Exploiting Petri Nets to Reduce Switch Crosstalk and Path-Dependent-Loss in Optical Interconnection Networks

Tolgay Karanfiller

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Doctor of Philosophy
in
Applied Mathematics and Computer Science

Eastern Mediterranean University
January 2012
Famagusta, North Cyprus

Approval of the Institute of Graduate Studies and Research

____

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Doctor of Philosophy in Applied Mathematics and Computer Science

____

Prof. Dr. Nazım Mahmudov
Chair, Department of Mathematics

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Doctor of Philosophy in Applied Mathematics and Computer Science

____

Prof. Dr. Rza Bashirov
Supervisor

Examining Committee

1. Prof. Dr. Rza Bashirov ________________________________

2. Prof. Dr. Ufuk Çağlayan ________________________________

3. Asst. Prof. Dr. İbrahim C. Arkut ________________________________

ABSTRACT

Although optical communication systems promise to meet demands of communication networks and multiprocessors in fast communication, they suffer from challenges such as path dependent loss and switch crosstalk. An innovative approach proposed in the present thesis is centered upon modelling OMINs with Petri nets and using P-invariants method for determination of the minimum number of stages m_{min} that is sufficient for realization of demanded communication patterns in an OMIN with variable number of stages. Being composed of m_{min} stages an OMIN of the minimal structure provides the least values for the path dependent loss and switch crosstalk.

Based on complexity results, we make sure about feasibility of our approach. Firstly, we prove that the P/T-nets created in the present research are in polynomial dependence on the problem size, which alleviates memory consumption significantly and reassures the fact that the task according to our approach can be completed in feasible time. Secondly, we compare P/T-nets obtained in the present reasearch with the complete unfoldings created in our previous reasearch and show that P/T-nets in the latter work are more compact in the size than the ones considered in the former research. This is improvement of the complexity results obtained in our previous work.

Finally, we verify validation of our approach through performing series of computer tests and showing that the results of the computer experiments agree with known analytical results.

ÖZ

Optik interkoneksiyon ağları, bilgisayar iletişim ağlarının ve çok işlemcili sistemlerin hızlı iletişime olan ihtiyaçlarını karşılamanın yanı sıra yola bağımlı kayıplar ve anahtarlamada hatların karışması gibi zorluklardan etkilenirler. Bu tezde önerilen yenilikçi yaklaşımın temelinde optik interkoneksiyon ağlarının Petri ağları ile modellenmesi ve P-invariantlar metodunu uygulayarak çok basamaklı optik interkoneksiyon ağlarında verilen permütasyona göre en küçük basamak sayısının bulunması yatar. En küçük basamak sayılı optik interkoneksiyon ağında yola bağımlı kayıplar ve anahtarlamada hatların karışması en küçük düzeyde seyredilir.

Önerilen metodun uygulanabilirliğinden emin olmak için bir karmaşıklık analizi yapılmıştır. Öncelikle tasarlanan P/T-ağların büyüklüğünün artış hızının polinomyal olduğu ispat edilmiştir. Bu olgu, P/T-ağ oluşumunda hafıza tüketiminin önemli ölçüde azaltarak, sıkça rastlanan "durum patlaması"ndan uzakta tutulabileceğini göstermektedir. Sonra takdim edilen çalışmada elde edilen P/T-ağlarını, bir önceki çalışmamızda tasarlanan tamamen açılmış Petri ağlarıyla kıyasladık. Sonuç olarak elde ettiğimiz Petri ağlarının daha küçük olduğu ispat edilmiştir.

Alınan analitik ve kuramsal sonuçların geçerliliği yapılan bilgisayar testleri ile kanıtlanmıştır.

To My Family

and

Sabriye Topal

ACKNOWLEDGMENTS

I would like to thank my supervisor Prof. Dr. Rza Bashirov for his continuous support and guidance in the preparation of this study.

I would like to thank all members of Department of Mathematics for helping me with various issues while I was writing my thesis.

I would like to thank all my family members for their great moral support during my thesis writing. Besides, I would like to thank all my friends who always supported and helped me.

I would like to thank my fiancée for her helping during the study and morale support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

BP	Class of **B**it **P**ermute permutation
BPC	Class of **B**it **P**ermute **C**omplement permutations
CP-net	**C**olored **P**etri net
DC	**D**irectional **C**oupler
DFT	**D**iscrete **F**ourier **T**ransform
HPC	**H**igh **P**erformance **C**omputer
LC	**L**inear Combination **C**omplement permutation
LIN	**LIN**ear combination
MCTN	**M**ultistage **C**ube **T**ype **N**etwork
MIN	**M**ultistage **I**nterconnection **N**etwork
NP-completeness	**N**on **P**olinomial completeness
OMIN	**O**ptical **M**ultistage **I**nterconnection **N**etwork
PFLOPS	**P**eta **FL**oating Point **O**perations **P**er **S**econd (10^{15} FLOPS)
P-invariant	**P**lace invariant
P/T-net	**P**lace **T**ransition net
PE	**P**rocessing **E**lement
SDM	**S**pace **D**ivision **M**ultiplaxing
Tbyte	**T**era byte (10^{12} byte)
T-invariant	**T**ransition invariant
TDM	**T**ime **D**ivision **M**ultiplexing
VLSI	**V**ery **L**arge **S**cale **I**ntegration
WDM	**W**avelength **D**ivision **M**ultiplexing

LIST OF SYMBOLS

m	number of stages
$m_{\min}$	minimum number of stages
2^n	number of inputs/outputs
O	upper bound symbol
$C(k)$	bit complement permutation
$P(r,t)$	bit permute complement permutation
$PC(r,t)$	bit permute complement permutation
L	linear permutation
LC	linear complement permutation
σ	shuffle-exchange permutation
σ^{-1}	unshuffle-exchange permutation
β_i^k	the i th k -ary butterfly permutation
δ_i^k	the i th k -ary baseline permutation
E_k	the k -th cube permutation
$PTNCW_{2\times 2}$	P/T-net model of 2×2 DC
$PTNOMIN_{2^n,m}$	P/T-net model of $2^n \times 2^n$ $k -$stage OMIN

Chapter 1

INTRODUCTION

1.1 Motivation

Tremendous progress in designing HPCs has been achieved over the past decades. Many problems arising in scientific, engineering and industrial domains, however, are still demanding new solutions towards further improvement of the performance. The communication and interconnection but not memory or logic is the major factor that limits the performance of HPCs. While the computational capability of modern computing systems reaches PFLOPS limit and the memory capacity pushes Tbyte border, the communication is limited by speed of electronic circuits. There is nowadays a great demand for fast data communication which cannot be handled by electrical communication networks.

Optical interconnection networks have become an appealing candidate to meet ever increasing demands for fast communication in high-performance computing. Low latency, high throughput and high bandwidth are among other advantages of optical interconnection networks. Optical communication technologies suffer from numerous challenges despite a wide spectrum of advantages provided by these technologies. For instance, optical signals lose strength and become weak after passing through a long optical link. This unwanted effect, which is known as pass dependent loss or attenuation, causes signal distortion [18, 23, 31]. Pass dependent loss consists of several components. An essential part of path dependent loss increases with increase of number of components in optical interconnection network [13, 21]. Path dependent loss leads to increase of power consumption in the system and errors in transmission of optical signals.

Switch crosstalk is another challenging problem of optical communication. Switch crosstalk is defined as the crosstalk from one optical channel to the other [15, 30]. Switch crosstalk affects clarity of the optical signals, limits the size of optical interconnection network and leads to error rate degradation.

Path dependent loss and switch crosstalk are in focus of the researchers. Reducing path dependent loss and switch crosstalk is of theoretical and practical interest in optical communication.

1.2 Related Work

In the past decade, much research has been conducted in order to develop directional couplers with reduced path dependent loss [30]. Most of this research is based on implementation of up-to-date achievements in optical technology to reduce the path dependent loss. In [8] the authors suggest a method for reducing path dependent loss through minimizing the number of stages in an OMIN. The main result obtained in [8] is a necessary and sufficient condition for admissibility of special BPC permutations to an $m-$stage $2^n \times 2^n$ OMIN employing the shuffle-exchange interstage communication pattern for $n < m \leq 2n-1$. When $1 \leq m \leq 2n-1$, the minimum number of stages required to pass a permutation through shuffle-exchange OMIN can be determined in $O(2^n n)$ time. For a BPC permutation this can be found in $O(2^n)$ time. The minimum number of stages needed to perform a permutation can be determined in $O(2^n n \log n)$ time. An $O(2^n n)$ algorithm that determines whether a permutation is admissible to $2^n \times 2^n$ MCTNs was introduced in [28]. In [27] this result was extended to $k-$extra stage MCTN with $k=1$. In the same paper it was shown that a permutation is admissible to a $k-$extra stage MCTN if, and only if, the conflict graph is $2k-$colorable. NP-completeness of the 2^k-coloring problem in graphs, for $k>1$, does not allow us to develop a general method for analysis of the permutation admissibility with polynomial dependence on the number of extra stages. Although there exist efficient algorithms for checking the permutation admissibility for $k=0$ and $k=1$, no such algorithm is known for $k>1$ [8]. The admissibility of frequently used permutations that belong to BP, BPC, LIN and LC classes was investigated in [24, 27, 29].

We distinguish between three approaches for reducing switch crosstalk in optical interconnection networks, namely SDM, TDM and WDM approaches. Considerable effort has been devoted to all three approaches in the literature. All three approaches unfortunately suffer from numerous drawbacks [7, 23, 30, 33]. For example, SDM approach requires doubling the original OMIN hardware to achieve the same permutation capability. SDM approach is far to be resource saving approach. In TDM approach, a permutation is generated in n passes instead of single pass. TDM approach is not time effective approach. Finally, the role of WDM in switching, with or without wavelength conversion, is not very clear and requires more careful study.

