersecting connection* is one which contains a link in the channel graph of the request. An intersecting connection is an i-intersecting connection if it first (last) intersects the channel graph in a stage-i link when counted from the input (output) side.

4.3 Classical Multicast WSNB Results

Much less is known for WSNB; perhaps because it is not easy to come up with intelligent routing algorithms which can make a difference. Suppose that $\mathrm{Log}_d(N, m, p)$ is constructed by vertically stacking p copies of $\mathrm{BY}_d^{-1}(n,m)$, denoted by M_1, M_2, …, M_p. We show five evident routing algorithms in the following.

1. Save-the-unused (STU). Do not route through an empty M_j unless there is no choice, where $j = 1, 2, \ldots, p$.
2. Packing (P). Route through anyone of the busiest M_j's, where $j = 1, 2, \ldots, p$.
3. Minimum index (MI). Route through the M_j with the smallest index if possible, where $j = 1, 2, \ldots, p$.
4. Cyclic dynamic (CD). If M_j is used in routing the last request, try M_{j+1}, M_{j+2}, …, in that cyclic order.
5. Cyclic static (CS). Same as CD except starting from M_j, where $j = 1, 2, \ldots, p$.

Note that STU includes P.

Chang et al. [6] showed that the number of copy networks required for WSNB under each of the above five routing strategies in the $\mathrm{Log}_d(N, 0, p)$ network is same as required for SNB, thus dashing any hope of saving hardware while retaining the nonblocking property.

Tscha and Lee [44] proposed a multicast WSNB algorithm, denoted by *window algorithm*, for $\mathrm{Log}_2(N, 0, p)$ network. Define $\delta = 2^{\lfloor n/2 \rfloor}$. They partitioned the N outputs of $\mathrm{BY}^{-1}(n, m)$ into N/δ windows, each containing the δ outputs reachable from the same crossbar at stage $n + m - \lfloor n/2 \rfloor + 1$. In other words, if the outputs are labeled by binary n sequences, then a θ-window consists of those outputs, which have the same $n - \theta$ most significant bits. Although an output can be reached by $2^{\theta-1}$ crossbars at stage $n + m - \theta + 1$, each such crossbar reaches the same window due to the well-known "buddy" property of banyan type networks. Fig. 4.3.1 shows that the outputs {0,1,8,9}, reachable from the first crossbar at stage five, form a 2-window of $BY^{-1}(4, 2)$.

By window algorithm, an *f*-cast request will be split to several *f*-cast subrequests each consisting of outputs in a given θ-window. Two rules are observed in this θ-window routing :

1. Each subrequest uses one path up to $n - \theta$ stage (for a *n*-stage network).
2. The subrequests from the same request are treated as independent requests, i.e., they cannot share any link.

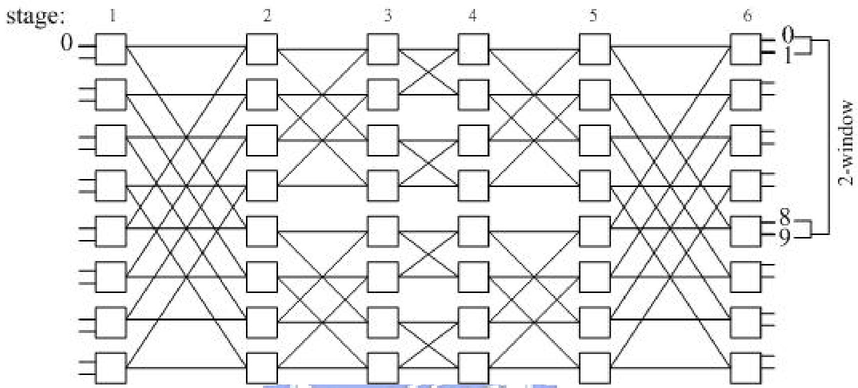


Fig. 4.3.1 A 2-window of $BY^{-1}(4, 2)$.

Tscha and Lee [44] proved

Theorem 4.3.1 $Log_2(N, 0, p)$ is multicast WSNB under the window algorithm if $p \geq \lfloor n/2 \rfloor 2^{\lfloor (n-1)/2 \rfloor} + 1$.

At first, they stated Theorem 4.3.1 as an SNB result. However, Kabacinski and Danilewicz [29] pointed out that their proof using "windows" to split a multicast call implies a routing algorithm, hence, their result is WSNB instead of strictly nonblocking. Note that Theorem 4.3.1 was proved by setting $\theta = \lfloor n/2 \rfloor$.

