Using Graph Enumeration and Topography Reasoning to Analyze Blocking in WDM Networks Without Wavelength Interchange

by

Xiao MA

B.S of Communication Engineering, BUPT, 2010

M.S in Telecommunications, University of Pittsburgh, 2012

Submitted to the Graduate Faculty of

[School of Information Science in partial fulfillment

of the requirements for the degree of

M.S in Telecommunications

University of Pittsburgh

2012

UNIVERSITY OF PITTSBURGH

SCHOOL OF INFORMATION SCIENCES

This thesis was presented

by

Xiao MA

It was defended on

Aug 2, 2012

and approved by

Richard Thompson, Professor, School of Information Sciences

David Tipper, Associate Professor, School of Information Sciences

Roger Flynn, Associate Professor, School of Information Sciences

Thesis Director: Richard Thompson, Professor, School of Information Sciences

Using Graph Enumeration and Topography Reasoning to Analyze Blocking in WDM Networks Without Wavelength Interchange

Xiao MA, MST

University of Pittsburgh, 2012

This thesis presents techniques and analysis results which look promising for analyzing blocking in WDM networks without wavelength interchange (WLI). The approach includes two parts—graph enumeration and blocking analysis for graphs, which are linked by a theorem [1] that reduces blocking analysis for an entire WDM system to the blocking analysis of the backbone topography and link capacity. Graph enumeration utilizes Burnside's Theorem and MATLAB computation. Blocking analysis for graphs are implemented in MATLAB. Some general phenomena are observed from the analysis result of graphs with four, five, and six nodes. Node resilience is also discussed.

TABLE OF CONTENTS

PREFACE

This thesis is considered a milestone for my long time keen on mathematics. I have been enjoying the rigorous logic beauty brought about by mathematics since I was a child. Mathematicians use their magnificent ways of reasoning and analysis to improve human's recognition to nature. I love the content and wisdom from mathematics.

I appreciate my parents' long time encourage on my interest in mathematics. Mathematics does not guarantee imminent achievement and is not appreciated by most people's career plans. But my parents keep my heart of science in this field. They told me that I should feel proud of myself since I am working, or studying some science, that once before, changed the world.

I do feel lucky that I make acquaintance with my current advisor—Dr. Thompson, who sees and recognizes my love and potential on mathematics. I do appreciate Dr. Thompson's commitment for this thesis work. He works so patient pick up every single grammar mistake from the virgin scholar-purpose script from a foreign student. Without his guidance and instructions I could not go this far.

I have to say thanks to all my friends who help and encourage me during my hardest time in the US in the past two years, especially Shu Ye and Qun Yu. To some extent, my friends double my joy and divide my frustration in my academic journey in the US.

1.0 CHAPTER 1 INTRODUCTION

1.1 PURPOSE AND STRUCTURE OF THIS THESIS

This thesis presents techniques and analysis results which look promising for analyzing blocking in WDM networks without wavelength interchange (WLI). The approach includes two parts—graph enumeration and blocking analysis for graphs, which are linked by a theorem [1] that reduces blocking analysis for an entire WDM system to the blocking analysis of the backbone topography and link capacity. Graph enumeration utilizes Burnside's Theorem and MATLAB computation. Blocking analysis for graphs are implemented in MATLAB. Some general phenomena are observed from the analysis result of graphs with four, five, and six nodes. Node resilience is also discussed.

This paper contains two logic parts. The first part is closely related to graph enumeration, which enables the counting of different topographies. The second part is related to the topic of blocking analysis. Both parts are intense in mathematics. The graph enumeration part reviews enumeration theory and gives some computer enumeration results. The blocking analysis part reviews some basic rules in blocking analysis. Chapter 2 introduces enumeration theory, which can be used to enumerate different n-node graphs considering symmetry. The theoretic approach is hard and opaque. This approach has high computational complexity with increase of n. This approach, currently, does not consider connectivity.

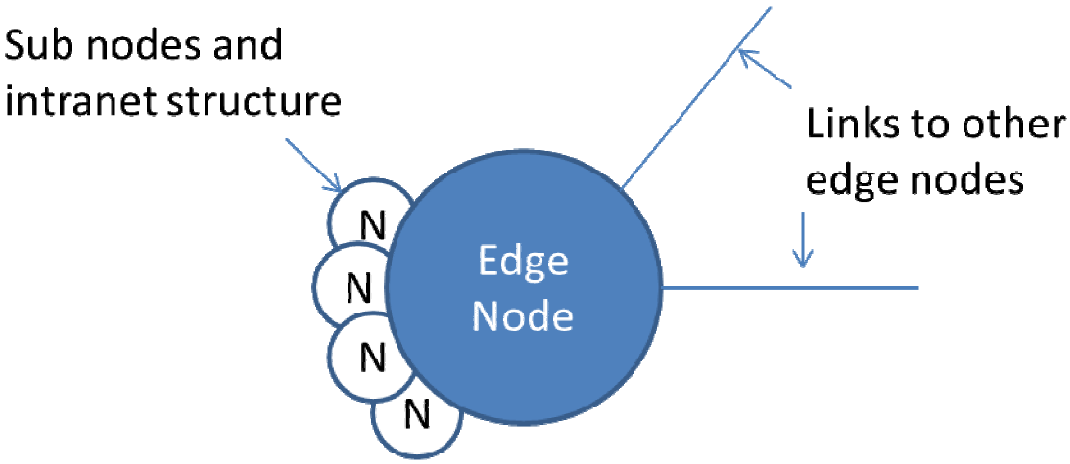
Chapter 3 introduces the background for the computer enumeration of n-node connected graphs using a recursive approach generating all possible candidate n-node connected graphs from “n-1” node graphs. After generating all the candidate graphs, we can filter out duplicate candidates. The results are the non-isomorphic connected n-node graphs. The theoretic background for blocking analysis is also included in this chapter. Complex traffic is reduced and modeled as traffic patterns. Blocking analysis is approached by examining all possible traffic patterns.

Chapter 4 analyzes the result of the previous enumeration blocking analysis. Some rules are implied and indicated. The topography affects whether the graph is blocking or not. Non-blocking graphs have two distinct categories—star topography non-blocking graphs and non-Star non-blocking graphs. Analysis of these two categories is done in the scope of node failure resilience. Chapter 5 concludes, and summarizes the content and logic of the entire paper.

This thesis does NOT completely solve this problem. Its purpose is to demonstrate techniques that promise to approach a solution.

1.2 BRIEF INTRO TO WDM BACKBONE NETWORK

An optical WDM transmission scheme is similar to the FDMA scheme in electrical transmission. Instead of frequencies, wavelengths are the source of the transmission medium. A WDM backbone network can serve many edge nodes. The edge nodes serve to switch inter-node traffic and to transmit and receive inter-node traffic to and from the backbone network. An edge node can be illustrated as the following:



Pic 1.1 Edge node

The links connecting edge nodes in a WDM network can be considered containing certain number of wavelengths as transmission and receiving resource. An edge node can transmit and receive data on multiple wavelengths at the same time. This paper discusses inter-node traffic in such WDM networks.

The infrastructure of optical WDM is different from the electrical FDMA system in that the wavelength interchanger technology does not have the performance of its counterpart in FDMA systems. It is easy to modulate and demodulate electric signals. But the optical "modulator" and "demodulator"— the wavelength interchanger—is less sophisticated. Besides, WLI is expensive, and is not likely to be cheap in the upcoming future. WLI also suffers from scale problems when the number of wavelengths is large. So there is great potential to research the topic of WDM networking without WLI, while meeting the requirement of data connectivity.

In chapter 3 the analysis of a non-blocking configuration for WDM networks without WLI is discussed. Networks are represented as graphs. Traffic patterns are represented as permutations.

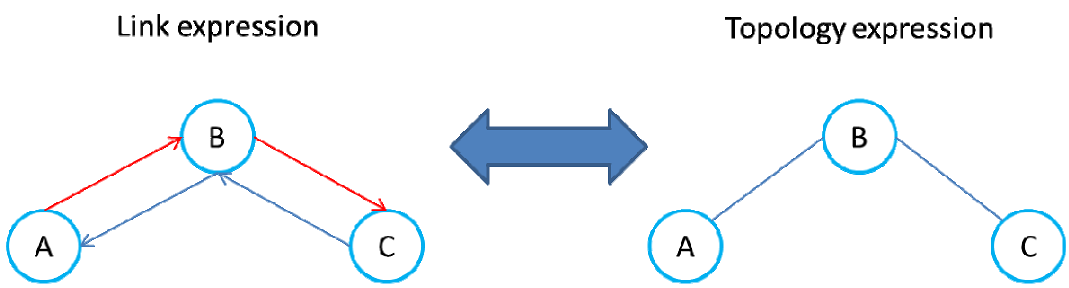
1.3 INTRODUCTION TO BLOCKING IN WDM NETWORK

For the simplicity of stating the concept, the assumption that every link has a capacity of 1 wavelength is used in the illustrations and examples in this section. In this section the transmission mechanism and a blocking analysis are briefly introduced.

An edge node's functions and limitations in the scope of the whole network are listed below. At a particular time, an edge node can:

- Process intra-node traffic without interfering with any inter-node traffic capacity
- Send data from itself to one other destination via one link
- Receive data destined to this node from one link
- Forward tandem information as long as the incoming link and outgoing link is available.

The statements change a little in multiple wavelengths per link scenario. Substitute the term "link" by "one particular wavelength in the link". The following picture depicts a toy network illustrating the infrastructure and topography of a WDM network. Pic 1.1 assumes one wavelength per link. It should be noted that the wavelength in every link on the path for a particular data stream is the same so as to avoid WLI.



Pic 1.2 Traffic AC taking path ABC

Considering the left part of Pic 1.2, suppose A is sending data to B, and B is sending data to C, via the red links, and C is sending data to A through B via the blue link. In this scenario all the traffic mentioned above is not blocked. Though B can't receive any data destined to B from links other than the red, the blue traffic can also pass through B since the blue links, which are connected to B, and at the same time required by the traffic passing through B, are all available. Though the data from the incoming, passing, and outgoing links are all on the same wavelength, they do not collide because they are processed by different interfaces.

In the WDM network structure mentioned above, intra-node traffic does not affect inter-node performance. But the network may suffer from a variety of blocking problems, in which case it ends up unable to serve some traffic. The discussion for network blocking analysis is covered in section 3.2.

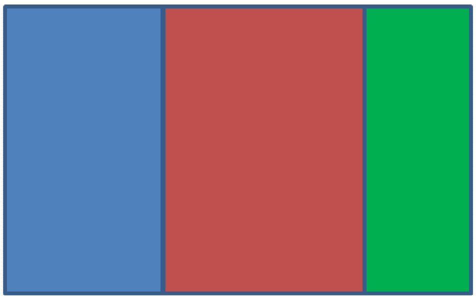
This thesis discusses the techniques for the analysis of blocking in WDM networks without WLI. A WDM network without WLI is an approach which aims to reduce the cost and increase the system reliability and robustness for optical networks.

2.0 CHAPTER 2 ENUMERATION THEORY

This chapter introduces Burnside's theorem, and its application to graph enumeration. Graph enumeration is different from other general enumeration problems in that symmetry between graphs should be considered. In more precise terms, isomorphic graphs should be considered as one pattern during the enumeration. Different topography configurations are desired but not graphs with same topography and different appearance.

2.1 INTRODUCTION OF ENUMERATION WITH SYMMETRY

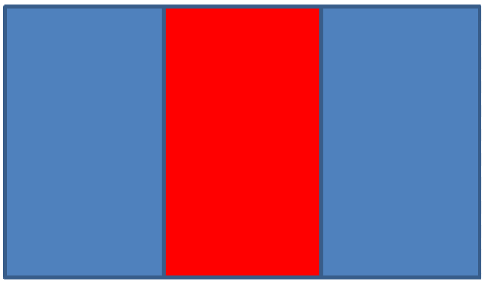
This section begins with an example leading to the core concept of enumeration considering the symmetry factor. Consider a case of the flag painting problem, in which the flag has k stripes with different width. Each stripe is to be painted a specific color which is picked from n different choices. For simplicity take k=3 and n=3 in the example case. The flag is like the following picture:



Picture 2.1 Flag without symmetry

In this case, enumeration of all painting patterns is easy because every stripe is different, thus every coloring method produces a different pattern. The final result is n^k patterns for this case.