In [2] permutation capability of MINs has been analyzed through reducing this problem to marking reachability in CP-nets and performing model checking in CP-nets. It was also shown that [5] CP-nets can be used to check the permutation admissibility through creating a CP-net model, then unfolding related CP-net into equivalent optimized P/T-net and finally using T-invariants to decide on permutation admissibility.

1.3 Contributions

The research conducted in the scope of this thesis differs from existing ones in two respects. Firstly, the existing approaches are based on use of achievements in photonic industry to deal with path dependent loss and switch crosstalk. Traditional approaches reduce path dependent loss and the number of switch crosstalks by implementing technologically more mature directional couplers. In this thesis this is done through proposing an optimal OMIN architecture employing minimum number of directional couplers. The main idea behind the present research is to reduce path dependent loss and the number of switch crosstalks by keeping the number of directional couplers as small as possible.

Secondly, Petri nets have never been used in the context of photonic switching. The present research utilizes Petri nets in the context of photoinic swtiching to the benefit of both fields. It is done by checking a permutation for admissibility to 1- through $m-$stage $(1 \leq m < 2n-1)$ OMIN and consequently finding the minimal number of stages that is needed for establishing a desired communication pattern. The essential phases of our approach includes creating a P/T-net model directly from OMIN specifications and performing reachability analysis with P-invariants. It should also be noticed that avoidance of CP-net modelling, unfolding and optimization phases in the present research leads to the refinement of memory and run time constraints compared to those obtained in [5].

This work copes not only with path dependent loss but also with switch crosstalk in OMINs. The total switch crosstalk in any OMIN is the sum of switch crosstalks in all directional couplers. In order to minimize the total path dependent loss and/or the total number of switch crosstalks in an OMIN, we determine the minimal number of stages $m_{\min}$, and consequently the minimal number of directional couplers that is sufficient to generate a requested communication pattern. Once $m_{\min}$ is known, a requested communication pattern can be established in the given OMIN with minimal path dependent loss and a minimal number of switch crosstalks. In the present research this is achieved by representing each communication pattern as a permutation of OMIN's inputs into its outputs, and checking 1- through $(2N-1)-$ stage OMINs employing the same network topology for permutation admissability.

Chapter 2

OPTICAL INTERCONNECTION NETWORKS

Almost 35 years passed since Kao and Hockham in their pioneering work [14] proposed the idea of using light in place of electrons in data communication. Almost half that time ago, Standard Telecommunications Laboratories and British Post Office Laboratories implemented first fibre system in telephone traffic [19]. Since then exceptional breakthrough has been achieved in optical communication. Particularly, OMINs have become an integral part of HPCs. In what follows, we give background in switching theory, and detail the main achievements in optical communication as much as it is required to explain further material.

2.1 Background in Switching Theory

Modern HPCs employ many independently functioning PEs. Although PEs operate independently they need to communicate through interconnection network to get ready for next computational step. Concerning HPC architecture it is essential to be capable to establish simultaneous communication links between N inputs and M outputs. Usually these inputs and outputs belong either to PEs or the memory modules. An $N \times M$ crossbar network is ideal interconnect for establishing simultaneous links between N inputs and M outputs in small-scale HPCs. In crossbar network a new link between idle input/output pair can always be established at any time.

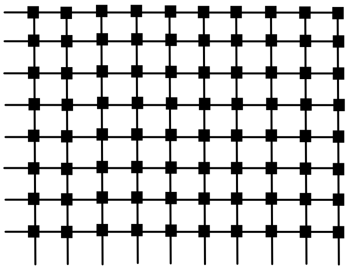


Figure 2.1: An example of 8x10 switching matrix.

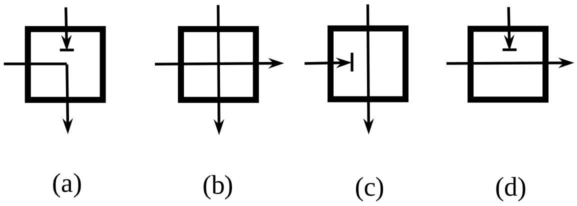


Figure 2.2: Four possibilities for input/output connections.

An $N \times M$ crossbar network can be easily implemented as an $N \times M$ switching matrix. An $N \times M$ **s**withing matrix consists of horizontal and vertical "wires" with $N \times M$ switch points placed at intersections of horizontal and vertical "wires". Schematical description of 8×10 switching matrix is ilustrated in Figure 1. Switching pole or open/close switch is a kind of light switch or reed relay that is dedicated to change the direction of connection. In a switching matrix input i is granted access to output j if the related switch pole is closed. A switching point can be set to one of the following states: (a) left input is granted bottom output (or left-to-bottom connection is allowed) and upper input requesting the same output is blocked (or top-to-bottom connection is forbidden); (b) left input is granted right output and upper input is granted lower output; (c) upper input is granted bottom output and left input requesting right output is blocked; (d) left input is granted right output and upper input is requesting bottom output is blocked. All four configuration for a switch point are shown in Figure 2.2.

Crossbar network is an ideal interconnect in sense that it allows to establish simultaneous links between input/output pairs in a single communication step. Unfortunately, crossbar networks suffer from hardware constraints such as number of pins and number of wires dramatically increase with increase of number of its inputs and outputs. Apart from this, crossbar network is prohibitively expensive. Crossbar networks are practically implemented to connect 2^4 or 2^5 input/output pairs in small-size HPCs.

Over 6-7 decades ago the network designers had moved to conceptually new network architecture which was later practically implemented in many HPC projects. In order to avoid challenges caused by size of crossbar network, the experts had suggested to organize independent small-sized crossbar switches into a cascade of stages. Such an MIN should be capable to simulate the actions performed by large crossbar switch. Under what circumstances multiple stages of small-sized independent crossbar switches can fully simulate the functionality of a large crossbar switch was posed as a question and remained so for long time. Reasonable answer to the question is sufficient conditions on rearrangeability of MINs. First results on rearrangeability of MINs were obtained and published at the end of 50s and in the beginning of 60s [6]. Many researchers have tried to contribute to the solution of this problem. It is nowadays known that $m^n \times m^n$ $(2n-1)-$ stage interconnection network made of independent $m \times m$ crossbar switches and employing regular interstage communication pattern is capable to perform any one-to-one mapping of MIN's inputs into its outputs [1], meaning that a MIN with regular interstage wiring between neighboring sages is functionally equivalent to $m^n \times m^n$ crossbar switch.

Let $x, y \in X$ where $X = \{0, \ldots, N-1\}$, $N = 2^n$, $x = x_n x_{n-1} \cdots x_1$ and $x = y_n y_{n-1} \cdots y_1$ in binary with $x_i, y_i \in \{0,1\}$ for $1 \le i \le n$. A permutation π is a bijection from X to X. Although expressed in a different form, the following definitions are equivalent to the ones given in many papers.

Definition 1. Permutation $C(k)$ is said to be bit complement permutation if $C(k): x_n \cdots x_k \cdots x_1 \to x_n \cdots \bar{x}_k \cdots x_1$ for $1 \le k \le n$. All bit complement permutations form a class denoted as BC.

Definition 2. Permutation $P(r,t)$ is said to be bit permute permutation if $P(r,t): x_n \cdots x_r \cdots x_t \cdots x_1 \to x_n \cdots x_t \cdots x_r \cdots x_1$ for $1 \le k \le n$. The class of all bit permute permutations is denoted by BP.

Definition 3. Permutation $PC(r,t)$ is said to be bit permute complement permutation if $P(r,t): x_n \cdots x_r \cdots x_t \cdots x_1 \to x_n \cdots x_t \cdots \bar{x}_r \cdots x_1$ for $1 \le k \le n$. The class of all bit permute complement permutations is denoted by BPC.

Definition 4. Permutation $L(x_n \cdots x_1) = y_n \cdots y_1$ is said to be linear permutation if $y_i = \sum_{j=1}^{n} \lambda_{ij} x_j$ for $1 \le i \le n$ and nonsingular (or invertible) binary matrix (λ_{ij}) where the addition and multiplication operations are of those modulo 2 arithmetic. The class of all linear permutations is denoted by LIN.

Definition 5. Permutation $LC(x_n \cdots x_1) = y_n \cdots y_1$ is said to be linear complement permutation if $y_i = \sum_{j=1}^{n} \lambda_{ij} x_j \oplus 1$ for $1 \le i \le n$, where for $\oplus$ is modulo 2 bitwise addition operation. The class of all linear complement permutations is denoted by LC.

Definition 6. Shuffle permutation σ is a left circular shift permutation such that $\sigma(x_n \cdots x_1) = x_{n-1} \cdots x_1 x_n$.

Definition 7. Unshuffle permutation σ^{-1} is a right circular shift permutation such that $\sigma^{-1}(x_n \cdots x_1) = x_1 x_n \cdots x_2$.

Definition 8. The ith k-ary butterfly permutation β_i^k, $1 \le k \le n$, is defined as $\beta_i^k(x_n \cdots x_i \cdots x_1) = x_n \cdots x_1 \cdots x_i$.

Definition 9. The ith k-ary baseline permutation δ_i^k, $1 \le k \le n$, defined as $\delta_i^k(x_n \cdots x_{i-1} x_i x_{i+1} \cdots x_1) = x_n \cdots x_{i-1} x_1 x_i x_{i+1} \cdots x_2$ shifts the $i+1$ least significant digits in the index to the right for one position.