Kabacinski and Danilewicz [29] extended the fixed window-size algorithm in [40] to variable window size and proved

Theorem 4.3.2 $\text{Log}_2(N, 0, p)$ is multicast WSNB under the θ-neighborhood routing if

$$p \geq \begin{cases} \theta \cdot 2^{n-\theta-1} + \left\lceil 2^{n-2\theta-1} \right\rceil, & \text{for } 1 \leq \theta \leq \lfloor n/2 \rfloor, \\ 2^{\theta} + (n-\theta-2)2^{n-\theta-1} - 2^{2\theta-n-1} + 1, & \text{for } \lfloor n/2 \rfloor \leq \theta \leq n. \end{cases}$$

Besides, Tscha and Lee [44] stated in conclusion that whether their approach could be extended to $\text{Log}_2(N, m, p)$ was unclear. Danilewicz and Kabacinski [13, 14] made such an attempt but encountered some difficulties. They treated the worst case as each request is point-to-point. Though in most cases the minimum p is obtained for window-size equal to $\lceil (n + m)/2 \rceil$, there are cases when this number is obtained for window-size less than $\lceil (n + m)/2 \rceil$. At the end, they had no general formula for WSNB switching networks for window-size larger than $\lceil (n + m)/2 \rceil$. In section 4.4, we will give such an extension for the variable window-size algorithm by adopting a channel graph blockage analysis first used by Shyy and Lea [40] on a single-cast network. The $\text{Log}_2(N, m, p)$ network is much more difficult to analyze because of multipaths in the channel graph and each link having a different impact on blockage. We also determine the optimal window size for given m, and then compare the performance among different m in section 4.5.

4.4 WSNB $\text{Log}_2(N, m, p)$

In this section, we [28] further extended Theorem 4.3.2 to $\text{Log}_2(N, m, p)$. Following Tscha and Lee [44], we split a multicast request into w multicast subrequests if the involved outputs spread into w windows, while each subrequest must be routed through the same copy of $\text{BY}^{-1}(n, m)$. When we are discussing a multicast request with respect to a given θ-window, we refer to it as the *designated* θ*-window*. Further, a θ'*-window* is *designated* if it contains the designated θ-window. As Tscha and Lee [44] dealt only with $\text{BY}^{-1}(n)$, the connection from an input to an output is unique, and whether two connections intersect is determined. Therefore, an intersection graph among the connections within a designated $\lfloor n/2 \rfloor$-window can be defined, and its maximum degree plus one becomes the number of copies of $\text{BY}^{-1}(n)$ sufficient for nonblocking. Besides, we assume $\theta < n$ to avoid trivial cases.

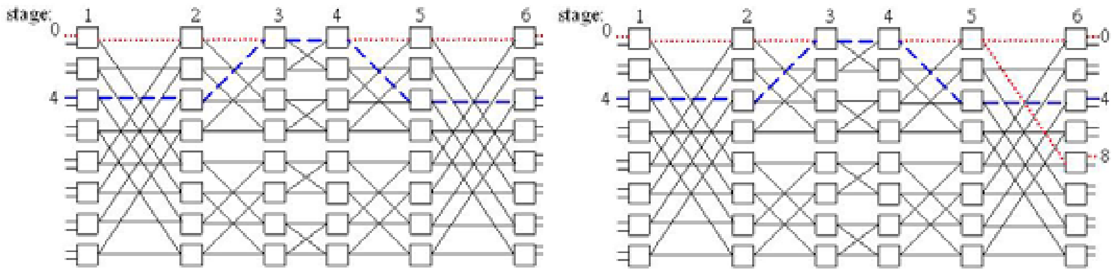
For $\text{BY}^{-1}(n, m)$, the analysis is much more complicated as the connection between an input and an output is not unique. First of all, we have to be more specific about the window algorithm. We propose the delayed-splitting θ-window algorithm, which prescribes that a multicast connection to outputs in the same θ-window cannot be split before stage $(n + m - \theta + 1)$. Note that further delay is not always possible, since stage $n + m - \theta + 1$ is the last stage where all outputs in the same window have common reachable crossbars. Also note that such an algorithm fixes only the relative routing of two outputs in the same θ'-window, $\theta' \leq \theta$, but not the absolute routing to an output. Thus, whether two connections intersect is uncertain and the notion of an intersection graph used by Tscha and Lee [47] is not applicable. Instead, we adopt the method of channel graph blockage analysis.

Recall that a link connecting stage i and stage $(i + 1)$ is called a *stage-i-link*. Consider a k-cast request in a θ-window. An *intersecting connection* is one which contains a link in the channel graph of the request. We can count an intersecting connection either from its input end or its output end. An intersecting connection is an i-intersecting connection if it first (last) intersects the channel graph in a stage-i link when counted from the input (output) side.

We count all i-intersecting connections, $n + m - \theta \leq i \leq n + m - 1$, from the output side. Note that the outputs of these connections must all be in the designated θ-window. Thus, there are, at most, $2^{\theta} - k$ of such connections. Further, they have different impacts in blocking the paths in the channel graph, depending on i. For example, for $m \geq 2$, an $(n + m - 1)$-intersecting connection blocks a proportion of 1/2, since the channel graph has only two stage-$(n + m - 1)$ links, while an $(n + m - 2)$-intersecting connection blocks a proportion of 1/4, since the channel graph has four stage-$(n + m - 2)$ links.

On the other hand, we will count all i-intersecting connections, $1 \leq i \leq n + m - \theta - 1$, from the input side. Again, an i-intersecting connection has a greater (or equality permitted) blocking impact than an $(i + 1)$-intersecting call for $i \leq \lfloor (n + m)/2 \rfloor$. We will show that we never need to count from the input side over the stage $\lfloor (n + m)/2 \rfloor$. Therefore, we adopt the method used in [29] to count from small i to large i to maximize the blocking impact.