It would be different if the 3 stripes are of the same width(see figure 2, it seems that red stripe is a bit narrow but it is an optical illusion.). In this scenario the color method [Red Blue Green] and [Green Blue Red] are considered to be the same pattern. Under this constraint introduced by the property of symmetry, duplicated patterns should not be counted.



Picture 2.2 Same stripe flag

This problem can be solved another way, by dividing the patterns into 2 categories. The symmetric coloring such as [Red Green Red] is considered as 1 pattern itself. The non-symmetric colorings such as [Red Green Blue] and [Blue Green Red] are considered as 2 different coloring methods but they produce the same pattern. In the following content of this paper, a pattern denotes a set of coloring methods which produce a mutually symmetric result. A color method simply represents a single way of coloring without considering the property of symmetry.

It can be observed that: in this case, one symmetric coloring method contributes one pattern(pattern RGR only contains color method RGR); two non-symmetric coloring contributes one pattern(pattern RGB is composed of coloring method RGB and BGR). It should be noted that this difference is the hard issue for the enumeration with symmetry problem.

The overall 27 coloring method can be divided into 2 classes under the previous discussion: 9 of 27 color methods are self-symmetric coloring methods; the remaining 18 color methods are non-symmetric coloring methods. So an overall of $9 + 18 \div 2 = 18$ patterns are enumerated for this case. This fact means that, considering symmetry, there are only 18 possible painting results.

Restate previous calculation into alphabet variables:

$$\text{Enumeration} = \frac{\text{overall} - \text{symmetry}}{2} + \text{symmetry} = \frac{\text{overall} + \text{symmetry}}{2} \quad (2.1)$$

To further investigate this kind of problem, intuitive reasoning becomes insufficient when there are multiple ways of “symmetry” defined. In order to investigate the multiple symmetry case, more effort needs to be conducted to delve into core of this problem, based on the previous case.

2.2 A POLYNOMIAL TRICK

A polynomial trick can be implemented to investigate the previous case in detail. Denote each choice of color as a distinct variable, e.g. denote Red as x, Blue as y, Green as z. In the scenario where symmetry is not considered, each stripe to be painted can be expressed as the polynomial $(x+y+z)$. In this scenario, if there are 3 stripes to be painted, the painting result can be expressed, in the polynomial way: $(x+y+z)^3$. Expand this polynomial $(x+y+z)^3$:

$$x^3+y^3+z^3+3x^2y+3xy^2+3x^2z+3xz^2+3y^2z+3yz^2+6xyz \qquad (2.2)$$

Inspect every part of the previous polynomial. $3x^2y$ means that there are three ways to paint the flag into two red(denoted by x) stripes and one blue(denoted by y) stripe. By expanding the polynomial, certain paint patterns can be observed. By setting all the variables to one, the polynomial becomes an integer—27—which is the previous result.

For the scenario considering “symmetry”, just like the case indicated by picture 2.1, some tricks can be used in this case presenting the effect of “symmetry”. In this case the problem is approached according to (2.1) by calculating the “overall” and “symmetry” patterns respectively. “Overall” patterns are already calculated by (2.2) above. The trick for symmetry case is as below.

The fact is utilized that the left stripe is symmetric to the right stripe. In this case, the left stripe and the right stripe can be considered as a single unit which is colored twice with same color, thus eliminating the confusion brought about by the property of symmetry. The polynomial for the symmetry case is: $(x^2+y^2+z^2)(x+y+z)$. x^2 means that 2 stripes are forcefully regulated to be painted by red color, which indicates the symmetry. Expand the previous polynomial:

$$x^3+y^3+z^3+x^2y+xy^2+x^2z+xz^2+y^2z+yz^2 \qquad (2.3)$$

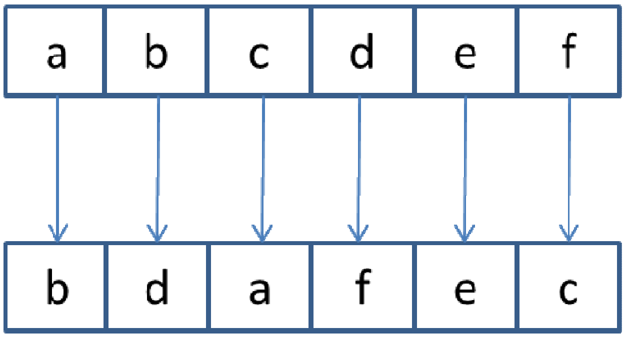
Compare this expansion with (2.2), it can be observed that, first, no three color(xyz) pattern is enumerated; then, only one pattern of painting is enumerated for the flags which only has two-and-one color patterns. It matches the observation that no any three color painting is symmetric and for each two-and-one color pattern there is only 1 pattern which is symmetric, e.g. only RGR is symmetric among all possible three patterns [RRG GRR RGR]. This observation validated our trick.

By setting all variables to 1, the result of the polynomial is 9, which corresponds to the previous intuitive calculation. More can be approached using this polynomial trick and the following Burnsides theorem is built on this fundamental, but first, the concept of permutation, which describes the property of symmetry, with the precise and rigorous mathematical manner, should be introduced.

It should be noted that, if the symmetry rule becomes that flipping the left and middle strips is considered the same, other than the previous flipping left and right strip, the result is the same. This phenomenon comes from the fact that, the 2 symmetry rules are similar in structure.

2.3 BRIEF INTRODUCTION TO PERMUTATIONS

Intuitively, the concept of permutation can be expressed as a rearrangement of the order of some objects, or a reshuffle of the objects. The reshuffle process of a set of poker cards is vivid example of permutation. Consider a set of objects $\{a, b, c, d, e, f\}$ being listed by some order in a stack like following picture:

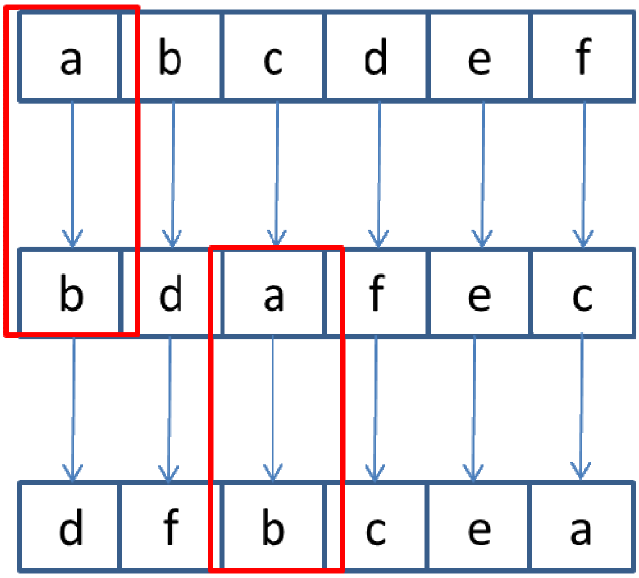


Pic 2.3 Illustration for Permutation

In the stack above, the original sequence is abcdef. After a reshuffle process, the sequence becomes bdafec. This process can be described as a permutation as following:

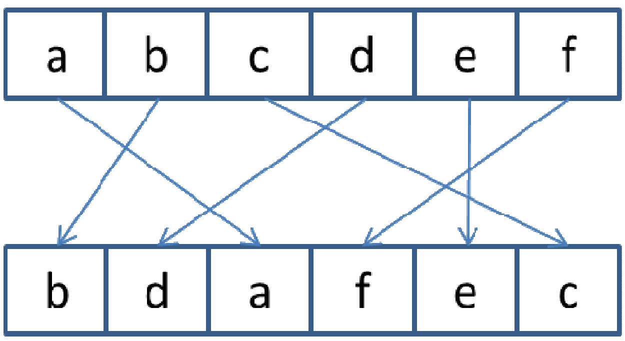
(abdfc)(e)

The content abdfc in the parenthesis means that if a particular slot has content a, the permutation changes the content a to b for that slot; if a particular slot contains content b, the permutation changes it to d, …, and if a particular slot contains content c, the permutation changes it to a. (e) means that after the permutation the content for the slot containing e is not changed. The no-change permutation—more formally—identity permutaion, in this scenario, is written as (a)(b)(c)(d)(e)(f). This permutation is formally called identity permutation. If we further deploy the same permutation for the previous reshuffle result once more, it would become:



Picture 2.4 Stack after two same reshuffle

Permutation is an abstract concept from intuitive behavior, aiming to solve more complicated problems. It is important to notice that, during the application of permutation, the definition of the permutation should be clear. For the previous reshuffle problem, there is another way to apply the concept of permutation to deal with the problem. It is very tricky and sometimes confusing. Consider the following picture restating this problem:

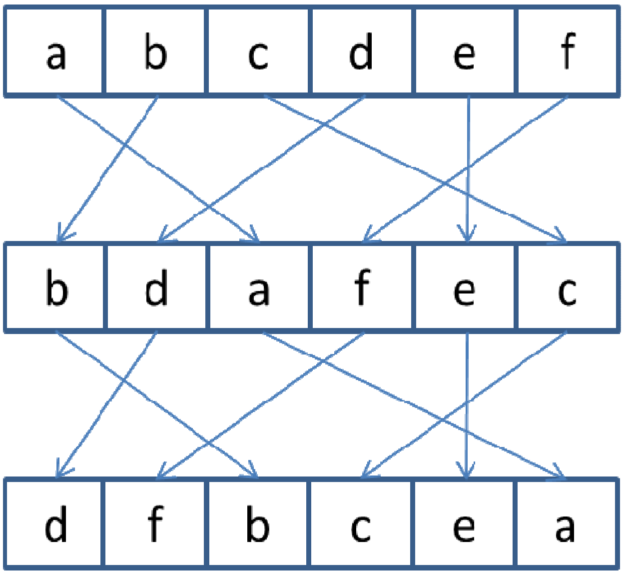


Picture 2.5 Restatement of the reshuffle problem

In this case, consider the reshuffle process as slot oriented, not content oriented as above. It can be observed that, slot a takes the position of slot c, slot c move to position of slot f, f moves to d, d moves to b, b moves back to a, and e remains unchanged. So in this way the permutation can be written as:

(acfdb)(e)

By re-applying this permutation again, the result is:



Pic 2.6 the result of 2 same shuffle in the 2nd approach

It can be seen that, the result is same as the previous content oriented approach. Though the ways to approach the problem, and the permutation in each approach is not the same, the result is the same. This fact indicates that, during the application of permutation to practical problem, how the permutation is constructed must be clear and precise to avoid ambiguous definition, which is a disaster validating the solution process, and may bring about very ridiculous solutions.

2.4 PERMUTATION AS AN OPERATION

As mentioned above, Permutations can be intuitively considered as some reshuffle method for a set of cards. Consider concatenating reshuffle methods: α_1, α_2, as stated below.

The expression: $\alpha_1 \cdot \alpha_2$ means that, for the original card sequence, first, shuffle it with method α_2, and then shuffle it with method α_1. It should be noted that this operation between the permutations does NOT satisfy the commutative property, like the matrix multiplication.

It can be intuitively deduced that there exists a permutation $\alpha_3 = \alpha_1 \cdot \alpha_2$. It means that there is a particular shuffle, which is equivalent to the two sequential shuffles. Based on this, no matter how and how many shuffle processes one experienced, the result can be achieved by only one particular shuffling. This particular shuffling is equivalent to all the shuffle process it experienced.