Definition 10. The kth cube permutation is defined as $E_k = C(k)$ for $1 \le k \le n$.

Definition 11. A MIN is unique path network if it employs a single path between each input/output pair.

Definition 12. A MIN is full access network if it employs a path between each input/output pair.

Definition 13. A MIN is Banyan network if it is full access and unique path network.

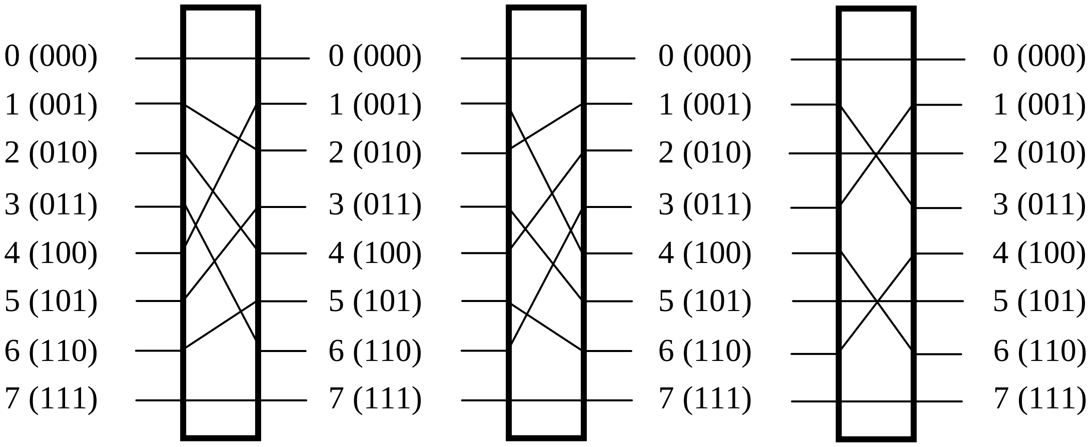


Figure 2.3: σ, σ^{-1} and bit reversal arranged from left to right for $N = 8$.

Let us turn attention to $2^n \times 2^n$ n-stage banyan networks with regular interstage pattern between neighbouring stages. These MINs have been on focus of researches for many years. In baseline network, interstage pattern between ith and $(i+1)$st stages is defined by baseline permutation δ^i_{n-k} for $1 \le i \le n$. Meantime, interstage pattern betweeen the first two stages is shuffle permutation σ. Butterfly network employs butterfly permutation β^k_i between stages i and $i+1$ where $1 \le i \le n$. In indirect cube network, a wiring between stages i and $i+1$ is determined by butterfly permutation β^k_{n-i}, $1 \le i \le n$. Finally, omega network uses shuffle permutation σ to link the neighbouring stages. All these $2^n \times 2^n$ n-stage banyan networks are commonly called Delta networks or MCTNs [10].

Banyan network is a resource saving network. This is why banyan networks are important compared to non-banyan full-access networks. Typical banyan network is composed of $2^{n-1}n$ 2×2 switches and $2^n(n-1)$ interstage wires. A banyan network is capable to perform $2^{2^{n-1}n}$ distinct one-to-one mappings of the network's inputs into its outputs.

Omega, baseline, butterfly, indirect cube and related inverse networks are banyan networks that are oftenly referred in the lierature. Omega is an banyan network with permutation σ as an connection pattern between neighboring stages (see Figure 2.3). In butterfly network, connection pattern between stages $i-1$ and i is defined by permutation β_i^k for $1 \le i \le n$ (see Figure 2.4). Baseline network employs permutation δ_i^k between stages $i-1$ and i, for $1 \le i \le n$ (see Figure 2.5). In indirect cube network, wiring between stages $i-1$ and i is selected to be permutation β_{n-i}^k for $1 \le i \le n$ (see Figure 2.6). Inverse of a MIN is a mirror reflection of the given MIN.

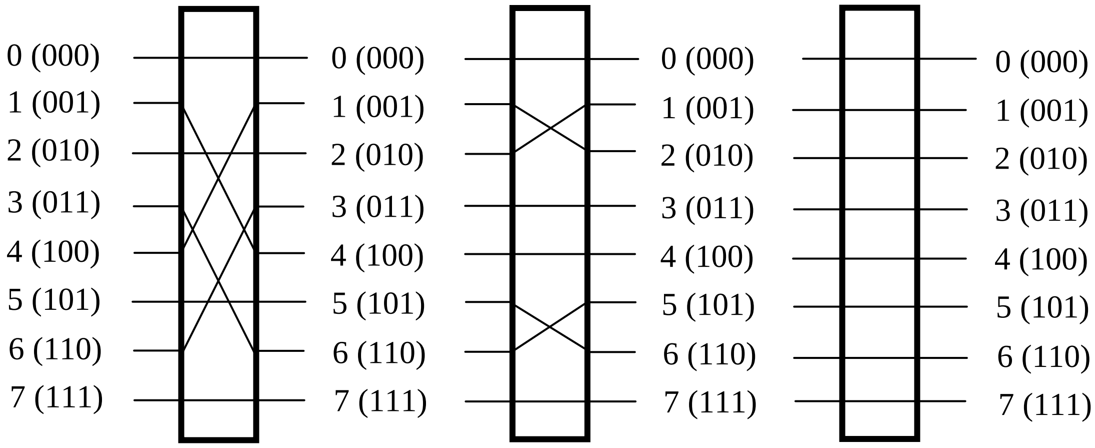


Figure 2.4: 2nd, 1st and 0th butterfly arranged from left to right for $N = 8$.

It was shown that omega, baseline, butterfly and indirect cube networks are topologically equivalent. This particlarly means that all of above MINs have the same functional capability. In the literature, these MINs are referred to as MCTNs. The number of ways in which network's inputs can be connected into its outputs is known as network's permutation capability or combintorial power. Any banyan network is capable to connect N input/output pairs in $2^{2^{n-1}n}$ ways, though there exist $N!$ distinct ways for permuting N inputs into N outputs. Modern HPCs demand MINs with even higher combinatorial power.

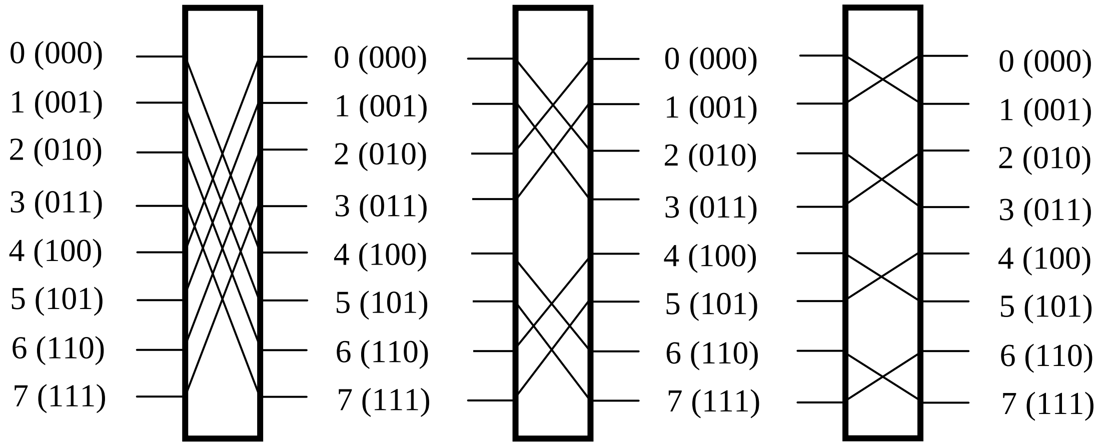


Figure 2.5: 2nd , 1st and 0th cube arranged from left to right for $N = 8$.

According to permutation capability MINs are classified into three classes: blocking, nonblocking and rearrangeable MINs. A MIN is said to be blocking if there exists at least one permutation not realized by the network. Banyan network is a classic example to blocking network. For example, identity permutation cannot be realized in banyan network. Being capable to realize only limited subset of the set of all permutations, blocking networks have restricted permutation capacity. A blocking network however has some attractive characteristics. A blocking network is usually composed of minimal number of switches and channels. Because of compact structure a blocking network is also a low cost network. Blocking networks are integral part of architecture of special purpose computers. Network topology for blocking network designed for special purpose computer depends on the class of dedicated algorithms that the computer performs. For instance, omega network with shuffle interstage pattern between the neighboring stages suits best to the architecture of a DFT special purpose computer.

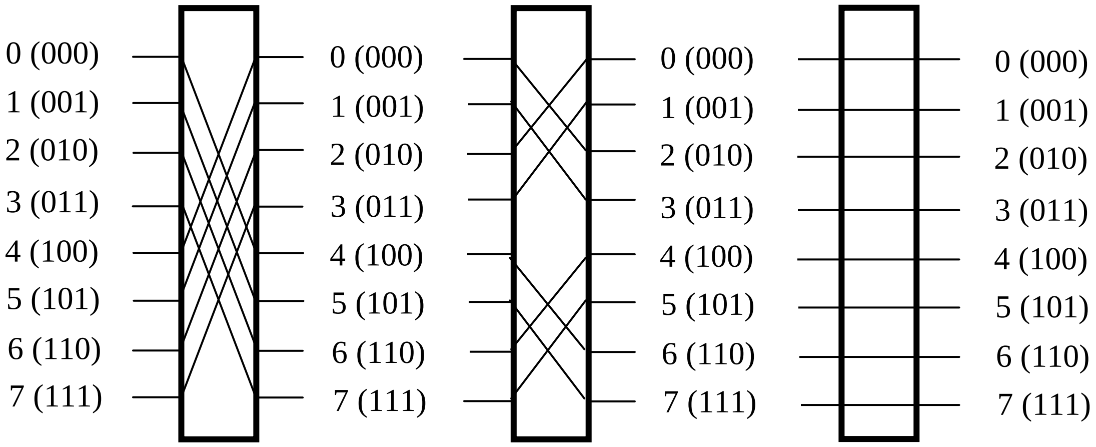


Figure 2.6: 2nd , 1st and 0th baseline arranged from left to right for $N = 8$.