In section 4.1, we have known that $BY^{-1}(n)$ and many other networks have buddy property. Note that in a buddy network, the set of inputs which can generate an intersecting connection to a multicast request is independent of the size of that request. To see this, consider a 2-cast call from input i to two outputs o and o'. Then an input $i' \neq i$ can generate a k-intersecting connection (at a crossbar u') to the path from i to o' if and only if it can generate a k-intersecting connection (at a crossbar u) to the path from i to o, since the buddy property assures that if i'can reach u', it can reach u. Hence, increasing the size of the request does not increase the number of inputs which can generate intersecting connections, but the fact that these outputs are in the request makes them unavailable as outputs to generate intersecting connections (see Fig. 4.4.1, for example). Further, each intersecting connection blocks one copy, so it is the number of intersecting connections that counts. Obviously, a 1-cast request maximizes that number.



(a) $BY^{-1}(4, 2)$ (b) $BY^{-1}(4, 2)$

Fig. 4.4.1 Input 4 generates a 3-intersecting connection (4, 4) to (a) a 1-cast request (0, 0) and (b) a 2-cast request (0, {0, 8}).

For $BY^{-1}(n, m)$, although the same analysis on the number of intersecting connections applies, the i-intersecting connections block different fractions of a copy, depending on i. Since more outputs in a multicast request induce more i-intersecting calls for larger i, the worst case is not necessarily a 1-cast request.

We consider two cases.

A. $0 \le m \le 1$

The number of stage-i links, $1 \le i \le n + m - 1$, in the channel graph is constant, one for $m = 0$, and two for $m = 1$. Therefore, each intersecting connection has the same impact, regardless of which stage it intersects. The worst case occurs when there is a maximum number of intersecting connections, i.e., $2^{\theta} - 1$ from the designated window, which cause a blocking of $(2^{\theta} - 1)/2^{m}$ copies.

B. $2 \le m$

Let R denote the part of the new request which goes to a designated θ-window. Suppose R is k-cast and a 1-window contains r outputs in R. Then it can block, at most

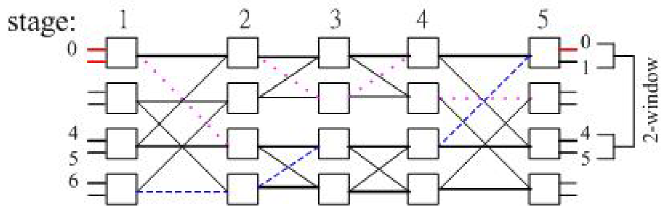
$$2 \times \frac{1}{4} = \frac{1}{2} \text{ if } r = 0$$

$$\text{(only for the 1 - window which is in the designated 2 - window),}$$

$$1 \times \frac{1}{2} = \frac{1}{2}, \quad \text{if } r = 1$$

$$= 0, \quad \text{if } r = 2.$$

For instance, in Fig. 4.4.2, the first output crossbar corresponds to the case r = 1, and the third output crossbar corresponds to the case $r = 0$.



$BY^{-1}(3, 2)$

Fig. 4.4.2 Assume $\theta = 2$ and (0, 0) is the request. $r = 1$ in the first output crossbar and connection (6, 1) blocks 1/2 copy, while $r = 0$ in the third output crossbar and connections (4, 4) and (5, 5) each blocks 1/4 copy. Dotted lines indicate channel graph between the first input and the first output crossbar.

Therefore, a 1-window can block, at most, 1/2 copy of the channel graph. Consequently, a θ-window can block, at most, $2^{\theta-2}$ copies, which is achieved by having either $k = 2^{\theta-1}$ (each 1-window has $r = 1$) or $k = 2^{\theta-2}$ (half of the 1-window has $r = 1$ and half has $r = 0$).

To count i-intersecting connections for $1 \le i \le n + m - \theta - 1$ we consider two cases.

A. $\theta \le \lfloor n + m/2 \rfloor - 1$

The argument for this part is a straightforward extension of the argument in [29] for $m = 0$.

There are 2^{i-1} inputs which can generate an i-intersecting connection. Further, an i -intersecting connection can reach all windows for $i \le m$, and $2^{n-\theta-i+m}$ windows for $i \ge m$. In the worst-case scenario, an i-intersecting connection is a multicast connection going to one output in each window it can reach, except the designated window for $1 \le i \le \theta$. The reason for the exception is that all outputs in the designated window are already counted in the part concerning $n + m - \theta \le i \le n + m - 1$. Since an i-intersecting connection blocks 2^{-i} copies for $i \le m$ and 2^{-m} copies for $m \le i \le \lfloor (n + m)/2 \rfloor$, the total blocking of up to stage θ is

$$\begin{aligned}
&\sum_{i=1}^{\theta} 2^{i-1}(2^{n-\theta} - 1)2^{-i} \\
&= \sum_{i=1}^{\theta} 2^{n-\theta-1} - \sum_{i=1}^{\theta} 2^{-1} \\
&= \theta(2^{n-\theta-1} - \frac{1}{2}) \quad \text{for } \theta \le m
\end{aligned}$$

and

$$\begin{aligned}
&\sum_{i=1}^{m} 2^{i-1}(2^{n-\theta} - 1)2^{-i} + \sum_{i=m+1}^{\theta} 2^{i-1}(2^{n-\theta-i+m} - 1)2^{-m} \\
&= \sum_{i=1}^{m} 2^{n-\theta-1} - \sum_{i=1}^{m} 2^{-1} + \sum_{i=m+1}^{\theta} 2^{n-\theta-1} - \sum_{i=m+1}^{\theta} 2^{i-m-1} \\
&= \theta 2^{n-\theta-1} - \frac{m}{2} - 2^{\theta-m} + 1 \quad \text{for } \theta \ge m.
\end{aligned}$$