Reconsider the permutation mentioned in section 2.3: Denote:

$$\alpha_1 = (abdfc)(e)$$

It can be inferred that $\alpha_1^5 = (a)(b)(c)(d)(e)(f)$, which is the identical permutation(no-change reshuffling). This fact can be extended that the sequence comes back to the very beginning state if shuffling with same method for some particular limited times.

Now, formally, consider the permutations as the elements in a set. That is, for a set defined as: $S = \{\alpha_1, \alpha_2, \alpha_3, \cdots, \alpha_n\}$ and the permutation concatenation as the operation defined on set S, S can be considered as a Group if S satisfies the following properties:

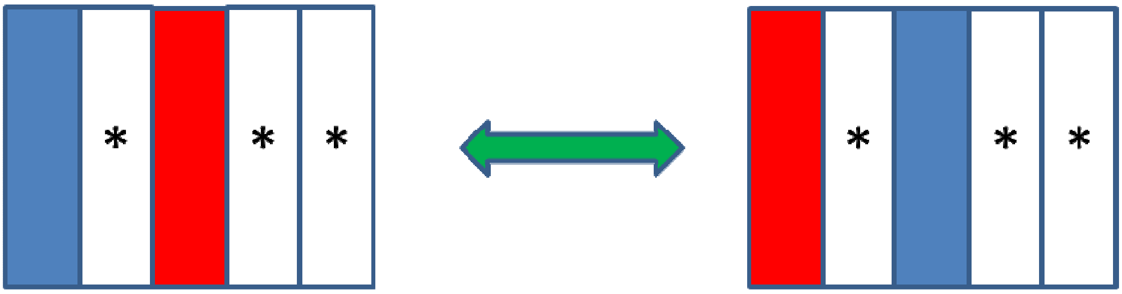
- S is closed for the operation "·"
- Commutative law: for every $x, y, z \in S$, $x \cdot (y \cdot z) = (x \cdot y) \cdot z$
- Identity: there exists an element $e \in S$ st for every $x \in S$ $e \cdot x = x \cdot e = x$, e is the

Identity

- Inverse: for every $x \in S$ $\exists x' \in S$ st $x' \cdot x = x \cdot x' = e$.

The proof of the above properties are too deep in theory and is covered in abstract algebra textbooks. For this reason, this paper does not provide scrutinized discussion or proof on this topic.

Permutations can be applied to describe the "symmetry" property in the flag coloring problem mentioned in section 2.1. The symmetry pattern, can be written as a permutation (13)(2). This permutation presents the idea that, for some color method, if the colors of 1st and 3rd stripe are switched, the new flag is considered the same color pattern with the original color method, thus depicting "symmetry". Also some odd and unintuitive "symmetry" pattern can be presented in a mathematics manner with the permutation expression. For example, consider a 5 stripe flag and the odd symmetry permutation: (13)(2)(4)(5). This permutation indicates a strange symmetry pattern, which is illustrated below.

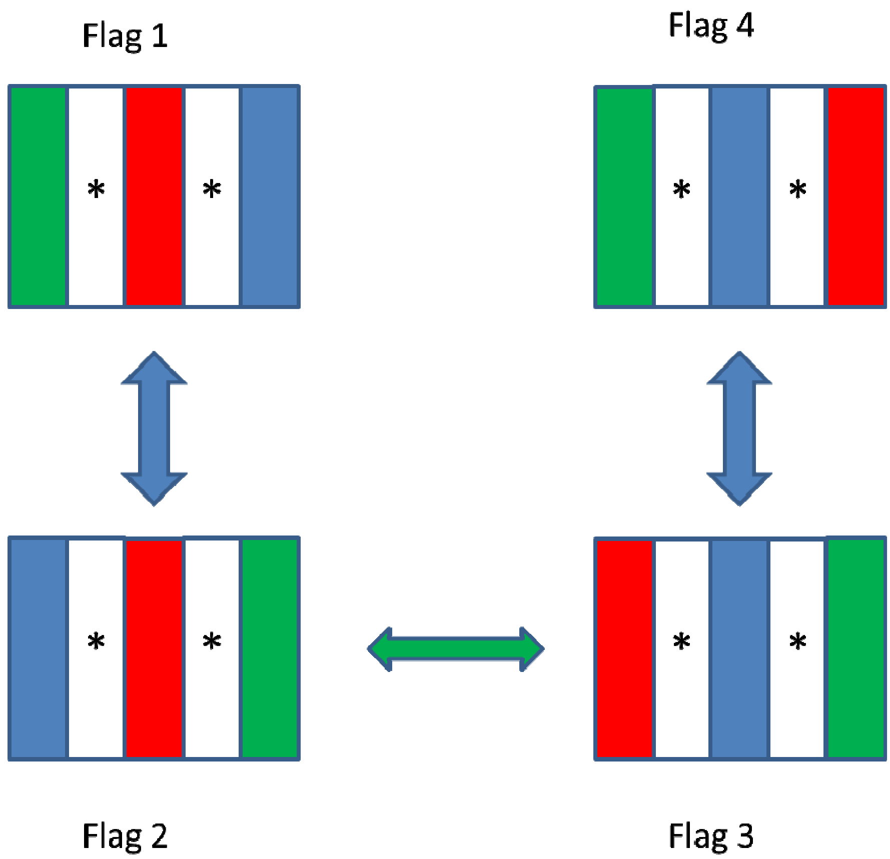


Pic 2.7 An odd symmetry pattern

In the picture above, the left flag is considered symmetric to the right flag under the symmetry rule defined by the permutation (13)(2)(4)(5). This unintuitive symmetry rule seems unreasonable and deviates from common sense. But this freedom to depict symmetry leads the way to solve some difficult symmetry enumeration problem, which is a hard approach by intuitive reasoning and enumeration.

It should be noticed that symmetry is deducible. For a case with multiple different rules for symmetry, the enumeration problem becomes significantly more difficult. The reason for this includes that the number of symmetry rules becomes large, and that the combination of some of the rules may lead to a new rule, which is not explicitly mentioned. The following case illustrates this potential rule problem.

Consider the previous flag-painting scenario in which both the "mirror symmetry"(15)(24)(3) and the "unintuitive symmetry" (13)(2)(4)(5) mentioned above are considered. The following picture clearly illustrates how multiple heterogeneous symmetry rules generate the color pattern.



Pic 2.8 example for multiple heterogeneous symmetry rules

In the picture above the blue arrow indicates that the 2 flags are considered as same color pattern under the mirror symmetry rule. The green bidirectional arrow indicates that the 2 flags are considered as same color pattern under the odd symmetry rule. It is obvious that Flag 1 and Flag 2 are symmetric according to the mirror symmetry rule. Flag 2 and Flag 3 are symmetric according to the odd symmetry rule. Flag 3 and Flag 4 are mirror symmetric. Clearly, under the multiple symmetry rules, all 4 flag belongs to the same color pattern. But it would be unintuitive to judge whether Flag 1 and Flag 3 are the same given the 2 clearly mentioned rules since the direct permutation from Flag 1 to Flag 3 is (153)(24), which is not explicitly stated. This new permutation is the concatenation of the 2 permutation: $(15)(24)(3)\cdot(13)(2)(4)(5)$. (153)(24) is one of the implied symmetric rule of this case. Also, the permutation from Flag 1 to Flag 4—(1)(35)(2)(4)—is another implied symmetric rule.

Given the facts above, it can be derived that, some particular permutations should occur together, forming precise and integrated symmetry rules. It can be inferred that, the set of the permutations describing the symmetry should form a Group in algebra structure, so as to be mathematically precise for next step of analysis. By forming all the permutations into a Group, the set of permutations are closed for permutation concatenation. This means that, in the Group, all possible potential symmetry rules are included. It would not be necessary to claim all possible potential patterns when describing the symmetry rules by language. But it should be rigorous from the scope of mathematical purpose.

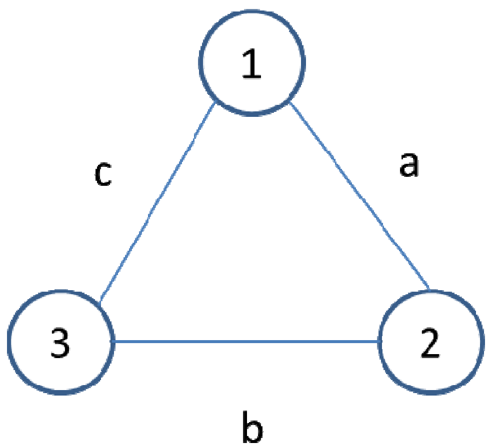
2.5 SYMMETRY IN GRAPH ENUMERATION

A graph is defined as a set of nodes V, and a set of edges E which connect the nodes in V. Not necessarily all the nodes are connected. In this paper only undirected graphs are studied in the scope of symmetry.

It is straightforward to investigate the symmetry patterns in the graph enumeration problem. The problem of finding different graph, or more formally, non-isomorphic graphs, is not as easy as the flag-painting case because a lot of unintuitive symmetry rules. That means that a particular graph has a lot of chances to become some other graph with same structure but a different appearance.

Consider a graph with n nodes. This graph would allow a maximum of $\frac{n(n-1)}{2}$ edges, denoted as k_{max}. Consider the case that the k_{max} edges are stripes to be colored with two choices—connected or not—under the set of symmetry rules that depicts graph isomorphism. This approach is analogous to the flag-painting case. But the following problem, brought about from this approach, need to be specified: how does one analyze the isomorphism from the scope of permutation on the edges?

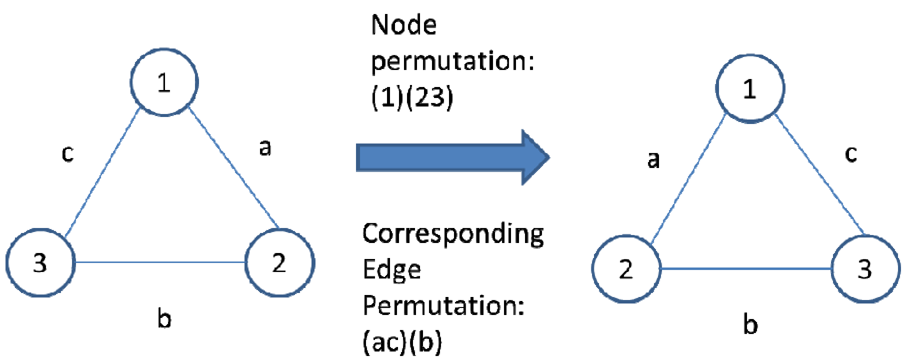
Consider a simple 3 node case, denote the 3 nodes as Node1, Node2, and Node3, denote the edge connecting Node1 and Node2 as edge a, denote the edge connecting Node2 and Node3 as edge b, denote the edge connecting Node1 and Node3 as edge c. The following picture illustrates this scenario:



Pic 2.9 Graph Definition Illustration

The number of nodes does NOT indicate that the 3 nodes are different but serve as labels which help us to enumerate the isomorphism situation clearly. The method—"collect" the permutation of edges—is to find the change of position of edges which follows the reshuffle of nodes. While moving the nodes, it should be noticed that, the edges are moved according to the nodes they connect, so as to guarantee the same structure.