A MIN is said to be nonblocking if it provides simultaneous links between all input/ output pairs in the network without affecting already established links. Nonblocking MINs usually employ multiple paths between input/output pairs. Close network is a classic example to nonblocking networks. The design principle of a Close network is explained using the following example. In a 3-stage Close network first, second and third stages are respectively made of n copies of $n \times m$ switches, m copies of $n \times n$ crossbar switches, and n copies of $m \times n$ switches. Close network is capable to perform all $n^2 \times n^2$ mappings of network's inputs into its outputs. An example of 3-stage Close network with $n = 3$ and $m = 4$ is illustrated in Figure 2.7.

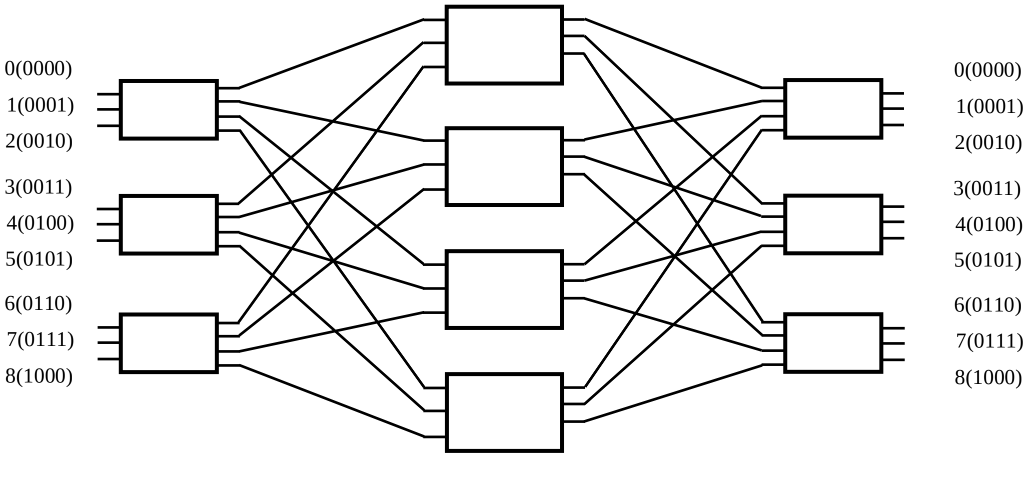


Figure 2.7: An example of Close network with $n = 3$ and $m = 4$.

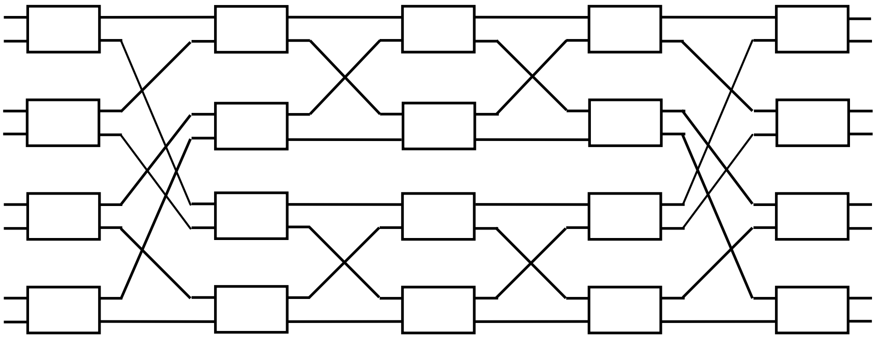


Figure 2.8: Benes network for $N = 8$.

In computing the cost, we observe that crossbar switch requires N^2 ($N = n^2$) switches while a banyan network requires $n^2 m$ switches in each of the first, second and third stages. Thus, total cost of a banyan network is $n^2 m$, which is essentially less compared to that of crossbar switch. Although nonblocking networks are cheaper in the cost than crossbar switch they are not cost-effective compared to rearrangeable MINs. In rearrangeable MIN N simultaneous paths between distinct input/output pairs can be established though it may require rearranging the existing paths. A rearrangeable MIN also employs multiple paths between input/output pairs but it is cheaper in the cost than nonblocking MINs. The best known rearrangeable network is Benes network, which is obtained by merging baseline network with its inverse such that the last stage of baseline overlapping with the first stage of inverse baseline. Figure 2.8 shows 8×8 Benes network.

2.2 Optical Interconnection Networks

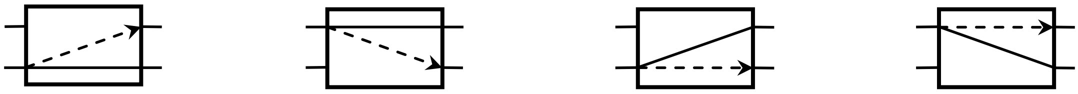
The debates between researchers about relative merits of optics and electronics in data transsmission and data processing have lead to reasonably clear consensus that electronics is the best choice for data processing meanwhile optics provides solutions towards fast communication. Over the past several decades spectacular progress has been made in design of optical communication devices. At the time when optical technologies are becoming the driving force behind the fast communication it is the question of interest whether optical communication technologies can fully substitute their electronic counterparts. It is also necessary to understand new challenges that the optical communcaiton technologies bring to the concern of the practioners.

At the present time purely optical devices create numerous problems such as they are either too power-consuming, too slow compared to the electronic counterparts, or both. These are particuarly the reasons why it is hard if not impossible to conduct optical communication using purely optical devices. The situation with, so called, all-optical communication will remain seemingly same in the foreseeable future. The best solution can most probably be achieved by hybrid optical communication, where optical and electronic components are reasonably comprimised within the same optical communication pattern. In hybrid optical communication the signals are optical but the control over the optical signals is carried out electronically.

DCs are broadly used in design of the OMINs. A single DC consists of a pair of waveguides. A large OMIN is made of many DCs arranged into stages where the stages are interconnected through interstage patterns. Each pattern is a collection of waveguides linking input-output pairs of neighbouring stages.

Due to physical properties of optics, the problems encountered in design of OMINs differ from those known for MINs. Some factors directly affect optical signal-transfer performance. Optical crosstalk and pass dependent loss are prime factors that affect optical signal-transfer performance. An OMIN comprises multiplicity of optical channels that oftenly cross each other causing optical crosstalk. Optical crosstalk in an OMIN arises in the form of channel crossover or switch crosstalk. Although it is required that the light guiding channels remain seperate and distinct, an OMIN experiences channel crossover when two light guiding channels existing in a common single crystal platelet cross each other to obtain a particular network topology. This unwanted effect is known as channel crossover. The number of channel crossovers essentially depends on the network topology. It is known that the number of channel crossovers in an OMIN can be reduced through selecting suitable planar network topology [17]. It is also known that the channel crossover depends on the intersection angle between the crossing optical channels. Is is possible to reduce channel crossover by changing the intersection angle between the optical channels [22].

It is however hard to overcome switch crosstalk [30, 15]. Switch crosstalk arises when two paths sharing same DC experience unwanted coupling from one path to another. The four possible configurations for switch crosstalk are depicted in Fig. 2.9.



(a) switch crosstalk caused in straight state

(b) switch crosstalk caused in straight state

(c) switch crosstalk caused in cross state

(d)switch crosstalk caused in cross state

Figure 2.9: Four possibilities for switch crosstalk.

In this figure, a path that switch crosstalk propogates through is indicated with dash line. One can easily seen that switch crosstalk in the straight state is directed either from the lower input to the upper output or from the upper input to the lower output. Likewise, there are two possibilities for a switch crosstalk in a cross state. It is either between the lower input and the lower output or the upper input and the upper output.

Many researchers investigated switch crosstalk and the ways to reduce switch crosstalk [7, 11, 12, 22, 23, 28, 31, 32, 34]. All these works are based on implementation of novel network architectures such as dilated Benes or Banyan networks or multi-plane Banyan architecture to reduce switch crosstalk. The main idea behind of approaches is to use the three multiplexing principles: WDM, TDM and SDM. It must be noticed WDM, TDM and SDM multiplexing techniques have been widely investigated. Despite advantages all three multiplexing techniques have some disadvantages. For instance, it is broadly known that SDM is not cost-effective in sense that it requires doubling of the DCs and waveguides just to achieve the same permutation capability. In TDM approach a permutation is realized in n passes through the network, although OMIN generates a permutation in one pass. This is a drawback that affects time-effectiveness of TDM approach. Finally, the role of WDM (with switching or without switching) in switching is still unclear and is a matter of extensive investigations.

In order to keep switch crosstalk as small as possible the OMINs are usually designed using DCs with 2 inputs and 2 outputs. Reducing the switch capacity leads to increase of the number of stages as well as number of DCs in a stage. Increase in the number of DCs consequently leads to increase of path dependent loss, which is another challenging problem that the OMIN designers try to cope with.

Optical signals usually become weak when passing through long optical path causing signal distortion or attenuation. This phenomenon is known as path dependent loss.

Signal attenuation is a drawback that results in error in transmission of optical signals. It is customary to increase the power consumption in order to avoid the signal attenuation.