Note that these i-intersecting connections, $1 \le i \le \theta$, use up a maximum of $\sum_{i=1}^{\theta} 2^{i-1} = 2^{\theta} - 1$ outputs in a window. Therefore, one $(\theta + 1)$-intersecting connection can still fit in if $\theta + 1 < n + m - \theta$, or $\theta \le \lfloor (n + m)/2 \rfloor - 1$, which is the case here. This $(\theta + 1)$-intersecting connection reaches windows for $\theta < m$, and $2^{n-2\theta-1+m} - 1$ windows for $\theta \ge m$, while each path to a window blocks 2^{-m} copy.

To summarize, the number of blockings from the input side is

$$\theta(2^{n-\theta-1}-\frac{1}{2})+2^{n-\theta-m}-2^{-m} \text{ for } \theta < m$$

$$\theta 2^{n-\theta-1}-\frac{m}{2}-2^{\theta-m}+1+2^{n-2\theta-1}-2^{-m} \text{ for } \theta \geq m.$$

B. $\theta \geq \lfloor n+m/2 \rfloor$

Then $\theta \geq m$. Note that i-intersecting connections for $n+m-\theta \leq i \leq n+m-1$ are counted from the output side. So the input side counts only up to stage $n+m-\theta-1$ (which is upper bounded by θ). Thus, the number of blockings from the input side is

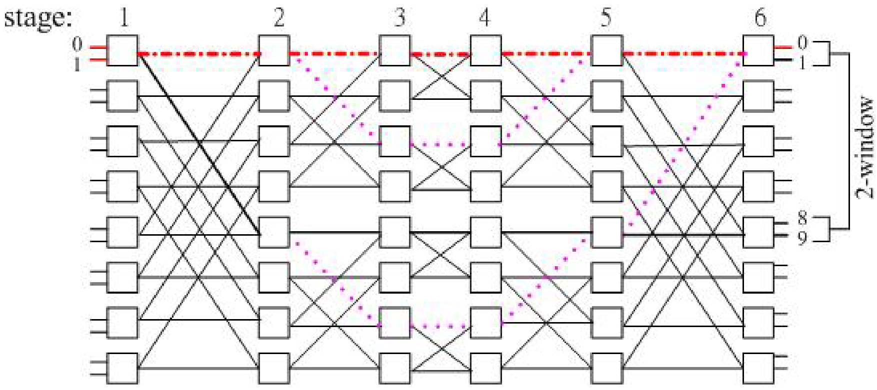
$$\sum_{i=1}^{m} 2^{i-1}(2^{n-\theta}-1)2^{-i}+\sum_{i=m+1}^{n+m-\theta-1} 2^{i-1}(2^{n-\theta-i+m}-1)2^{-m}$$

$$=(n+m-\theta-1)2^{n-\theta-1}-\frac{m}{2}-2^{n-\theta-1}+1$$

$$=(n+m-\theta-2)2^{n-\theta-1}-\frac{m}{2}+1.$$

Since each intersecting connection counted from the output side blocks in the worst-case scenario, i.e., $k = 2^{\theta-1}$ or $2^{\theta-2}$, at least 1/4 copy, there is no reason for the counting from input side to go over stage $n+m-\theta$, with one exception.

For $\theta \geq 2$, we can increase the blocking by allowing the unique 1-intersecting connection from the input side to also go to the designated window to reach an output blocking 1/4 copy (such an output exists when $k = 2^{\theta-2}$). Then this intersecting connection blocks 1/2 copy if counted from the input side, greater than its original value 1/4, as counted from the output side (see Fig. 4.4.3, for example). Note that no other such reversal of counting will bring any further increase, since the 1-intersecting connection is the only one which blocks more than 1/4 copy when counted from the input side. On the other hand, since all intersecting connections counted from the input side are before the middle stage, reversing them to the output side will only decrease their impact on blocking.



$BY^{-1}(4, 2)$

Fig. 4.4.3 Connection (1, 8) blocks 1/2 copy if counted from the input side, but only 1/4 copy from the output side. Dotted lines indicate channel graph between the first input and the first output crossbar.