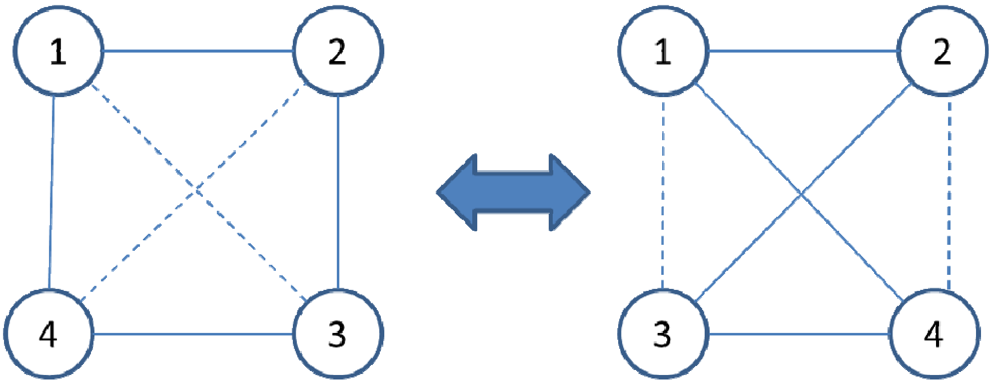
It should be noted that the final key to graph enumeration must be the permutation of edges, which are to be "painted" according to the approach. It is the edge that serves as a bridge connecting the logic of enumeration theory and the problem of graph enumeration. This process can be illustrated by the picture below. The permutations defined below are the changing-content permutations, just as the Pic2.4 indicates. The other approach is not introduced in this illustration.



Pic 2.10 Illustration of enumerating edge permutation

The statement can be extended as follows. For every possible permutation on n NODES, there is a corresponding edge permutation. For an n-node graph, there are n! possible node permutations, so are the corresponding edge permutations. All of the edge permutations depict a graph isomorphism, and at the same time, form an algebra structure— The Permutation Group.

It seems easier to get the previous result by observing that the right graph above is symmetric to the left by flipping it about the central axis. But this intuitive symmetry can only be applied in easy cases with a few nodes. For a more complicated 4-node graph it is less intuitive to figure out all possible symmetry patterns as illustrated below. In the following case, intuition-based reasoning and enumeration are not reliable. Duplicated patterns and missed patterns are likely to occur. It is never easy to observe the fact that, the two graphs in Pic 2.11 are isomorphic to each other with intuitive reasoning.



Pic 2.11 Unintuitive Isomorphism Pattern

2.6 ENUMERATION WITH SYMMETRY—BURNSIDE'S THEOREM

This section presents a step by step derivation with vivid examples illustrating unintuitive methods of symmetry reasoning. Burnside's Theorem uses the combination of polynomial trick and the permutation at an abstract level, solving more complicated problems.

The case used in Section 2.1, as illustrated in Pic 2.2 needs to be investigated further to help find the key to go further. The following table lists all possible color patterns listed for the case.

Table 2.1 Pattern enumeration

Enumerate Everything(according to (1)(2)(3))								
RRR	RRG	RRB	RGR	RGG	RGB	RBR	RBG	RBB
GRR	GRG	GRB	GGR	GGG	GGB	GBR	GBG	GBB
BRR	BRG	BRB	BGR	BGG	BGB	BBR	BBG	BBB
Enumerate Symmetric pattern according to (13)(2)								
RRR	RBR	RGR	GRG	GBG	GGG	BRB	BBB	BGB

The patterns under the symmetry constraint are shown in green color in order to distinguish them. The final 18 color patterns can be shown in the following table.

Table 2.2 Pattern classification using Table 2.1 label

1	2	3	4	5	6	7	8	9
RRR	RBR	RGR	GRG	GBG	GGG	BRB	BBB	BGB
RRR	RBR	RGR	GRG	GBG	GGG	BRB	BBB	BGB
10	11	12	13	14	15	16	17	18
RRG	RRB	RGG	RBB	RGB	RBG	GBB	GRB	BGG
GRR	BRR	GGR	BBR	BGR	GBR	BBG	BRG	GGB

It can be observed from above table that:

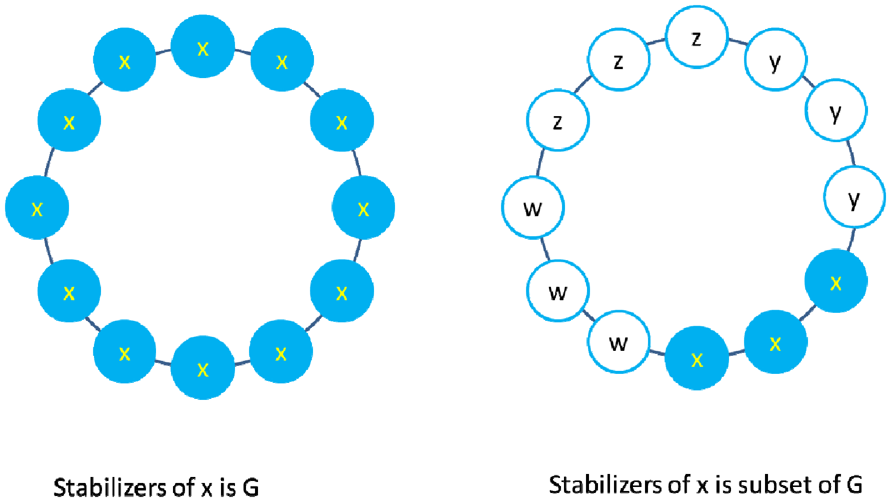
- Each Symmetric coloring contributes to a pattern by itself
- Each of non-symmetric coloring methods contributes to a color pattern with a partner color method, which is symmetric to it according to (13)(2)

From the above reasoning, it can be observed that the permutations and the coloring patterns should be both considered in this complicated approach. For a particular coloring method, if it comes back to itself after some permutation, the permutations can be denoted as the stabilizers of the coloring method. It can be inferred that,the identity permutation is a stabilizer of any color method. It can be observed that, each pattern from the patterns 1-9 in Table 2.2 contains only 1 color method. All the color methods labeled 1-9 in Table 2.2appear to have 2 stabilizers since both identity permutation and symmetric rule (13)(2) stabilize them. For patterns 9-18, each pattern has two distinct color methods, and each of the two color methods has one stabilizer respectively (identity permutation).

Formally, consider set $S = \{x_1, x_2, \cdots, x_n\}$, and a Permutation Group G defined on S. For $x \in S$, if there exist some $g \in G$ st $g(x) = x$, then $g \in Stab(x)$, g is called one of the stabilizers of x. Some mathematical statement constraining the Permutation Group G and set of S is not stated here for convenience. In this analysis the set S is the abstract expression of the "all possible coloring methods" in the case mentioned above. And each x, $x \in S$, represents a particular possible color method. Some color method, such as [RRR], is highly symmetric and has larger number of stabilizers. Some less symmetric color method such as [RGB] has less stablizers.

It can be proved that for any x(abstract term of a "color method"), the stabilizers of x form a permutation subgroup of G, which means that all the elements in the subgroup forms a group with the same operation—permutation concatenation—defined in G. It can also be inferred that, for any subgroup of G, the identity permutation must be included.

Now is clear that, with the same assumptions as above, and a Permutation Group G illustrating symmetry, there must be some color pattern, the stabilizer of which is the permutation G. This means that there exists some highly symmetric elements. Each of the elements, after any of the permutations in G, is itself. This is illustrated below.



Pic 2.12 Color pattern distribution

The circles on the circumstance of the large circle represent the permutations in the permutation group G. It should be noted that the large circle is just a visual expression and does not imply that G is a ring. The left part of the picture indicates that, for some highly symmetric color method x, which does not change after any permutation, contributes to a color pattern itself. The right part indicates that, for some less symmetric x, the whole circle is composed of some different colors, which form a color pattern. This phenomenon is intuitively reasoned as following:

First, if for some $g \in G, g(x) = w, w \neq x$, it means that w and x are mutually "symmetric" defined by Permutation Group G. That is, at least, w and x are considered as the same pattern.

Second, denote $H = stab(x)$, then, H is a subgroup of G. Investigate the left Cosets of H in G. Suppose $g, g_1, g_2 \in G, g_1, g_2 \notin H, h \in H, \exists h_1 \in H \text{ st } g_2 = g_1 h$, then $g_2(x) = g_1 h(x) = g_1(x) \neq x$. Denote $g_1(x) = w$, then for $\forall h \in H, \exists g' \in G \text{ st } g' = g_1 h \text{ and } g'(x) = w$. This means that, as expressed in Pic 2.12, if there are $|H|$ x-es in the large circle, there would also be $|H|$ w-s, which are different from x, in the circle. $|H|$ is the integer representing the number of permutations in subgroup H.

According to Lagrange' theorem and some related theorem, all different left cosets of H in G has same number of elements and do not have any single common elements. In this way, every Cosets mentioned above represents a particular "color method", which is part of a "color pattern".

From the reasoning above, the enumeration question is reduced to enumerating all the stabilizers for every x (color method) in the set S. It is not easy to enumerate the number of stabilizers according to x since every x is different in appearance and have different number of stabilizers. But it can be approached by enumerating the number of elements a particular permutation stabilizes, utilizing the polynomial trick. For a particular permutation, the number of elements it stabilizes can be calculated as following:

$$F(g) = \prod_{j} \sum_{i=1}^{n} x_i^{r_j}$$

By setting all x=1, F(g) is the number of elements a permutation stabilizes. The variable n is the number of colors, and r_j is the length of each ring in the permutation. For example, a permutation like (abcd)(e)(f)(gh) on 3 colors, the $F(g)$ is:

$$(x_1^4 + x_2^4 + x_3^4)(x_1 + x_2 + x_3)(x_1 + x_2 + x_3)(x_1^2 + x_2^2 + x_3^2)$$

Bunside's Theorem is, in short: $N = \frac{1}{|G|}\sum_{g\in G} F(g)|_{x_i=1}$, where N is the desired enumeration of different elements under permutation group G.

2.7 APPLYING BURNSIDE'S THEOREM TO GRAPH ENUMERATION

The 1st step for the non-isomorphic graph enumeration problem is to gather the permutation for the graph, which depicts the symmetry. Enumeration by hand, utilizing the method mentioned in Section 2.5, for a 4 node graph, with the notation similar to Section 2.5 , the permutation group contains the following 24 permutation:

Table 2.3 Permutation for four node graph

(a)(b)(c)(d)(e)(f)	(a)(be)(c)(df)	(aeb)(cfd)	(adcb)(ef)
(ae)(b)(cf)(d)	(a)(bd)(c)(ef)	(aecf)(bd)	(adf)(bec)
(a)(bf)(c)(de)	(abe)(cdf)	(abf)(cde)	(af)(b)(ce)(d)
(aed)(bcf)	(abcd)(ef)	(ab)(cd)(e)(f)	(afb)(ced)

(ade)(bfc)	(afce)(bd)	(ac)(bd)(e)(f)	(ac)(bedf)
(ad)(bc)(e)(f)	(afd)(bce)	(ac)(bfde)	(ac)(b)(d)(ef)

Then for every edge, there are two color choices for them: Connected or Not Connected. Applying this to the Burnside's Theorem, the result is:

$$N = \frac{1}{|G|}\sum_{g\in G} F(g)|_{x_i=1}$$

$$= \frac{1}{24}((x_1 + x_2)^6 + 9(x_1 + x_2)^2(x_1^2 + x_2^2)^2 + 8(x_1^3 + x_2^3)^2 + 6(x_1^2 + x_2^2)(x_1^4 + x_2^4))$$

The above expression, after setting all variable to 1, results in 11, which indicates that, there are 11 non-isomorphic graphs with 4 nodes.

CHAPTER 3 CONNECTIVITY AND BLOCKING

This chapter introduces the methodology of connected graph enumeration and the property of blocking under certain assumptions. Connected graphs are hard to enumerate by a purely mathematical method since it is hard to define “connectivity” in the problem of graph enumeration. MATLAB is used to help enumerate all possible topography configurations in a recursive approach.

Each different graph can be considered as a different topography configuration for networks consisting of particular number of nodes. The network must be connected and non-blocking for the system configuration. There are different types blocking situations, which will be investigated respectively.