Because of variety of loss contributions, total loss in a switch fabric can be accurately evaluated only based on experimental results. Main contributions to the total loss in any optical link come from several sources: pass dependent loss that arises in the form of propagation loss through the waveguide in a DC; signal loss in the medium; signal loss at waveguide bends; signal loss at waveguide crossovers; fiber-to-substrate and substrate-to-fiber coupling loss. Being more severe pass dependent loss essentially contributes to the total loss. In an OMIN, path dependent loss naturally increases with increase of the path length and consequently the number of DCs on that path. Pass dependent loss is proportional to the length of the path and consequently to the number of the DCs that the optical signal passes through. Thus, pass dependent loss essentially depends on the path that the optical signal passes through: different routing might result in different pass dependent loss. Insertion loss L is the worst-case loss evaluated on the longest path between input output pairs.

The insertion loss difference Δ is another parameter used to evaluate the total loss in an optical switch. The differential loss is determined as the differential attenuation between the most and least loss paths in an optical switch [25].

Chapter 3

PETRI NETS

Petri nets are useful graphical and mathematical modeling tool. Petri nets are frequently used for modeling and simulation of dynamic systems which are concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic. Petri nets are attractive because there exist dedicated software tools Petri net visualization tools models are visually represented software tool that is could be used as visual-communication and visual-communication assist the same with block diagrams, flow charts and networks. Moreover, to simulate the dynamic and simultaneous activities of systems tokens are used. On the other hand, as a mathematical tool, it is practicable to construct mathematical models, algebraic equations or static equations for arranging the attitude of systems. Both practitioners and theoreticians are used Petri nets. Consequently, Petri nets supply strong communication between practitioners to make their model more methodical. At the same time, practitioners can teach theoreticians to make their models more realistic.

Starting by 1960 plenty of papers have been published. The oldest dissertation submitted by Carl Adam Petri in 1962. Design and analysis of Petri Nets is based on precise and accurate mathematical theory. Because of the enhanced software tools Petri Nets are used modeling and an analysis of particular applications with strong mechanism.

The theory of Petri Nets satisfies a precise the theoretical mechanism for modeling of the discrete event systems and analysis of their characteristics. In order to solve, the problems which are arising in scientific, engineering and industrial domains for learning behaviors, Petri Nets could use successfully. Usage of Petri Nets scientifically increased in recent years. Discrete and dynamic systems could be implement successfully in different areas such as computers and their components, concurrent processes, serial and parallel processing, computer networks, computer programs, operating systems and discrete industrial systems such as factory pipelines. In order to examining the behavior of discrete systems, several mathematical models and methods have been used also Petri Nets represent such a system. It is important to implement Petri Nets because it can be surely understood whether or not these systems efficiently work.

In this section, some application areas considered in the literature will be given. Petri nets have been proposed for a several type of application. It is because of the generality and allowance inherent in Petri nets. Petri nets can be applied any area or system which are described graphically such as flow charts and that needs some means of representing parallel or concurrent activities informally. Despite the fact that explain, tradeoff between modeling generality and analysis capability have to consider. It means more general model with less responsible to analysis. Complexity problem is a lack of Petri nets. Petri nets based model transformed into a large form of analysis, even if for medium size systems. In order to apply Petri nets, it is important to add appropriate restrictions. The application of Petri nets can be classified into three parts. Firstly performance evaluation and communication protocols are two areas which are applied successfully. Modeling and analysis of distributed software systems, distributed database systems, concurrent and parallel program, flexible manufacturing/industrial control system, discrete-event system, multiprocessor memory systems, dataflow computing systems, fault-tolerant systems, programmable logic and VLSI arrays, asynchronous circuits and structures, compiler and operating systems, office information systems, formal languages, and logic programs are the examples for predicted applications areas. Finally there are some interesting areas applicable in the literature such as local-area networks, legal systems, human factors, neural networks, digital filters, and decision models.

3.1 Formal Definition

The rule for firing enabling and transaction can be seen very easy but it is implementation in Petri Net theory is very hard and complex. Directed graph can given as a part of Petri Nets. It is initial state called the initial marking and denoted by M_0. The underlying graph of Petri Net *N* and it is a directed, bipartite and weighted. It has places and transitions nodes which their arcs transition from one to another. In graphical formulation circles refer to places and bars (or boxes) refer to transitions. In addition to these arcs are labeled with positive integers as their weight and k -weighted arc can be interpreted as the set of *k* parallel arcs. Labels for unity weight are usually omitted. M_0 which is referring to marking state assign as nonnegative integer to each place. If M_0 is assigned as nonnegative integer *k*, at place *p*, then *p* is marked with *k* tokens. Graphically, *k* black tokens in place *p*. As it mentioned before marking is referred to *M*, an m -vector, where *m* is the total number of places. In addition to that the p^{th} component of *M*, is shown as $M(p)$ is the number of tokens in place p . In the modeling places stand for conditions and transitions stand for events. In order to interpret the preconditions and post conditions of an event, there are exact numbers of input and output places that transition has. There are several examples for explaining the transitions and places such as, if the input place is given as preconditions and transitions is an event then output places will be post conditions.

Definition 14. A Petri net is a 5-tuple $PN =< P, T, A, W, M_0 >$where:

$P = \{p_1, p_2, ..., p_m\}$ is a finite set of places,

$T = \{t_1, t_2, ..., t_n\}$ is a finite set of transitions,

$A \subseteq (P \times T) \cup (T \times P)$ is the set of arcs (flow relation),

$W : A \to \{1, 2, 3, ...\}$ is a weight function,

$M_0 : P \to \{1, 2, 3, ...\}$ is the initial marking,

$P \cap T = \emptyset$ and $T \cap P = \emptyset$.

A Petri net structure N=<P,T,A,W> without any specific initial marking is denoted by N . A Petri net with the initial marking is denoted by $PN =< P,T,A,W,M >$. The comportment of many systems can be visualized inwards system states and their changes. A state in Petri Nets is changed based on the transition rule, for simulating the dynamic behavior of a system.

A transition *t* is said to be enabled if each input place *p* of *t* is marked with at least *w*(*p*,*t*) tokens, where *w*(*p*,*t*) is the weight of the arc from *p* to *t*. An enabled transition may or may not fire (depending on whether or not the event actually takes place).

Source transition is a transition without any input place. Besides transaction without any output place is called *sink* transition.

Definition 15. Self-loop consist of a place p and transition t if and only if p is input and output place of t at the same time. Pure Petri net has no self loops. Moreover, ordinary Petri net means, all arc weighted as 1.

Example 1. An illustration of transition firing rule:

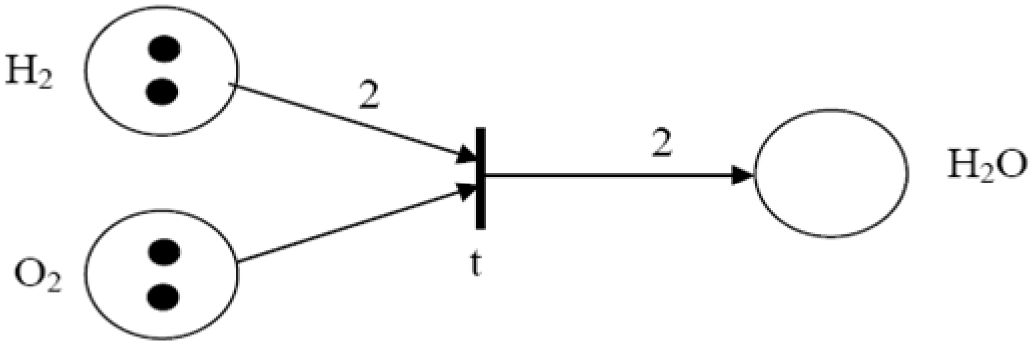


Figure 3.1: Transition firing rule for marking before firing the enabled transition *t*.

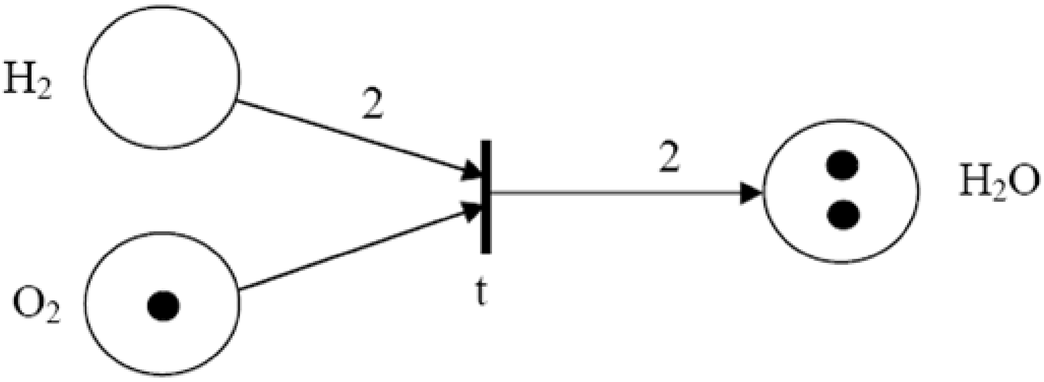


Figure 3.2: Transition firing rule for marking after firing *t*, where *t* is disabled.

3.2 Properties

There are two types of properties that are going to be discussed if Petri Net depended to M_0 then it is stated as behavioral properties. However, if it is independent M_0 and dependent only the structure of Petri Net it is called structural properties.

3.2.1 Behavioral Properties of Petri Nets

The reachability, boundedness, safety, limitation, liveness, deadlock, reversibility, home state, coverability, persistence and fairness are oftenly referred behavioral properties.