Combining the above, we have

Theorem 4.4.1 $\text{Log}_2(N, m, p)$ is WSNB for multicast under the θ-window algorithm if and only if

$$p > \begin{cases} \theta \cdot 2^{n-\theta-1} + 2^{n-2\theta-1} - 1, & \text{for } m = 0,\ \theta \le \left\lfloor \frac{n}{2} \right\rfloor - 1 \\ \theta \cdot 2^{n-\theta-1} + 2^{n-2\theta-1} - \frac{1}{2}, & \text{for } m = 1,\ \theta \le \left\lfloor \frac{(n+1)}{2} \right\rfloor - 1 \\ 2^{\theta} + (n-\theta-2) \cdot 2^{n-\theta-1}, & \text{for } m = 0,\ \theta \ge \left\lfloor \frac{n}{2} \right\rfloor \\ 2^{\theta-1} + (n-\theta-1) \cdot 2^{n-\theta-1}, & \text{for } m = 1,\ \theta \ge \left\lfloor \frac{(n+1)}{2} \right\rfloor \\ 2^{\theta-2} + \theta \cdot 2^{n-\theta-1} - \frac{m}{2} - 2^{\theta-m} + 2^{n-2\theta-1} - 2^{-m} + \frac{5}{4}, & \text{for } 2 \le m \le \theta \le \left\lfloor \frac{(n+m)}{2} \right\rfloor - 1 \\ 2^{\theta-2} + \theta \cdot 2^{n-\theta-1} - \frac{\theta}{2} + 2^{n-\theta-m} - 2^{-m} + \frac{1}{4}(0 \text{ if } \theta = 1), & \text{for } m > \max\{\theta, 1\}, \theta \le \left\lfloor \frac{(n+m)}{2} \right\rfloor - 1 \\ 2^{\theta-2} + (n+m-\theta-2) \cdot 2^{n-\theta-1} - \frac{m}{2} + \frac{5}{4}, & \text{for } \theta \ge \left\lfloor \frac{(n+m)}{2} \right\rfloor \ge m \ge 2 \end{cases}$$

Results for $m = 0$ correspond to the results in [29]; results for $m = 1, 2$ correspond to the results in [13] and [14].

Note that $\text{Log}_2(N, n-1, p)$ is the Cantor network.

Corollary 4.4.2 The Cantor network is WSNB for multicast under the θ-window algorithm if and only if $p > 2^{\theta-2} + \theta \cdot 2^{n-\theta-1} - \theta/2 + 2^{1-\theta} - 2^{1-n} + 1/4$ (0 if $\theta = 1$), for $n \geq 3$.

4.5 Optimization

Let $f(\theta, m)$ denote the maximum number of blockings required in Theorem 4.4.1 for given θ and m. In this section, we determine optimal θ^0 for given n and m, and also compare the optimal solutions among different m.

$f(\theta, 0)$ is decreasing in θ for $\theta \le \lfloor n/2 \rfloor - 1$. Hence, $\theta^0 = \lfloor n/2 \rfloor - 1$ in that range. Since

$$f\left(\left\lfloor \frac{n}{2} \right\rfloor - 1,\ 0\right) - f\left(\left\lfloor \frac{n}{2} \right\rfloor,\ 0\right)$$
$$= \left[\left(\left\lfloor \frac{n}{2} \right\rfloor - 1\right) \cdot 2^{\lceil n/2 \rceil} + 2^{n-2\lfloor n/2 \rfloor + 1} - 1\right] - \left[2^{\lfloor n/2 \rfloor} + \left(\left\lceil \frac{n}{2} \right\rceil - 2\right) \cdot 2^{\lceil n/2 \rceil - 1}\right] > 0 \ \text{ for } n \ge 3,$$

we conclude for $m = 0$ and $n \ge 3$, $\theta^0 \ge \lfloor n/2 \rfloor$. It was shown in [29] that $\lceil n/2 \rceil$ is a better choice than $\lfloor n/2 \rfloor$. Since $f(\theta, 0)$ for $\theta \ge \lfloor n/2 \rfloor$ has a unique minimum, we can start with $\lceil n/2 \rceil$ and increase the window size until $f(\theta, 0)$ increases. In general, θ^0 grows slowly with rate and can be quickly found.

$f(\theta, 1)$ is decreasing in θ for $\theta \le \lfloor n/2 \rfloor - 1$.

Since

$$f\left(\left\lfloor \frac{(n+1)}{2} \right\rfloor - 1, 1\right) - f\left(\left\lfloor \frac{(n+1)}{2} \right\rfloor, 1\right)$$
$$= \left[\left(\left\lfloor \frac{(n+1)}{2} \right\rfloor - 1\right) \cdot 2^{\lceil (n-1)/2 \rceil} + 2^{n-2\lfloor (n+1)/2 \rfloor + 1} - \frac{1}{2}\right]$$
$$- \left[2^{\lfloor (n-1)/2 \rfloor} + \left(\left\lceil \frac{(n-1)}{2} \right\rceil - 1\right) \cdot 2^{\lceil (n-1)/2 \rceil - 1}\right] > 0 \ \text{ for } n \ge 3$$

$\theta^0 \ge \lfloor (n+1)/2 \rfloor$. Again, $f(\theta, 1)$ has a unique minimum, and $\lfloor n/2 \rfloor + 1$ is a good value to start the upward searching.

Finally, for $m \ge 2$, we note that $f(\theta, m)$ is increasing in m for all $\theta \ge m$. Since a larger m implies more stages and larger cost, there is no reason to consider $m > 2$ when it costs more but performs worse. For $\theta \geq m = 2$

$$f(\theta, 2) = \begin{cases} \theta \cdot 2^{n-\theta-1} + 2^{n-2\theta-1}, & \text{for } \theta \leq \left\lfloor \frac{n}{2} \right\rfloor \\ 2^{\theta-2} + (n-\theta) \cdot 2^{n-\theta-1} + \frac{1}{4}, & \text{for } \theta \geq \left\lfloor \frac{n}{2} \right\rfloor + 1. \end{cases}$$

The first equation is decreasing in θ in its range. Hence, $\theta^0 = \lfloor n/2 \rfloor$.