3.1 ENUMERATING CONNECTED GRAPHS

The process of enumerating connected graphs includes the method to enumerate all connected graphs and the method to distinguish whether a new graph is isomorphic to an already found graph. It can be inferred that, for every n-node connected graph, it must have evolved from an n-1 node connected graph, which is also connected. Based on this fact, there is a recursive approach to enumerate all possible n-node connected graphs by step-by-step enumeration according to the number of nodes they have. The way to distinguish the newly generated graphs includes the function of identifying whether a particular graph is a new one, or in formal words, whether it is isomorphic to some previous known solutions. MATLAB provides this distinguishing function in the Graph Theory toolbox. The graph isomorphism scenario was introduced in Chapter 2.

For a particular connected graph with n nodes, there are $2^n - 1$ possible ways to generate an n+1 node connected graph. By adding a new node, there are n possible edges, and each possible edge has 2 choices: connect this edge, or not. It should be noted that, not only how many edges among the n are connected, but also how the nodes get connected, affect the topography. So every choice generates an (n+1)-node connected graph with the exception of the scenario that none of the new n edges are connected.

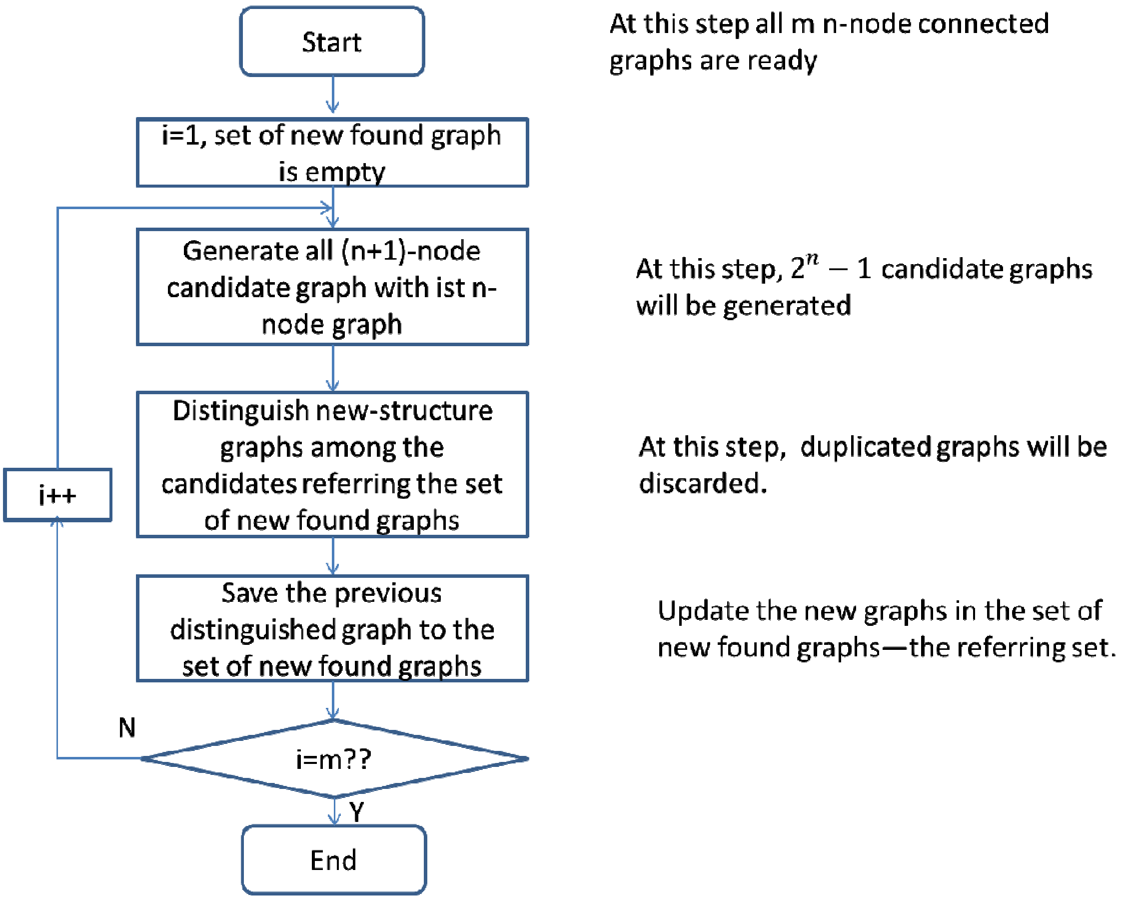
Suppose there are m non-isomorphic connected graphs consisting of n nodes, there are overall $m \cdot (2^n - 1)$ connected graph candidates with n+1 nodes. It is clear that some of the candidates are isomorphic, thus reducing the enumeration result.

It should be noted that, for every isomorphism identifying process:

- For all already acquired connected graphs with n nodes, set i=1;
- Generate $(2^n - 1)$ n+1 nodes graph from the ith graph by some sequence
- If one of the newly generated graphs—it may also be called candidate graph—is isomorphic to none of the previous found graphs, denote it is a new found graph
- The new found graphs are labeled according to their number of nodes and number of edges. For those different graphs with same number of nodes and same number of edges, arrange them by the sequence in which they are found.

- Consider this new found graph as one of the "previous found" graph to prepare for the next run
- i++

The previous steps can be shown in the following flow chart. The term "set of new found graph" denotes the set of non-isomorphism graphs that have been found out by the program. This set is used to distinguish new graphs.



Flow chart 3.1

Some of the key points in the above method should be noted:

- A graph can only be isomorphic to a graph with the same number of nodes and same number of edges. This property reduces the computational complexity of the isomorphism identifying process. In the generation of an n-node graph, only n-node new found graphs need to be considered in the isomorphism identification process.
- If a particular graph is found as new graph, which means that it is not isomorphic to any previous found graphs, it is labeled and saved with the previous found graphs. The next candidate graph may be isomorphic to this graph. After every new-found graph update, there is one more graph to be compared while distinguishing the following candidate graphs.

When all the possible candidates are compared and distinguished, all of the new n+1 nodes connected graph enumeration is finished.

3.2 CATEGORIES OF BLOCKING

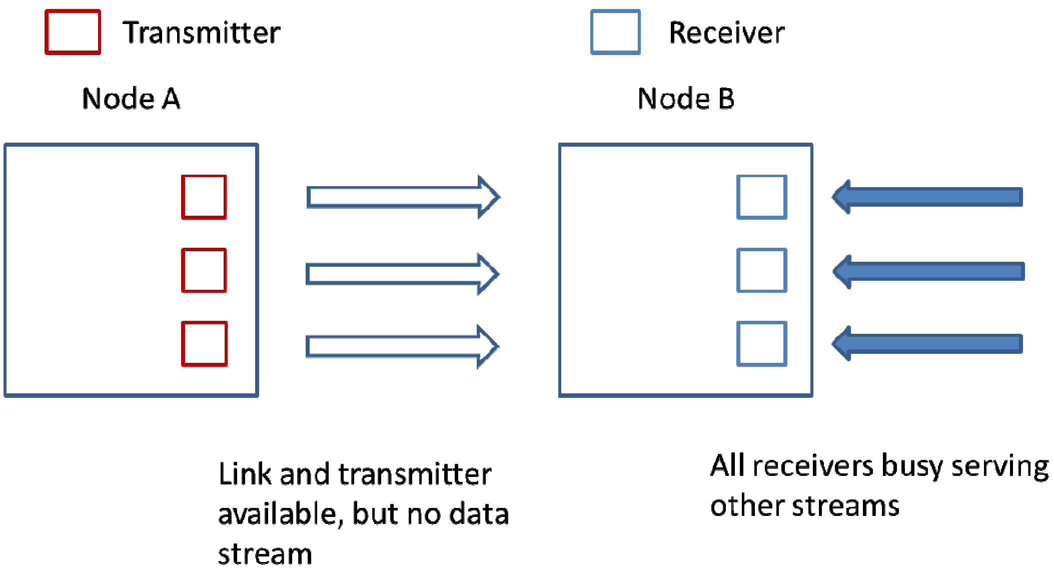
In the following paragraphs of this section, it is assumed that every link has multiple wavelengths. The concept and scenario of blocking is introduced under this generalized assumption. It should be noted that the node structure mentioned here is different from the edge nodes. This will be illustrated at end of this section.

Blocking occurs at a variety of different scenarios. The blocking scenarios can be classified in the following:

1. Mate blocking
2. Network blocking

While network blocking can be divided into two mutual exclusive sets of blocking: concentration blocking and path blocking.

Mate blocking occurs in the scenario where the link and path are available but physical transmitters, or receivers, are not available. The following picture indicates this scenario:

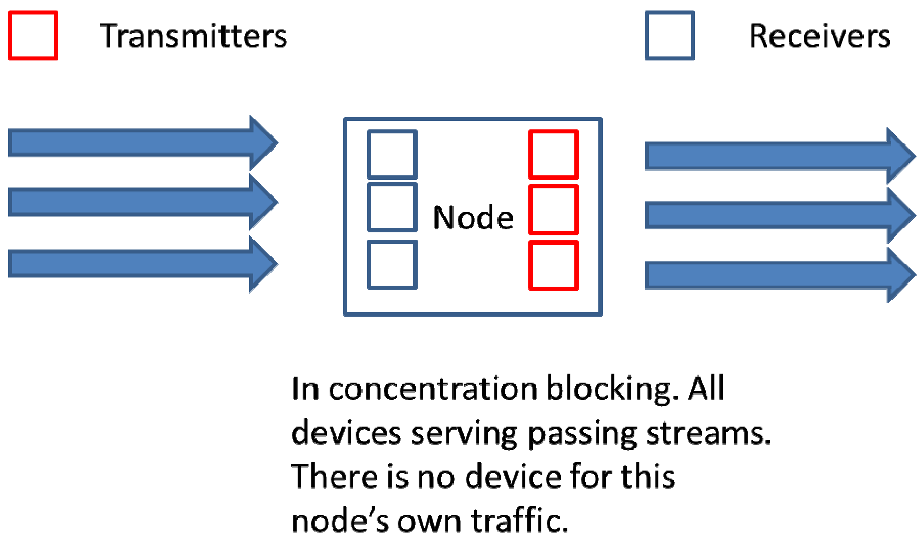


Pic 3.1 Mate Blocking

Pic 3.1 illustrates the scenario of mate blocking when all necessary receivers at terminal nodes are not available. In this picture, the Node B is the receiving node for the traffic originating from Node A. But all the receiving ports/devices are busy processing some other streams. At this scenario B can't handle the traffic from A, which is a mate blocking. Mate blocking occurs not only when receivers are not available, but when all necessary transmitters are not available as well. Mate blocking occurs in the scenario where the number of physical devices are less than the number of links/connections needed to serve. Some of lucky links enjoy the devices according to some mechanism such as FCFS, while the unlucky links are blocked when lucky links makes all devices busy. Mate blocking occurs only at the two nodes which are directly connected.

Network blocking describes the scenario that the network is unable to serve the stream while necessary physical devices are available at the two end nodes. In this situation, the exit and entrance are all available but a part of the road connecting the entrance and the exit may be jammed, or unavailable, hampering the transmission. Network blocking has two mutually exclusive parts: Concentration blocking and path blocking.

Concentration blocking is the situation where all the portals for the node are busy forwarding other stream so that its end points are blocking from transmitting or receiving data. If a node is said to be concentration blocking, the streaming originating from or destined to the node is blocked, but the stream forwarding the node is not blocked.