3.2.1.1 Reachability

In reachability problem it is important that to notice if given marking M_1 can be reached from the initial marking M_0. In an other word if an undesirable marking M_2 can be avoided.

Definition 16. A finite occurrence sequence σ is a sequence of markings and steps:

$$\sigma = M_0[S_{\ 1} \succ M_1[S_{\ 2} \succ \cdots M_{r-1}[S_r \succ M_r$$

such that $r \in N$, where N is the set of natural numbers.

M_r is reachable from M_0 in r steps. Analogously, an infinite occurrence sequence is a sequence of markings and steps

$$M_0 \succ [S_1 \succ M_1[S_2 \succ \cdots.$$

Definition 17. The marking M_r is said to be reachable from M_0 if there exists a firing sequence σ of transitions that will yield M_r, i.e. $M_0 \overset{\sigma}{\rightarrow} M_r$, denoted $M_0[\sigma \succ M_r$.

The set of all possible markings reachable from M_0 is denoted by $R(M_0)$.

3.2.1.2 Boundedness and Safety

It is necessary to know whether Petri net model is bounded. In order to define the size of the bounded property is useful.

Definition 18. A Petri net is k **-**bounded with respect to an initial marking M_0 if the number of tokens in any of its places never exceeds k for any marking in the reachability set , $R(M_0)$ i.e. $M(p) \leq k$, $\forall p \in P$ and $\forall M \in R(M_0)$, where is the number of tokens in place p in marking M.

Definition 19. A Petri net is said to be bounded (or limited) if it is k-bounded for $\exists k \in \aleph$.

Definition 20. A Petri net is safe if it is k-bounded and k = 1.

Safety is a important part of boundedness. 1-boundedness are safe. The notion of safe and bounded nets is independent of the transaction sequences. Boundedness indicates the finiteness of reachability space. Places are often used to represent buffers and registers for storing intermediate data. By confirming the net is bounded or safe then it is guaranteed that there will be no overflows in the buffers and there will be no problem whether firing sequence is taken.

3.2.1.3 Liveness

The concept of liveness is closely related to the complete absence of deadlocks in operating systems. A Petri net (N, M_0) is said to be live (or equivalently M_0 is said to be a live marking for *N)* if, no matter what marking has been reached from M_0 , it is possible to ultimately fire any transition of the net by progressing through some further firing sequence. This means that a live Petri net guarantees deadlock-free operation, no matter what firing sequence is chosen.

Definition 21. A Petri net is said to be live if for all transitions there is a way to fire transition in any marking M' reachable from the initial marking M_0 it means; $\forall M^{'} \in R(M_0)$ *and* $\forall t \in T, \exists M \in R(M^{'})$ *such that t is enabled in* M.

Definition 22. A marking M' reachable from the initial marking M_0 is a deadlock if none of the transitions of the Petri net is enable in M'.

Note that according to the definition of liveness, if a transition is not deadlocked then it must be live, which means that there exists a firing sequence such that the transition will be enabled. It turns out that liveness quarantees the absence of deadlocks. In other words, every transition of the net can fire an infinite number of times. Specifically, we say a transition *t* is live at

- level 0 if, it can never fire, meaning that *t* is a dead transition;
- level 1 if, it can fire at least once, which means that there exists a marking $M \in R(M_0)$ such that transition *t* is enable;
- level 2 if, given any positive integer n, there exists a firing sequence that contains *t* at least *n* times;
- level 3 if, there exists an infinite-length firing sequence in which *t* can potentially occur infinitely many times but it can be blocked;
- level 4 if, it can fire infinitely many times and there is no way to block it.

The definitions explained in previous section causes following conclusions;

i. A Petri net is live at level *i* if every transition is live at level *i*.

ii. A Petri net is live if all the transitions are live.

The liveness that is often of interest is the one at level 4 - the strongest one. It should be noticed that liveness at level *i* implies one at level $i-1$.

3.2.1.4 Reversibility and Home State

It is very useful property used in different many application

Definition 23. A Petri net (N, M_0) is said to be reversible if for each marking $M \in R(M_0), M_0$ is reachable from M .

Thus, in reversible net one can always get back to the initial marking or state. In many applications, it is not necessary to get back to the initial state as long as one can get to some (home) state. Therefore, we relax the reversibility condition and define a home state.

Definition 24. A marking M' is said to be a home state if, for each marking $M \in R(M_0), M_0^{'}$ is reachable from M .

The above three properties (boundedness, liveness and reversibility) are independent of each other. For example, a reversible net can be live or not live and bounded or not bounded.

3.2.2 Structural Properties of Petri Nets

Structural properties of a Petri net depend only on its structure, and not on the initial marking and the firing policy. These properties are thus of great importance when designing manufacturing systems, since they depend only on the layout, and not on the way the system will be managed, which is not known at the design level. Most of the structural properties can be easily assert by means of algebraic techniques. The structural properties of a Petri net include liveness, boundedness, conservativeness, repetitivity, consistency and controllability properties.

Definition 25. If there exists an initial marking M_0 then Petri Net is structurally live.

Accordingly, if Petri net is live then it is also structurally live, but it is not reversible. It is important to notice that, except for some particular types of Petri nets, it is impossible to verify structural liveness.

Definition 26. A Petri net is said to be structurally bounded if it is bounded for any initial marking M_0 .

Definition 27. A Petri net is conservative, if all transitions fire token-preservingly, it means that all transitions add exactly as many tokens to their postplaces (output places) as they subtract from their preplaces (input places).

Definition 28. A Petri net is said to be repetitive if there exists an initial marking M_0 and a firable sequence σ in which each transition appears an unlimited number of times.

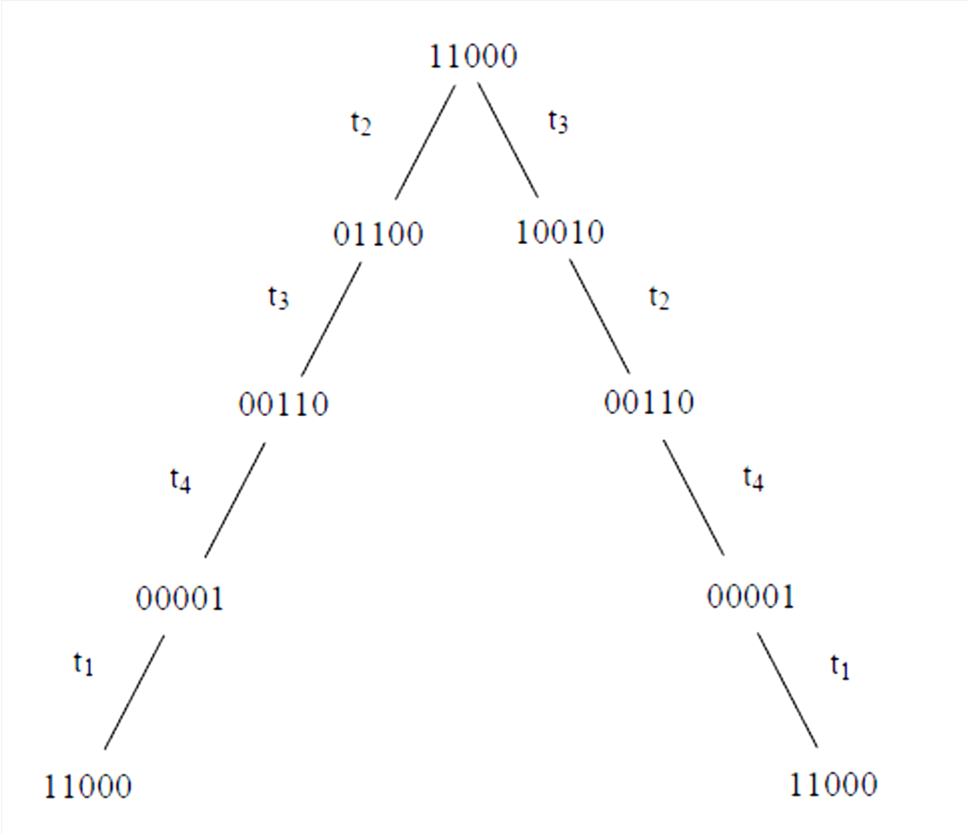
According to this definition, a Petri net which is structurally live is repetitive. However, the reciprocal is not true. The repetitivity is a necessary condition for structural liveness and consequently a necessary condition for liveness.

3.3 Analysis Method

Coverability tree and reachability graph, matrix equation approach and reduction (or decomposition) techniques are the tree methods used for analyzing Petri Nets. Coverability and reachability method used for medium size nets because of the complexity of state space explosion it is limited. On the other hand, other two methods are powerful but, they are used only for special cases.

3.3.1 Coverbility and Reachability Methods

From the initial marking M_0, it can be obtained as many new marking as the number of enabled transition. By this way, more markings obtained from each marking. Each node represents marking generated from M_0 and its successors. And each arc represents a transition firing, which transforms one marking to another. The reachability tree of Petri nets is shown in following figure.



For a bounded Petri Net, the coverability tree is called the reachability tree since it contains the all possible reachable markings. It is an exhaustive method and it is the only disadvantages of that.

The coverability graph of a Petri net (N, M_0) is a labeled directed graph $G = (V, E)$ where its node set V is the set of distinct labeled nodes in the coverability tree, and the arc set E is the set of arcs labeled with single transition t_k representing all possible single transition firings such as $M_i[t_k$ Mi[t_k >Mj where M_i, M_j $\in$V.

Because the reachabity tree is an exhaustive method, it can causes loss of information. Therefore, the reachability and liveness problems can not be solved by the coverbility tree method alone.