Since

$$f\left(\left\lfloor \frac{n}{2} \right\rfloor, 2\right) - f\left(\left\lfloor \frac{n}{2} \right\rfloor + 1, 2\right)$$
$$= \left\lfloor \frac{n}{2} \right\rfloor \cdot 2^{\lceil n/2 \rceil - 1} + 2^{n-2\lfloor n/2 \rfloor - 1} - 2^{\lfloor n/2 \rfloor - 1} - \left(\left\lceil \frac{n}{2} \right\rceil - 1\right) \cdot 2^{\lceil n/2 \rceil - 2} - \frac{1}{4} > 0 \text{ for } n \geq 4$$

$\theta^0 \geq \lfloor n/2 \rfloor + 1$. $f(\theta, 2)$ has a unique minimum and $\lfloor n/2 \rfloor + 1$ is a good value to start the upward searching.

We next compare the optimal solutions for $m = 0, 1, 2$. We will only compare the starting values in the search process.

$$f\left(\left\lceil \frac{n}{2} \right\rceil, 0\right) = 2^{\lceil n/2 \rceil} + \left(\left\lfloor \frac{n}{2} \right\rfloor - 2\right) \cdot 2^{\lfloor n/2 \rfloor - 1}$$
$$f\left(\left\lfloor \frac{n}{2} \right\rfloor + 1, 1\right) = 2^{\lfloor n/2 \rfloor} + \left(\left\lceil \frac{n}{2} \right\rceil - 2\right) \cdot 2^{\lceil n/2 \rceil - 2}$$
$$f\left(\left\lfloor \frac{n}{2} \right\rfloor + 1, 2\right) = 2^{\lfloor n/2 \rfloor - 1} + \left(\left\lceil \frac{n}{2} \right\rceil - 1\right) \cdot 2^{\lceil n/2 \rceil - 2} + \frac{1}{4}.$$

Clearly, $f(\lfloor n/2 \rfloor + 1, 1) < f(\lceil n/2 \rceil + 0)$.

Furthermore

$$f\left(\left\lfloor \frac{n}{2} \right\rfloor + 1, 1\right) - f\left(\left\lfloor \frac{n}{2} \right\rfloor + 1, 2\right) = 2^{\lfloor n/2 \rfloor - 1} - 2^{\lceil n/2 \rceil - 2} - \frac{1}{4} \geq 0.$$

So $m = 2$ does better in minimizing the number of copies required. However, we have to recall that a copy with $m = 0$ or $m = 1$ costs less. For all three m values, the number of crosspoints is about $O(N^{3/2}\log^2 N)$.

According to the above result, we choose $m = 2$, and compute the best choice of θ and the corresponding value of p for each n in Table 4.5.1.

Note that for $n = 17$, two θ's yield the same m-value. For larger n in the table, we show the p-values mainly for mathematical interest, not for practical use.

Table 4.5.1 Best choice of θ and corresponding value of p for $m = 2$ and some n.

n	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
θ	2	3	4	4	5	5,6	6	7	7	8	8	9	9	10	10,11
p	3	4	6	9	13	21	29	45	65	97	145	209	321	449	705

n	18	19	20	21	22	23	24	25	26	27	28	29	30	31
θ	11	12	12	13	13	14	14	15	15	16	16	17	17	18
p	961	1473	2049	3073	4353	6401	9217	13313	19457	27649	40961	57345	86017	118785

n	32	33	34	35	36	37	38	39	40	41
θ	18	19	19,20	20	21	21	22	22	23	23
p	180225	245761	376833	507905	770049	1048577	1572865	2162689	3211265	4456449

n	42	43	44	45	46	47	48	49	50
θ	24	24	25	25	26	26	27	27	28
p	6553601	9175041	13369345	18874369	27262977	38797313	55574529	79691777	113246209

Intuitively, one would expect the larger m is, the more connecting power the $\mathrm{Log}_2(N, m, p)$ is, and hence, the fewer copies are needed for nonblocking. One would also expect the optimal m grows with N. We obtain the surprising result that $m = 2$ is optimal universally. But this is a technical result, for which we have no insight into why it is so. Nonetheless, it is a very valuable result, since regardless of how large is N, we need only to use moderate-size $\mathrm{Log}_2(N, m, p)$, i.e., $\mathrm{Log}_2(N, 2, p)$, which are relatively inexpensive to construct.

Like all routing algorithms, the delayed splitting algorithm restricts the scope of ways in connecting a multicast call. But it also restricts the scope of interference a multicast connection has on other requests. It is a tradeoff whose net value we do not know for sure. However, the delayed splitting algorithm simplifies routing to a degree that an analysis of the nonblocking condition becomes tractable.