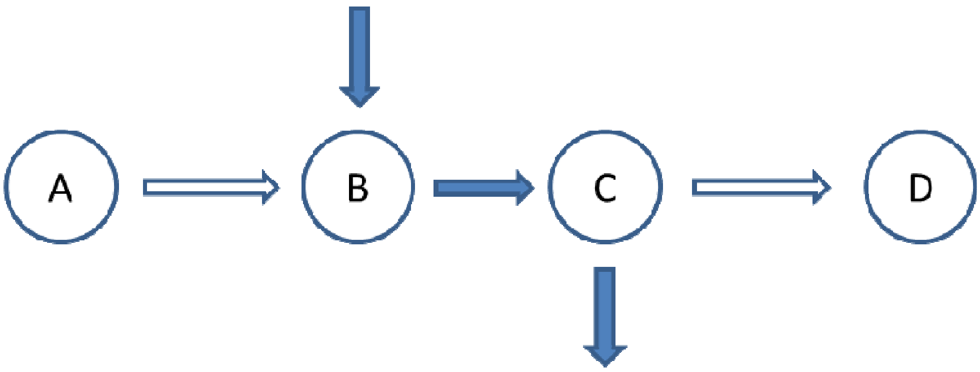


Pic 3.2 Concentration Blocking

In Pic 3.2, all the devices on the intermediate node is busy processing other materials, rendering the stream originating from or destined to the node blocked. In this situation the network should shift some stream passing through this node to another available path in order to make space for the node's own traffic.

This scenario should be noted that, it seems that concentration blocking and mate blocking are similar, but they have significant difference. Mate blocking occurs in a node-to-node scenario. We can rely on the two directly connected nodes to negotiate the solution for mate blocking. For concentration blocking, the network connecting the nodes is responsible for this issue, which is a complicated problem.

Path blocking happens when some or all the parts of the necessary path in the network for a stream to its destination is occupied, blocking the way of the stream. Path blocking is usual in networks and is caused by link congestion in computer networks. Traffic jam is a good analogy of this scenario.



Pic 3.3 Path Blocking

Picture 3.3 is the Path blocking situation, owning to the fact that the blue stream passing B and C occupied the B-C path, The A-D stream can't be served by this network. At this situation, both of the terminal nodes have devices and links available for the data stream. Note that A can send data destined to B in this situation that B can receive data while forwarding the blue traffic given that there is no mate blocking between A and B caused by the blue stream.

If there is one transmitter and one receiver deliberately assigned for every connected link in the edge node, mate blocking can be eliminated. That is, a WDM network composed of edge nodes with the configuration just mentioned does not suffer from mate blocking. Network blocking, which is the combination of concentration blocking and path blocking, is the only type of blocking in the network under this circumstance.

3.3 Traffic Patterns

The concept of Traffic pattern is introduced in this section to further investigate the network blocking issue. In this section of analysis, assume every link has only a capacity of 1 wavelength, for the simplicity of presenting the concept. The multi-wavelength scenario can be considered as multiple different crosstalk-free “1 wave length links”. At the beginning it is easy to investigate the one wavelength easy case.

This section discusses the traffic blocking problem from the perspective of traffic patterns. In this scenario, the traffic pattern is reduced so that it can be modeled as a permutation. The Validity of this model is stated later in this chapter. For example, consider a 4-node network, traffic pattern (1234) means that node 1 is sending data to node 2, 2 is sending to 3, 3 is sending to 4, 4 is sending back to 1. It should be noted, that, the fact 1 is sending to 2 represents that data from 1 are destined to arrive at 2, it does not obligate that the path connecting node 1 and 2 is used for this transmission. There may be no direct path connecting 1 and 2 in that network, but 1 can still send data to 2 from some other path. If the traffic pattern is (123)(4), (4) means that node 4 only has intra node traffic. (4) can’t be the destination of another stream and does not originate streams. All the links connected to (4) can be used to forward data streams.

It is of great value that a particular network topography would support every possible traffic pattern, a non-blocking routing configuration. It has to be noted that traffic pattern is an ideal concept in that every node has only one destination for data stream. The multiple destination scenario may be achieved by rearranging traffic patterns with time, given that the whole network is perfectly synchronized. This field remains blank.

3.4 REARRANGEABLY NON-BLOCKING

A network topography is said to be rearrangeably non-blocking if this topography is non-blocking for every possible traffic pattern. This means that, given the assumption that the overall transmission and route changes are perfectly synchronized; every possible transmission can be accomplished in the backbone network without WLI and blocking. It should be noted that a WDM network is more complicated than the network spatial topography. The topography of a particular WDM network depicts how the edge nodes are connected in the network. The endpoints a particular serves is not included in the analysis of the topography mentioned in this thesis.

A Rearrangeably non-blocking network is defined on the idea that the network can be non-blocking by shifting current paths to make space for new transmissions or data flow changes. Theoretically, if the network is controlled by an central authority and the control is highly synchronized, it can be considered that the network is rearrangeably non-blocking, if there is a route for each possible traffic pattern.

Reaarangeably non-blocking is feasible, as mentioned above, in the scenario that the network is non-blocking for every possible traffic pattern. Traffic pattern can be considered as a set of organized streams in the scope of network transmission. A network topography is non-blocking for a traffic pattern means that the streams in the pattern can arrive at their destinations in the network by some path at the same time. The fact that the a network topography blocks a particular traffic pattern indicates that, some of the streams in the traffic pattern can't arrive at their destinations at the same time. In other words, some of the streams in the traffic pattern could be considered mutually incompatible with each other. In Pic 3.4 stream A to D and stream B to C is mutually incompatible and can't be served simultaneously.

It is indicated in [1] that a theorem has been proved which connects the blocking analysis of WDM network and the blocking analysis for connected graphs according to traffic patterns. In [1] this result is derived from Clos structure inequalities [2]. The theorem can be restated as following:

"A WDM network is rearrangeably non-blocking without wavelength interchange if:

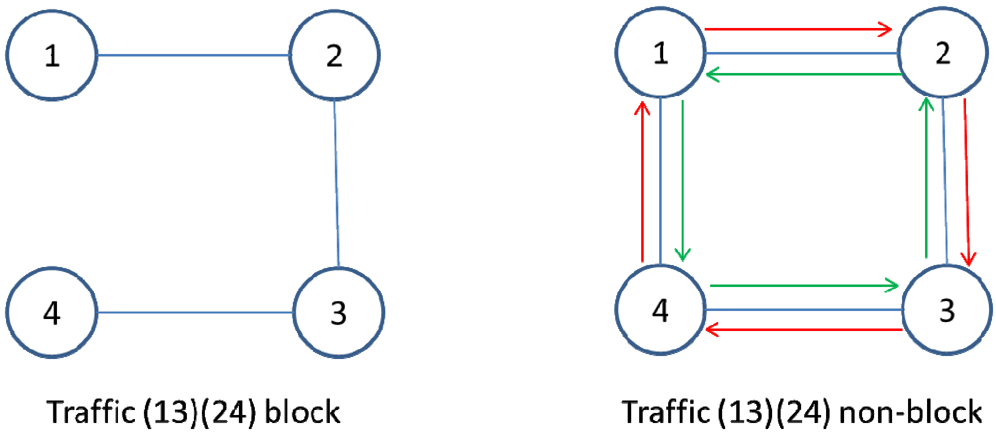
- *Each node's multiplexers satisfy the RNB Clos inequality ($m \geq n$) and*

- *The network's spatial topology is at least rearrangeably nonblocking."*

In the theorem above, m is the number of wavelength in each link. The variable n represents the maximum number of endpoints(subnodes) any node homes in the network. It should be noted that the 2 conditions must all be meet, making a WDM network necessarily non-blocking. This theorem indicates the importance investigating the blocking analysis for the network topography.

3.5 ROUTING ANALYSIS

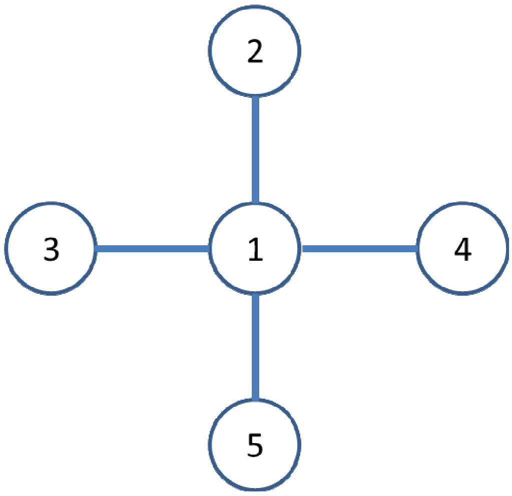
This section introduces the routing for non-blocking traffic patterns for some particular networks. If the graph has proficient connectivity, the routing problem for this network may be easier than that in a less-connected graph. For some less connected graphs, some traffic patterns may be blocked, or have only one solution, which may be hard to find.



Pic 3.4 Simple traffic case

In the left network in Pic 3.4 the traffic pattern (13)(24) blocks. But in the right network traffic (13) can be routed on the red path. (24) can be routed on the green path. By using this configuration this traffic pattern is not blocked. This observation indicates the fact that more connectivity provides greater opportunity for data streams reaching their destination.

How to find the routing path is another difficult problem. Generally speaking there is no trick to find a generalized routing problem in this scenario. Computers can be used to enumerate every possible solution and test whether there is a way available. It can be observed that the star topography is always non-blocking. The Star topography is the minimum connection configuration that is non-blocking over all possible patterns.



Pic 3.5 Star topography and routing

As illustrated in Pic 3.5, in a 5 node network there are 4 edges connecting them, which is the minimum connection requirement. The routing method for the streams in the traffic patterns can be list as following:

- Stream originating from center node: choose direct path to destination
- Stream destined to center node: choose direct path to destination
- Other Stream: send it to center, center forward it to destination

The advantages of star topography are: First, it requires the least connectivity to satisfy the non-blocking condition. Second, the routing method is easy. The path for any stream is at most 2 hops, and is convenient to manage. The disadvantage of this topography mainly includes lack of robustness. Failure on any link, any interface, or any node, would result in failure of part or the entire network.

From the scope of robustness, graphs with redundant connectivity are appreciated, which guarantees alternative routes when part of the network goes wrong. It also implies that a node has fewer edges connected to it in case of a node failure. If the node 1 in Pic 3.5 has some problem and is not working, all the 4 edges in the network are useless owning to the node failure. The next chapters include the data and analysis for the non-blocking and robustness property of graphs with 4, 5, and 6 nodes.

CHAPTER 4 DATA AND ANALYSIS

In this Chapter the simulation and calculation result of graph enumeration is presented. The number of different graphs with 4,5,and 6 nodes are listed. This chapter also includes the blocking analysis of the listed graphs.

4.1 GRAPH ENUMERATION

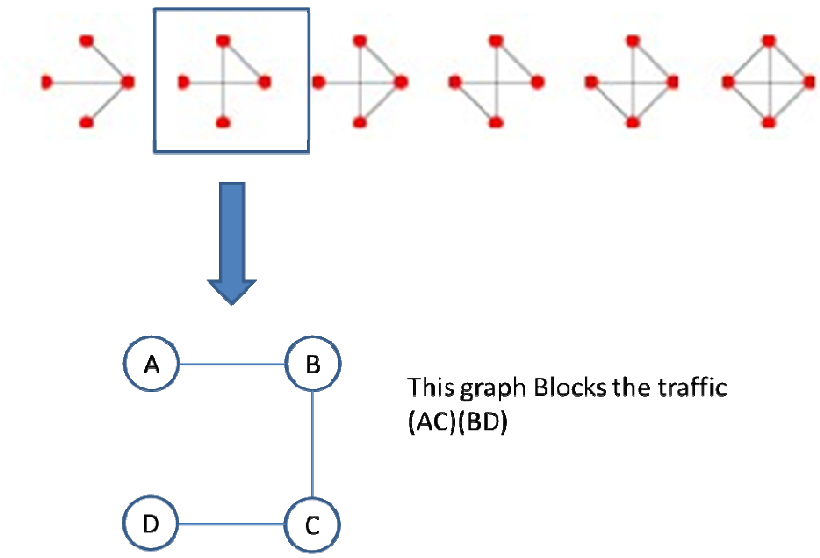
As stated in Section 2.7, non-isomorphic graph enumeration can be calculated with Burnside's Theorem—a combinatorial mathematics method. Section 2.7 already showed that there are 11 graphs enumerated. It would be very complicated to calculate the 5 nodes scenario since there would be 120 permutations in the permutation group. For 6 nodes scenario there would be 720 of them. The data for this is from OEIS A000088 and OIES A001349.