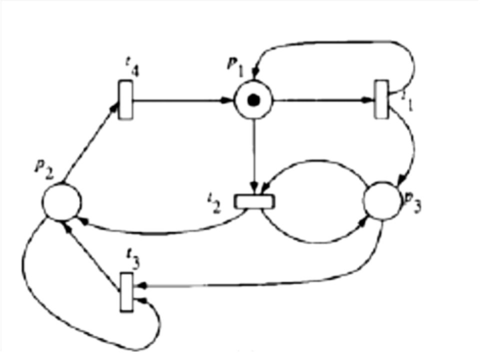


Figure 3.3: Live Petri Net

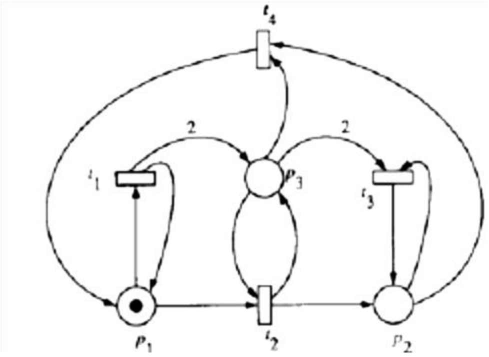


Figure 3.4: Non Live Petri Net

3.3.2 Place Invariant

If the set of place's token numbers count same for all possible marking then it is called place invariant. A single invariant is shown by an n-column vector x, where n is the number of places of the Petri net, whose nonzero entries stand for the places that concern the particular invariant and zeros everywhere else. A place invariant is defined as every integer vector x that gratifies

$$x^T\mu=x^T\mu_0$$

where p is the net’s initial marking, and p represents any subsequent marking. The equation shows that the weighted sum of the tokens in the places of the invariant remains constant at all markings and this sum is determined by the initial marking of the Petri net. The place invariants of a net can be computed by finding the integer solutions to

$$x^TD = 0$$

where D is the n X m incidence matrix of the Petri net, with n being the number of places and m the number of transitions of the net. It is easily shown that every linear combination of place invariants is also a place invariant for the net.

3.3.3 Transition Invariant

An n- vector x of integer is called invariant if $A^Tx=0$. Furthermore, an n- vector $X \geq 0$ is a T- invariant if and only if there exists a marking M_0 and firing sequence σ from M_0 back to M_0 with its firing count vector $\bar{\sigma}$ equal to x.

Chapter 4

MODELING AND SIMULATION

4.1 Modeling Optical Interconnection Networks with Petri Nets

In this section we present a P/T-net model of a DC and $2^n \times 2^n$ m-stage OMIN and explain the relationship between the components of modeled net and objects used in modeling tool.

Definition 29. A P/T-net of a DC is a 5-tuple $PTNCW_{2\times 2} = (P, T, A, W, M_0)$, such that

- $P = P_{IN} \cup P_{OUT}$, where $P_{IN} = \{\text{in0}, \text{in1}\}$ is the set of input places, $P_{OUT} = \{\text{in}i\text{out}j\}$, $0 \leq i, j \leq 1$ is the set of output places;
- $T = \{\text{straight}, \text{cross}\}$ is the set of transitions;
- $A = \begin{cases} (\text{in}i\text{,straght})\text{or } (\text{in}i\text{,cross}), & \text{for } i\text{=0,1} \\ (\text{straight,in}i\text{out}i), & \text{for } i\text{=0,1} \\ (\text{cross,in}i\text{out}j\text{ }), & \text{if } j\text{=1-}i \text{ and } i\text{=0,1} \\ \text{empty} & \text{othherwise} \end{cases}$
- $W = \{1\text{: for all } a \in A\}$;
- $M_0 = \begin{cases} 1, \text{if } p = \text{in}i \text{ for } 0 \leq i \leq 1 \\ 0, \text{otherwise} \end{cases}$

Table 4.1: Relationship between P/T-net objects and DC's components

P/T-net object	**DC's equivalent component**
in0, in1	inputs (input 0, input 1)
in0out0, in1out0	upper output (output 0)
in0out1, in1out1	lower output (output 1)

A P/T-net model of a DC is shown in Fig. 4.1. Relationship between P/T-net objects and DC's components are described in Table 4.1. In Fig. 4.1 (a) the P/T-net is in the initial state. Occurrence of *straight* transition changes the initial state to on state (Fig. 4.1(b)). Likewise, when transition *cross* fires the initial state gets changed to off state (Fig. 4.1(c)). $PTNCW_{2\times 2}$ fully describes DC's functionality.

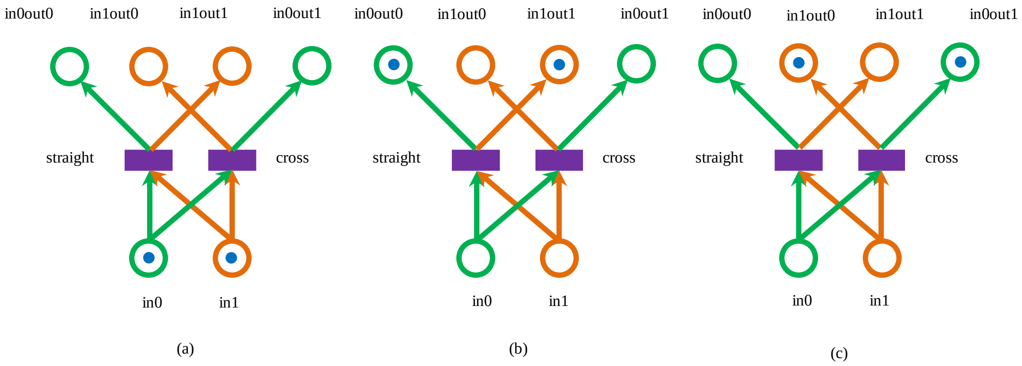


Figure 4.1: P/N-net model of 2 × 2 DC in (a) the initial state, (b) straight state, (c) cross state.

Definition 30. A P/T-net of $2^n \times 2^n$ m-stage optical network is 5-tuple $PTNOMIN_{2^n,m} = (P, T, A, W, M_0)$, such that

- $P = P_1 \cup \cdots \cup P_{m+1}$, where $P_k = \{p_1^{(k)}, \ldots, p_{2^{k-1}N}^{(k)}\}$ if $1 \leq k \leq n$ and $P_k = \{p_1^{(k)}, \ldots, p_{2^{n-1}N}^{(k)}\}$ if $n \leq k \leq 2n-1$;

- $T = \{T_1 \cup \cdots \cup T_m\}$, where $T_k = \{t_1^{(k)}, \dots, p_{4^{k-1}N}^{(k)}\}$ if $1 \leq k \leq n$ and $T_k =$ $\{t_1^{(k)}, \dots, t_{4^{n-1}N}^{(k)}\}$ if $n \leq k \leq 2n-1$;
- $A \subset \bigcup_{k=1}^{m} (P_k \times T_k) \bigcup (T_k \times P_{k+1})$;
- $W = \{1\text{: for all } a \in A\}$;
- $M_0 = \begin{cases} 1, & \text{if } p \in P_1 \\ 0, & \text{otherwise} \end{cases}$

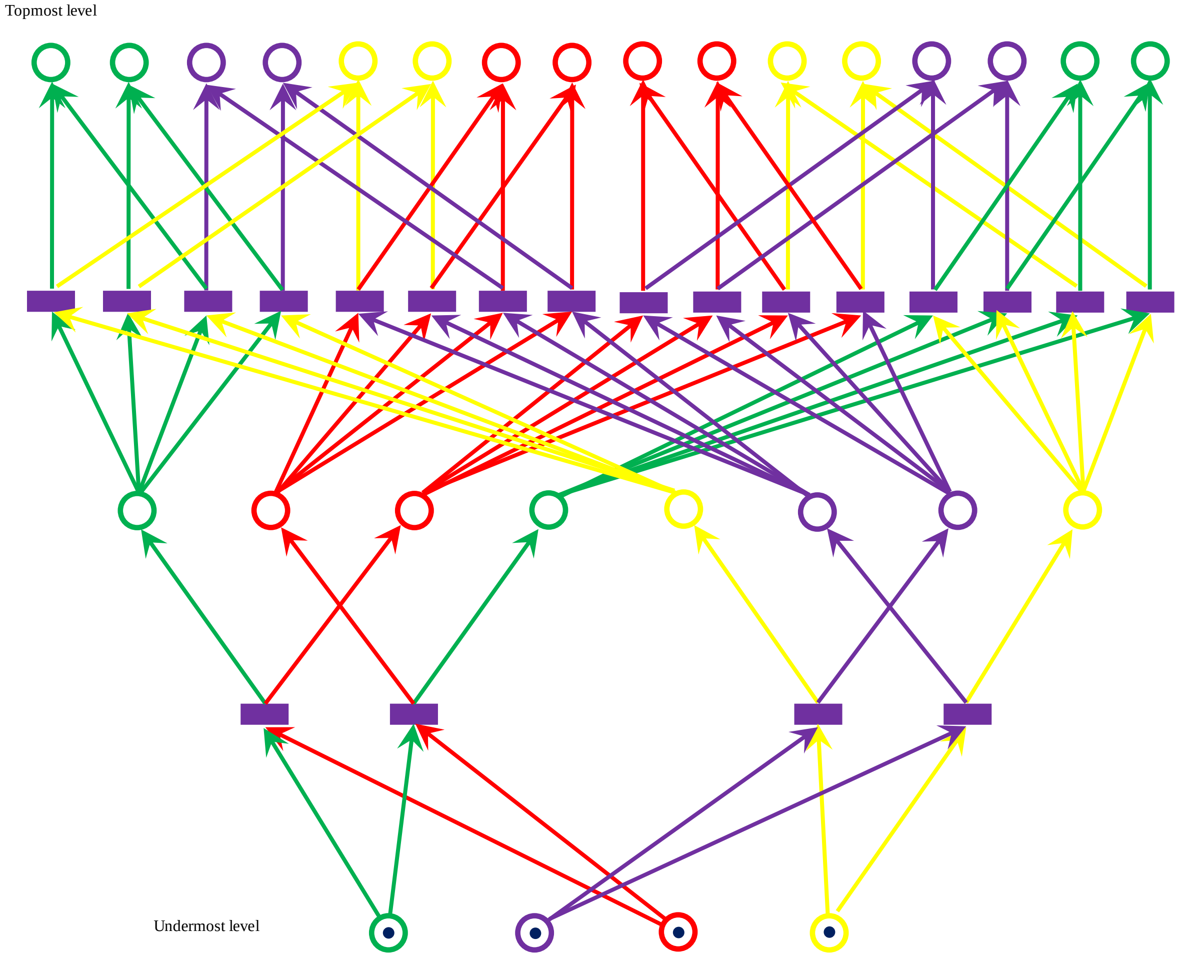


Figure 4.2: P/T-net model of 4 × 4 shuffle-exchange OMIN.