Chapter 5 Conclusions

For the triple-loop networks, there are not many known good shapes, i.e., short diameters with given N, to work with, and the existence of a given shape is sparse. Besides, there is no systematic way to optimize the parameters of a given shape. In the first part of this thesis, we greatly expanded the families of H_1 and H_2 to broaden their applicability. We also proposed a method to choose better parameters of H_1' and H_2', thus improving their efficiency. Finally, we gave 3-diameters of H_0, H_1' and H_2' by constructing three node-disjoint paths. It follows that after two arbitrary failures (nodes or links) the diameters of these triple-loops are at most $D + 2$.

For the $\text{Log}_2(N, m, p)$ networks, Tscha and Lee [44] stated in their conclusion that whether their approach to multicast WSNB problem could be extended to $\text{Log}_2(N, m, p)$ was unclear. Danilewicz and Kabacinski [13, 14] also made an attempt to extend their results to $\text{Log}_2(N, m, p)$, but encountered some difficulties. In the second part of this thesis, we extended the WSNB results of multicast $\text{Log}_2(N, 0, p)$ network to multicast $\text{Log}_2(N, m, p)$ network. Then we compared our variable-size result for $m = 0$ with Tscha and Lee's result, and our result improves over Tscha and Lee's result. Finally, we obtained the surprising result that $m = 2$ is optimal universally.

We propose the following topics for further research:

In chapter 3, we have proved that the triple-loops H_0, H_1' and H_2' are 3-connected by constructing 3 node-disjoint paths from any node i to any other node j. For another family proposed by Chen and Gu [7] with a better efficiency 0.078, the wide-diameters are not known yet.

By theorems 3.2.4, 3.3.6 and 3.4.6, we know that the k-diameters of H_0, H_1' and H_2' are at most $D + k - 1$ for $k = 1, 2, 3$, respectively. Can we prove that the k-diameter of every triple-loop is at most $D + k - 1$ for $k = 1, 2, 3$?

In section 4.4, the WSNB result on the $\text{Log}_d(N, m, p)$ network for multicast under the θ-window algorithm is not known. Besides, the results on $\text{Log}_d(N, m, p)$ network under other routing algorithms are unknown, too.

Reference

[1] D.P. Agrawal, Graph theoretical analysis and design of multistage interconnection networks, *IEEE Trans. Comput.*, C32, pp. 637-648, 1983.

[2] F. Aguiló, M.A. Fiol, An efficient algorithm to find optimal double loop networks, Discrete Math. 138, pp. 15-29, 1995.

[3] F. Aguiló, M.A. Fiol, C. Garcia, Triple loop networks with small transmission delay, Discrete Math. 167/168, pp. 3-16, 1997.

[4] F. Agulió-Gost, New dense families of triple loop networks, Disc. Math. 197/198, pp. 15-27, 1999.

[5] V.E. Beneš, Mathematical Theory of Connecting Networks and Telephone Traffic. New York: Academic, 1965.

[6] F.H. Chang, J.Y. Guo, F.K. Hwang, Wide-Sense Nonblocking For multi-$\log_d N$ Networks under Various Routing Strategies, preprint, 2004.

[7] S. Chen, W. Gu, Exact order of subsets of asymptotic bases, J. Number Theory 41, pp. 15-21, 1992.

[8] C. Chen, C.S. Hung, The existence of two types of hyper-L triple-loop networks, Master Thesis, Department of Applied Mathematics, National Chiaotong University, Hsinchu, 2002.

[9] C. Chen, F.K. Hwang, The minimum distance diagram of double-loop networks, IEEE Trans. Comput., vol. 49, pp.977-979, 2000.

[10] C. Chen, F.K. Hwang, J.S. Lee, S.J. Shih, The existence of hyper-L shapes in triple-loop networks, Disc. Math. 268, pp. 287-291, 2003.

[11] M.Y. Chen, F.K. Hwang, C.H. Yen, Tessellating shapes in the plane, preprint, 2001.

[12] C. Chen, F.K. Hwang, K.Y. Lan, Equivalence of buddy networks with arbitrary number of stages, preprint, 2004.

[13] G. Danilewicz and W. Kabacinski, $\text{Log}_2(N, m, p)$ broadcast switching networks, in *Proc. Int. Conf. Communications*, Helsinki, Finland, pp. 604–608, June 2001.

[14] G. Danilewicz and W. Kabacinski, Wide-sense nonblocking multicast switching networks composed of $\text{Log}_2 N + m$ stages, in *Proc. IEEE Int. Conf. Telecommunications*, Bucharest, Romania, pp. 519–524, June 2001.

[15] H.M. Dao, C.B. Silio Jr., Ring-network reliability with a constrained number of consecutively-bypassed stations, IEEE Trans. Rel. 47, pp. 35-43, 1998.

[16] D.M. Dias and J.R. Jump, Analysis and simulation of buffered delta networks, IEEE Trans. Comput. C-30, pp. 273-282, 1981.

[17] P. Esqué, F. Aguiló, M.A. Fiol, Double commutative-step digraphs with minimum diameters, Discrete Math. 114, pp. 147-157, 1993.

[18] D. Farmer, E. Newhall, An Experimental Distributed Switching System to Handle Bursty Computer Traffic, Proc. ACM Symposium on Problems in the Optimizations of Data Commns., pp. 1-23, October 1969.