Table 4.1 Graph enumeration

	All different graphs	Connected different graphs	Ratio
4 Nodes	11	6	0.5455
5 Nodes	34	21	0.6196
6 Nodes	156	112	0.7179

It can be inferred that, as the number of nodes increases, the graph is more likely to be connected. The ratio of connected graphs takes more and more occupancy among all the graphs with increasing number of nodes.

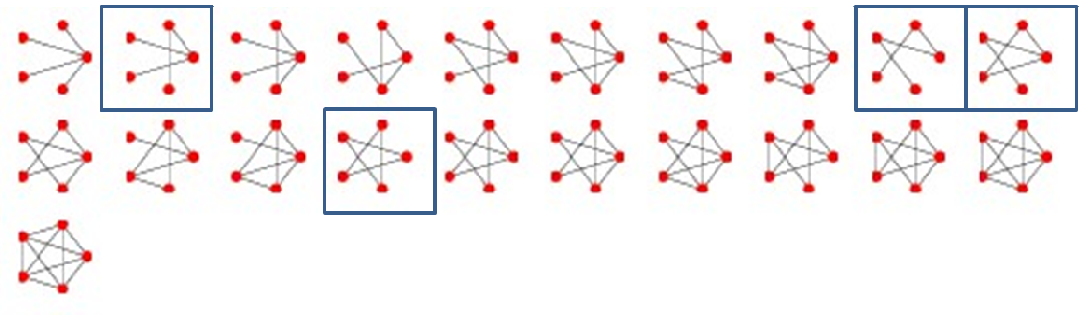
Among all the 6 connected graph with 4 nodes, only 1 of them blocks. This scenario is illustrated by the following picture. Blocking rate of four-node graph is 0.1666.



Pic 4.1 Blocking analysis of 4-node graph

It can be observed that, only one graph blocks and this graph contains the minimum number of edges to make it connected. The left most graph also contains 3 edges, but it is non-blocking owning to its star topology. It should be clear that the 4th graph from the left is the same as the square topography.

There are 21 different five-node connected graphs. Among them four graphs are blocked. This scenario is illustrated in the following picture.



Pic 4.2 Blocking analysis for 5 node graph

It can be observed that 4 of the 21 graphs are blocking. Two of the four blocking graphs have four edges. The two remaining graphs have five edges. All the graphs which have more than five edges don't block. The blocking rate for five-node graphs is 0.1905

For six-nodes scenario it is too complicated to list every graph since there are more than 100 of them. The analysis continues on blocking graphs and the number of edges of the blocking graphs.

Table 4.2 Block analysis for six-node graphs

Number of edges	Connected graphs	Blocked Graphs
5	6	5
6	13	12
7	19	8
8	22	3
9	20	0
10	14	0
11	9	0

12	5	0
13	2	0
14	1	0
15	1	0
Overall	112	28

It can be seen that 28 graphs block among all 112 different graphs. Graphs which have 9 or more edges don't block. The blocking rate for six-node graph is 0.25.

It can be observed that an n-node connected graph is less likely to be blocking with an increasing of number of edges. It is summarized in the following table.

Table 4.3 Blocking analysis according to number of edges

Number of nodes	Most edges for blocking	Fully connected
4	3	6
5	5	10
6	8	15

In the table above, it can be inferred that:

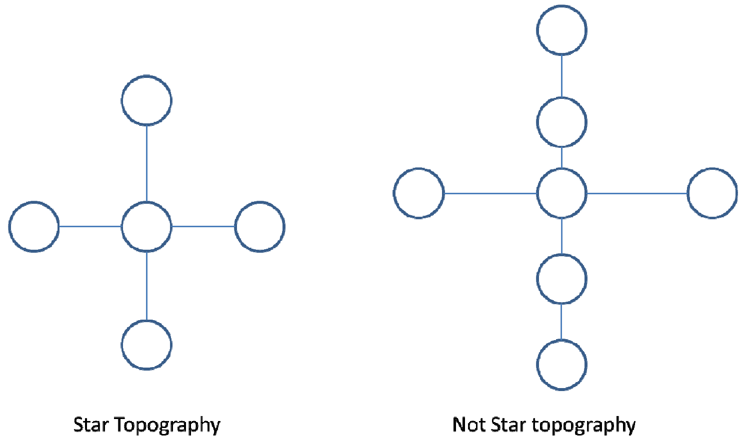
- For a four-node graph, if it has more than 3 edges, and it is connected, it is necessarily non-blocking
- For a five-node graph, if it has more than 5 edges, and it is connected, it is necessarily non-blocking
- For a six-node graph, if it has more than 8 edges, and it is connected, it is necessarily non-blocking

It should be noticed that the above statement is necessary condition for a graph to be non-blocking. A four-node connected graph with 3 edges may also be non-blocking. But a four-node connected graph with more than 3 edges, such as 4 edges, must be non-blocking, and can't be blocking.

4.2 TOPOGRAPHY REASONING

In this section some topography reasoning for blocking and non-blocking graphs are introduced. It is hard to figure out the exact condition or prerequisite to determine whether a graph is blocking or not. But there are some interesting phenomena observed in the scope of topography.

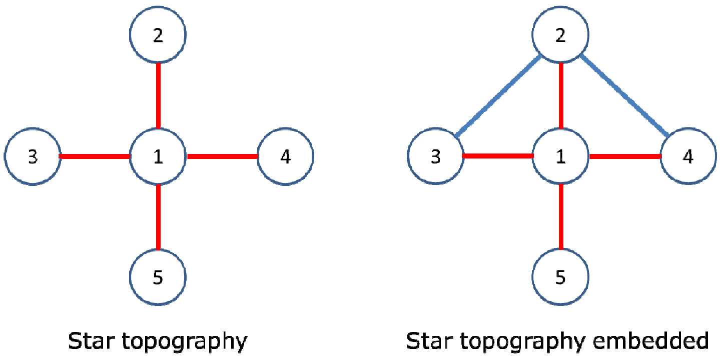
First interesting rule: star topography never blocks. Rigorously speaking, if an n-node graph has a star topography, there exists a node in the graph, which has n-1 edges connected to it. This statement indicates that, if a graph has a star topography, there must be a node which connects all other nodes directly in the graph.



Pic 4.3 Illustration for Star topography

As shown in the picture above, the left graph is star topography since the center node connects all other nodes in the graph. The right graph seems to be a star but not the star topography mentioned in this paper. There is no any node in the graph, which is able to connect all the rest nodes.

Just as stated in section 3.6, a star topology is least connected and non-blocking at the same time. It can be further deduced that if a star topography can be found embedded in a graph, this graph is also non-blocking. It can be illustrated by the following picture.



Pic 4.3 Illustration for star topography embedded graph

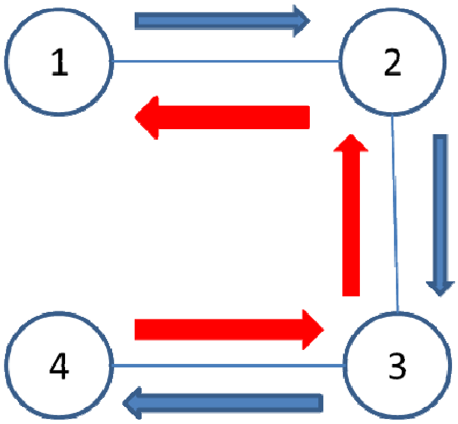
As observed, if a graph is embedded with a star topography, this graph is non-blocking because it has a non-blocking backbone—the star topography. Since star topography does not block, the star topography embedded graphs provide extra connectivity over star topography, and are also non-blocking.

It should be noted that a mere star is not risk tolerating. Any fault in any part of the network would turn the system to blocking. For star topography embedded graphs, as illustrated in the right part of picture 4.3, the blue link provides extra connectivity. If the some of the blue links do not work, the network remains non-blocking.

The inverse statement of this is not true. If a graph does not contain any star topography, it may also be non-blocking. Star topology is a mode which introduces non-blocking property, but it is not the only way.

Second interesting rule: for a graph with more than three edges, chain topology must block. For this chain topography, there are always some traffic patterns that the topography can't handle.

The chain topography scenario is illustrated in the following picture. All the nodes are connected in sequence. It is clear that some deliberately established traffic would be blocked. First let Node 1 and 4 transmitting to each other. This traffic can only be handled as the illustrated by the arrows. Blue arrows indicate traffic from Node 1 to Node 4. Red arrows indicate traffic from Node 4 to Node 1. It is clear that under this scenario all paths to 2 and 3 is blocked. So (14)(23) is one of the traffic patterns that this network can't afford. It is intuitive to see that this phenomenon can be extended to chain topography with more nodes.

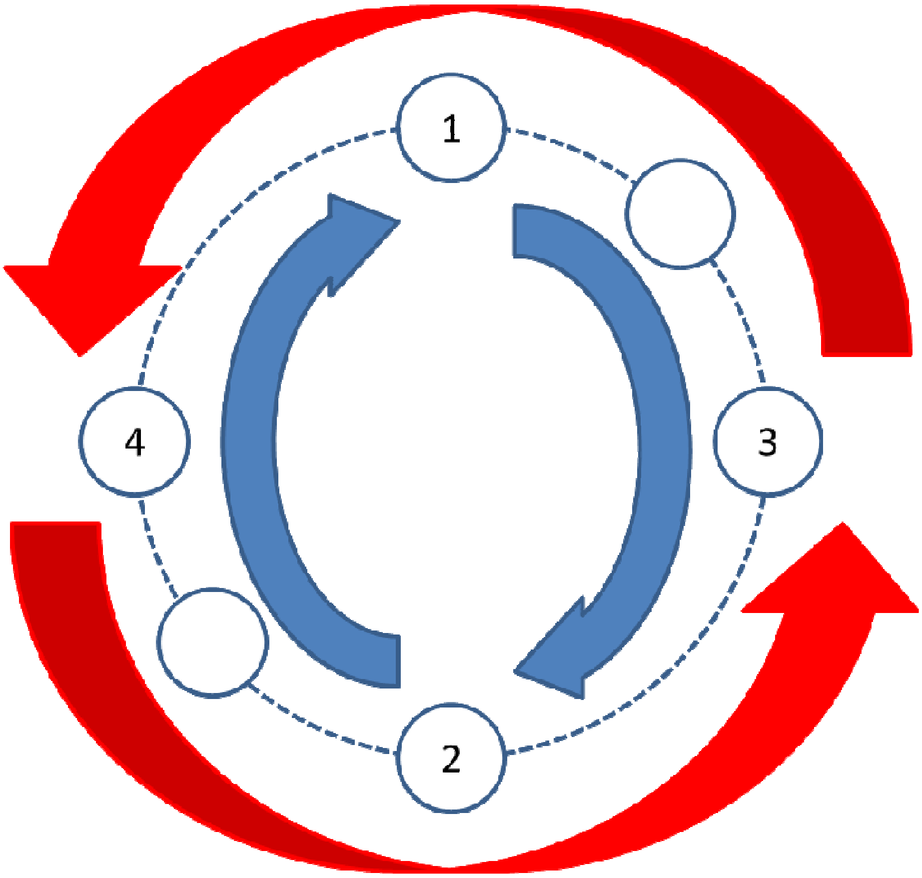


Pic 4.5 Illustration for chain topography

Third interesting rule: for a graph which has more than 4 edges, if it is a ring topography, this graph blocks. This rule is a bit further from the previous rule. The reasoning for this rule is more complicated than the reasoning of the previous, but it is not very hard.

Literally, this rule can be stated in the following way. For a ring topography, it seems that all nodes are connected, but the path does not have any diversity. In this scenario, if a node has to send data to another node which is several nodes away, all intermediate node has no choice but forward the data stream. Since every node is limited connected in a ring graph, if there are enough "trespassing" streams in this ring, some intermediate node would use all its connection forwarding stream for other nodes, remaining itself unable to send out or receive its own data stream.

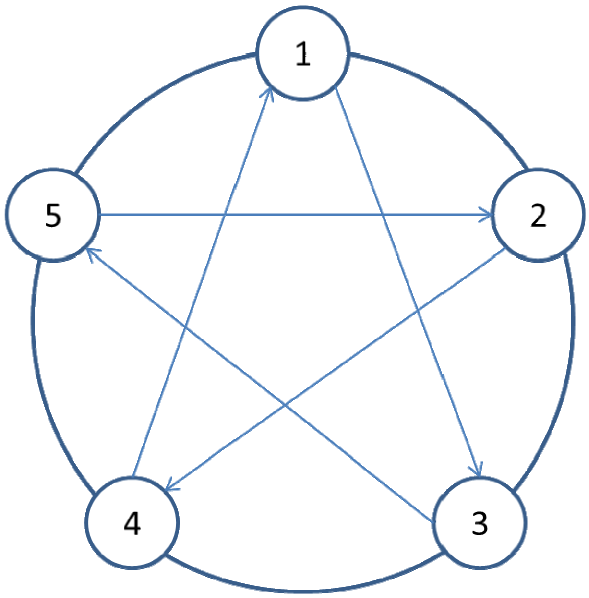
This situation can be expressed in the following picture. The dotted line means that there are nodes connected in sequence in the dotted line part.



Pic 4.6 Illustration for ring topography

In the ring topography above, if a part of the traffic pattern contains (12)(34), the traffic stream of (12)(34) has to be processed by the network as illustrated above. Traffic pattern (12) utilizes in clock-wise path. Traffic pattern (34) utilizes the counter clock-wise path. For these 2 traffics all links in the network have been utilized. And it is easy to observe that the 2 unlabeled nodes on the top right and bottom left of the picture can't communicate with each other. From above reasoning, it can be seen, if the ring contains at least 6 nodes, this ring blocks.

The reasoning for five-node ring topography is a bit further tricky. The five-node ring is presented in the following graph. This graph only fails to support 2 similar traffic patterns (13524) and (14253) in the same way. With similar reasoning above, some part of the traffic takes all the links, and the remaining traffic can't be served. The key point is that the traffic has to trespass some intermediate nodes in order to occupy more links for a single traffic, wasting the resource for the other traffic streams. The following picture illustrates a typical case when the graph is trying to serve traffic pattern (13524).



Pic 4.7 Five-node ring

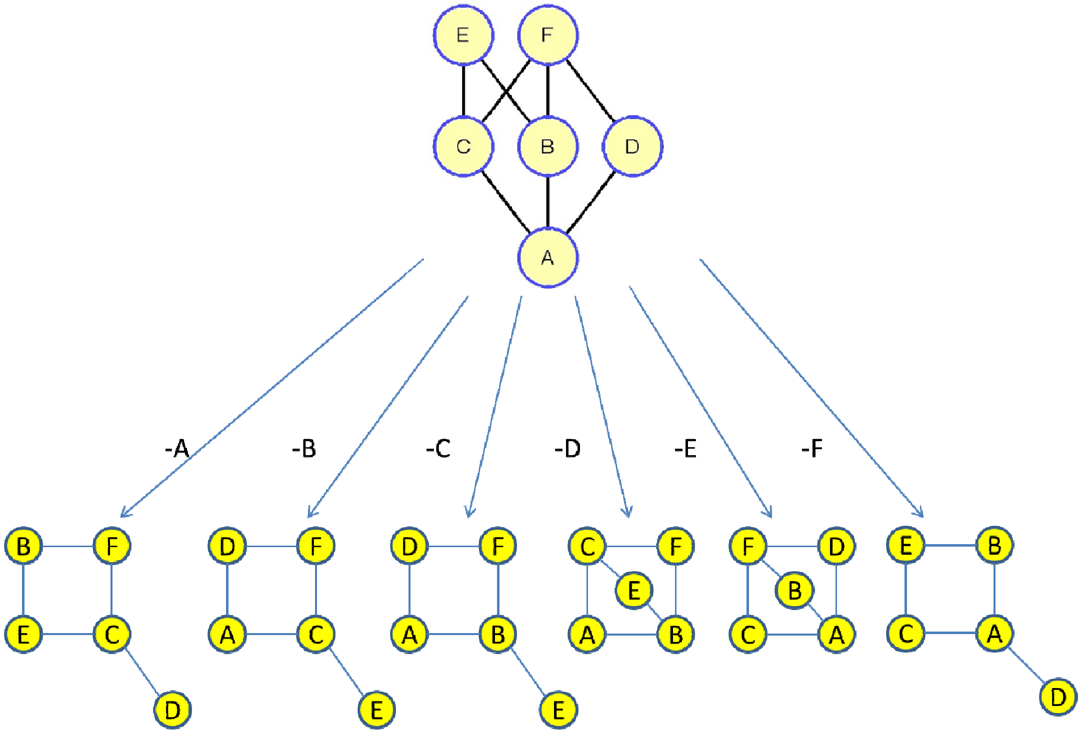
4.3 STAR NON-BLOCKING AND NON-STAR NON-BLOCKING

As mentioned above in section 4.1, if a graph contains star topography, this graph is non-blocking. But star topography introduces concentrated traffic. In a mere star graph, all traffic streams has to pass the center of the star in order to get to its destination. This star relied system may be vulnerable if some part of the core star topography is broken.

It is impractical, for the scope of security, to establish a full connected graph where every node can serve as a center of a star. An n-node fully connected graph can tolerate at least $\left\lfloor \frac{n-1}{2} \right\rfloor$ link failure since with any $\left\lfloor \frac{n-1}{2} \right\rfloor$ links broken there are always some nodes connecting all other nodes. This node can serve as the center of a star and this graph is star topography embedded. A full connected graph is tolerable for any number of node failures. This means that for a fully connected graph, the remaining nodes can keep positive communication behavior no matter how many nodes are broken.

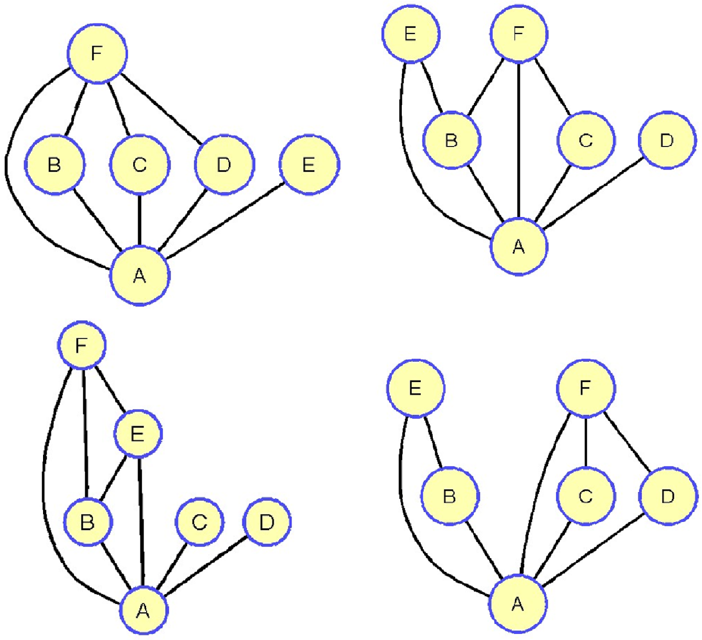
Non-Star non-blocking graph is the graphs that are non-blocking and don't contain any star topography. For an n-node graph, a node can have a maximum degree of n-1, meaning that all other nodes are connected to this node. An n-node non-star non-blocking graph is a non-blocking graph that every node has at most a degree of n-2. It can be inferred that it is hard to find a non-Star non-blocking graph. It can be inferred that, generally speaking a non-Star non-blocking graph is more balanced with a star topography embedded graph given that they have same number of nodes and same number of edges.

The following example vividly illustrates this phenomenon:



Pic 4.8 nSnB graph with node failure resilience

In the top of the picture there is a non-blocking graph with six nodes and eight edges. This graph has six nodes while the maximum node degree for this graph is only 3. Star topography requires the maximum node degree to be 5 for this graph. It can be shown that the remaining graph is unblocking referring Picture 4.3. No matter which single node is removed, the remaining graph is non-blocking.



Pic 4.9 Node failure for star topography

Next step examine all star-topography embedded graphs with six nodes and eight edges, as illustrated in Pic 4.9. There are four different six-node graphs with eight edges. It can be seen obviously that all graph suffer greatly if losing the center node of the star. The remaining graphs of them are even unconnected.

It can be inferred that given limited number of edges, generally speaking, there are some non-Star non-blocking graph with excellent single node failure resilience property than star topology with same number of edges. It should be noted that not every non-Star non-blocking graph is perfect single node failure. Star topology enjoys the fact that, any node failure other than the center node does not affect the remaining network.

CHAPTER 5 CONCLUSION

This chapter concludes this paper. The theory background, methodology, and data analysis are covered in brief in this section. This paper uses a variety of methodology investigating the blocking problem of WDM network without WLI. A WDM network without WLI is an novel idea and novel structure for optical communication. In this network the management of the data stream is a global job other than that of only nodes.

The basis of this problem is built up upon the enumeration of non-isomorphic graphs. The topography of the network determines significant network properties and features in non-WLI WDM network. Non-isomorphism graphs enumeration can be calculated using Burnside's theorem. Burnside's theorem is based on abstract algebra theory and is very complicated. Applying Burnside's theorem in the scenario where the graphs contain large number of nodes in the graph has great computational complexity.

Since networks must be connected. The problem becomes one of enumerating connected graphs. It is hard to define connectivity in mathematical word. It is even harder to apply this definition in the algebra system of Burnside's theorem. The enumeration of connected non-isomorphic graph is implemented on MATLAB. The routing analysis is also written in MATLAB to see whether a graph blocks a particular traffic pattern. This process has very high computational complexity.

Traffic patterns are introduced to provide an approach to analyze the blocking property of a particular graph. It can be inferred that, if a graph is non-blocking over all possible traffic patterns, this graph would provide maximum compatibility over the traffic. Complicated streams and multi-destination traffic can be implemented with time multiplexing or wave length multiplexing. This is a novel concept and has great research value.

After analysis in detail, some basic rules can be observed in the scope of blocking property of graphs. Generally speaking, the star topography is a structure which supports non-blocking property with least demand of number of edges. In this scope, non-blocking graph are divided into two mutually exclusive categories: star topography non-blocking graphs and non-Star topography non-blocking graphs. Security and system robustness can be analyzed in this approach. More work can be done in this field.

This topic has enormous potential. By proper scheduling of the streams, reliable systems can be built up. The streams may also be routed according to wavelengths. A good scheme with a proper graph will contribute to a new optical network system with brand new structure. The next steps may include:

- Figure out some pattern for graphs and routings
- Wavelength routing
- Time variant scheme
- Performance vs Scale
- Robustness analysis(node resilience or edge resilience)
- Robust routing considering failures

WDM network without WLI is an important idea in optical networks, bringing about new challenges in switching and routing. This field is new and has great research value. The progress or improvement in this field will lower the cost of implementing WDM network.

This thesis does NOT completely solves the problem to establish a WDM network without WLI. But the techniques, such as blocking analysis and enumeration method, may optimistically lead to the solution of this problem.

APPENDIX A:

BIBLIOGRAPHY

Bibliography entry. Single-spaced within entries. Usually ‘hanging’ from the second line on, like this.

References:

[1] Richard A Thompson, “Two Conditions under which WDM networks are Rearrangeably non-blocking without WLI Interchangers”, In 2nd Int'l Conference on the Network of the Future, November 28-30, 2011, Paris, France. (Submitted)

[2] Richard A Thompson, Telephone Switching Networks, Artech House, 2000.

[3] others refs to be added