[19] M.A. Fiol, On congruence in Z^n and the dimension of a multidimension circulant, Disc. Math. 141, pp. 123-134, 1995.

[20] M.A. Fiol, J.L.A. Yebra, I. Alegre, M. Valero, A discrete optimization problem in local networks and data alignment, IEEE Trans. Comput. 36, pp. 702-713, 1987.

[21] F. Hsu, On container width and length in graphs, groups, and networks, IEICE Transactions of Fundamentals of Electronics, Communications and Computer Science, V.E77-A, No.4, pp. 668-680, 1994.

[22] D.F. Hsu, T. Luczak, Note on the k-diameter of *k*-regular *k*-connected graphs, Discrete Mathematics, 132, pp. 291-296, 1994.

[23] J.Y. Hui, Switching and Traffic Theory for Integrated Broadband Networks. Boston, MA: Kluwer, 1990.

[24] F. K. Hwang, Choosing the best Log(N, m, p) strictly nonblocking networks. *IEEE Trans. Commun.*, 46, pp. 454-455, 1998.

[25] F.K. Hwang, A complementary survey of double-loop networks, Theor. Comput. Sci. 263, pp. 211-229, 2001.

[26] F.K. Hwang, The mathematical theory of nonblocking switching networks, World Scientific, Singapore, 1998.

[27] F.K. Hwang, B.C. Lin, *k*-Diameters of the hyper-L shape tile, Journal of Interconnection Network, vol. 3, Nos. 3&4, pp. 245-252, 2002.

[28] F.K. Hwang, B.C. Lin, Wide-sense nonblocking multicast $\log_2(N, m, p)$ networks, *IEEE Trans. Commun.*, 51, pp. 1730-1735, 2003.

[29] W. Kabacinski and G. Danilewicz, Wide-sense and strict-sense nonblocking operation of multicast multi-$\log_2 N$ switching networks, *IEEE Trans. Commun.*, vol. 50, pp. 1025–1036, June 2002.

[30] D.H. Lawrie, Access and alignment of data in an array processor, *IEEE Trans. Comput.*, 25, pp. 1145-1155, 1976.

[31] C.T. Lea, Multi-$\log_2$ networks and their applications in high-speed electronic and photonic switching systems. *IEEE Trans. Commun.*, 38, pp. 1740-1749, 1990.

[32] C.T. Lea , D.J. Shyy, Tradeoff of horizontal decomposition versus vertical stacking in rearrangeable nonblocking networks, *IEEE Trans. Commun.*, vol. 39, pp. 899–904, June 1991.

[33] T.T. Lee, Nonblocking copy networks for multicast packet switching, *IEEE J. Select. Areas Commun.*, vol. 6, pp. 1455–1467, Dec. 1988.

[34] B.C. Lin, F.K. Hwang, Generalizing and fine tuning triple-loop Networks, preprint, 2005.

[35] M. Listani and A. Roveri, Switching structures for ATM, *Computer Commun.*, vol. 12, pp. 349–358, Dec. 1989.

[36] M.T. Liu, Distributed Loop Computer Networks, Advances in Computers, Vol. 17, Academic Press, New York, pp. 163-221, 1981.

[37] J. Pierce, Network for Block Switches of Data, Bell Syst. Tech. Jour., Vol. 51, No. 6, pp. 1133-1145, July/August 1972.

[38] C.S. Raghavendra, J.A. Sylvester, A survey of multi-connected loop topologies for local computer networks, Comput. Network ISDN Syst. 11, pp. 29-42, 1986.

[39] Schrijver, P. D. Seymour and P. Winkler, The ring loading problem, SIAM J. Disc. Math. 11, pp. 1-14, 1984.

[40] D.J. Shyy and C. T. Lea, $\text{Log}_2(N, m, p)$ strictly nonblocking networks, *IEEE Trans. Commun.*, vol. 39, pp. 1502–1510, Oct. 1991.

[41] H.J. Siegel, S.D. Smith, Study of multistage SIMD interconnecting networks, Proc. 5th Ann. Symp.

Comput., pp. 307-314, 1978.

[42] H.S. Stone, The organization of high-speed memory for parallel block transfer of data. IEEE Tans. C-19, 1, pp. 47-53, 1970.

[43] Y. Tscha and K. H. Lee, Non-blocking conditions for multi-$\log_2 N$ multiconnection networks, in *Proc. GLOBECOM*, pp. 1600–1604, 1992.

[44] Y. Tscha and K.H. Lee, Yet another result on multi-$\log_2 N$ networks, *IEEE Trans. Commun.*, vol. 47, pp. 1425–1431, Sept. 1999.

[45] J. Wolf, M.T. Liu, B. Weide, D. Tsay, Design of a Distributed Fault-Tolerant Loop Network, The 9th Annual Int. Symp. Fault-Tolerant Computing, Madison, pp. 17-24, June 1979.

[46] J. Wolf, B. Weide, M.T. Liu, Analysis and Simulation of the Distributed Double Loop Computer Networks, Proc. Computer Networking Symposium, NBS, Gaithersberg, pp. 82-89, December 1979.

[47] C.K. Wong, D. Coppersmith, A combinatorial problem related to multimode memory organizations, J. ACM 21, pp. 392-402, 1974.