$PTNOMIN_{2^n,m}$ has multilevel structure with the places and the transitions located at distinct levels. The places at the undermost and the topmost levels respectively represent the input and output terminals of OMIN. According to the definition the arcs are between the net components belonging to the neighboring levels and directed from bottom to top. Occurrence of the maximal set of transitions in kth level sets all DCs in kth stage, thus allowing transitions in level $k+1$ to occur. This continues in the same way until transitions at the second topmost level get turn to fire. Then P/T-net becomes dead. Each dead marking specifies a permutation of the OMIN inputs into its outputs. Thus, one pass through a P/T-net realizes a particular permutation.

A *complete occurrence sequence* is a sequence of length m. By changing the order in which the transitions occur we can obtain a new complete occurrence sequence. There usually exist multiple complete occurrence sequences for each setting of the DCs in OMIN. All complete occurrence sequences are in fact equivalent in sense that they result in the same dead marking $\mathcal{M}_d$. Hence, the complete occurrence sequence can be represented in the following way:

$$\sigma_{i_1,\ldots,i_m} = M_0[S_1 \succ M_1[S_2 \succ \cdots M_{m-1}[S_m \succ M_d \qquad (1)$$

where $S_1 \subset T_k$ for $1 \leq k \leq m$ and M_d is a dead marking.

$PTNOMIN_{2^n,m}$ with $n = m = 2$ is illustrated in Fig. 4.2.

4.2 Modeling OMIN Topology

Arc determination in Definition 2 is rather cumbersome procedure. Indeed, we defined an arc to be element of $\bigcup_{k=1}^{m}(P_k \times T_k) \bigcup (T_k \times P_{k+1})$ and left question unanswered how to determine the two end points. The following two definitions are important.

Definition 31. A pattern arc $(p, t) \in P_k \times T_k$ is incident from p to t and highly depends on OMIN topology.

Table 4.2: Construction of pattern arcs for shuffle exchange type OMIN.

```
Input:  $2^n \times 2^n$ $m$-stage OMIN
Output: related P/T-net
LPA$(l, r, s)$ {constructs pattern arcs for the places in the left half-set}
for $k = 1$ to $r$ do
        for $i = 1$ to $2^{s-2}N$ do
                for $j = 1 + (i-1)2^s$ to $i2^s$ do
                        $A = A \cup (p_i^k, t_j^k)$
                end_for
        end_for
end_for
RPA$(l, r, s)$ {constructs pattern arcs for the places in the right half-set}
for $k = 1$ to $r$ do
        for $i = 1 + 2^{s-2}N$ to $2^{s-1}N$ do begin
                $t \equiv i \; mod \; 2^{s-2}N$;
                for $t = 1$ to $2^{s-2}N$ do
                        for $j = 1 + (t-1)2^s$ to $t2^s$ do
                                $A = A \cup (p_i^k, t_j^k)$
                        end_for
                end_for
        end_begin_for
end_for
Pattern Arc
if $m \leq n$ then begin $\mathbf{LPA}(1, m, k)$; $\mathbf{RPA}(1, m, k)$ end
        else begin
                $\mathbf{LPA}(1, n-1, k)$; $\mathbf{LPA}(n, m, n)$; $\mathbf{RPA}(1, n-1, k)$; $\mathbf{RPA}(n, m, n)$
        end_begin;
end_if
```

Definition 32. A switch arc $(p, t) \in P_k \times T_k$ is incident from t to p no matter which communication pattern is chosen to link the neighboring stages.

Pattern arcs inherit the characteristics of the interstage communication pattern. They are created according to the permutation induced by the interstage communication pattern.

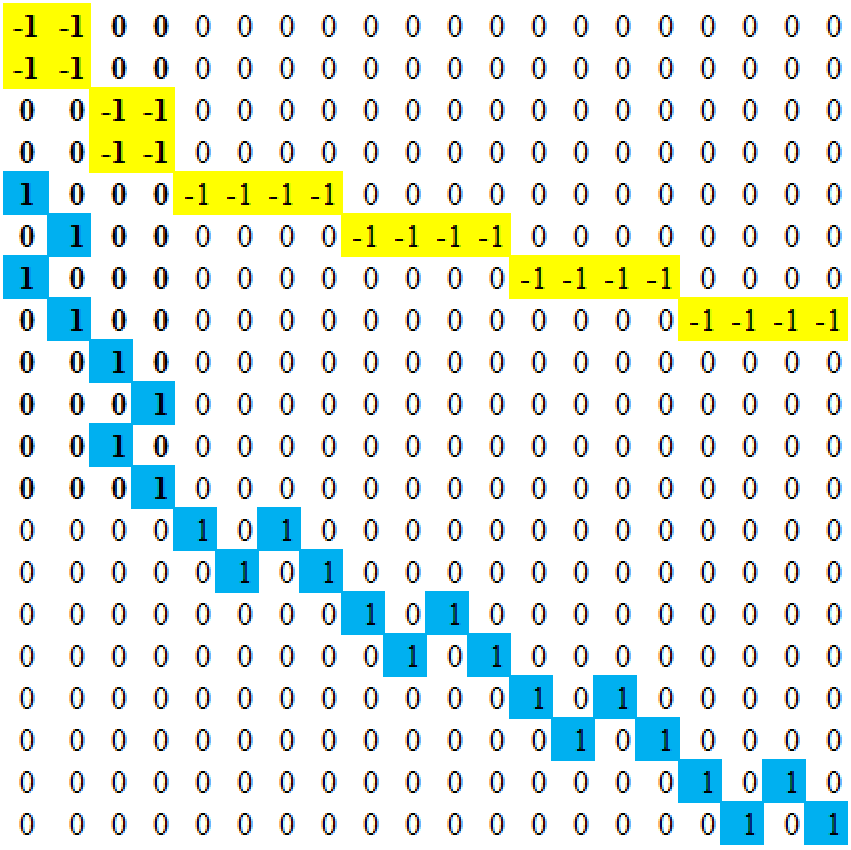
An example of algorithm that creates pattern arcs for shuffle-exchange type OMIN is shown in Table 4.2. The algorithm creates pattern arcs in breadth-first manner. In this algorithm, module $\mathbf{LPA}(l, r, s)$ is used to create pattern arcs in the left half-set. Similarly, module $\mathbf{RPA}(l, r, s)$ is use to design pattern arcs in the right half-set. Pattern arcs are equivalent to the shuffle interstage channels in OMIN (see figure 4.2).

A P/T-net obtained in such a way fully adequate to the logic of OMIN and can be used for further analysis of OMIN characteristics. We can now draw an incident matrix for P/T-net. Related incident matrix is a sparse block-diagonal matrix. Number of zeros depends on the matrix size but not the type of interstage pattern.

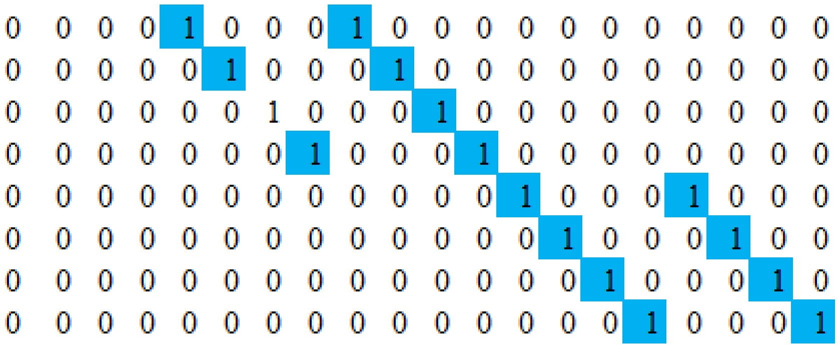
Interstage pattern however affects the distribution of zeros in the incidence matrix. For example, distribution of non-zero and zero elements in incident matrices of 4 × 4 1- and 2-stage shuffle-exchange OMIN are illustrated in In this figure large incidence matrix fully covers the small one. It can be easily seen that small matrix completely coincides with the related elements of the big one.

4.3 Analysis of Complexity and Acyclicity

Consider $PTNOMIN_{2^n,m}$. Let $|T|$ and $|P|$ be respectively the number of transitions and number of places. The following proposition measures the rate of increase of the P/T-net.



-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	-1	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	-1	-1	-1	-1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1



$$
\begin{matrix}
0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1
\end{matrix}
$$

Figure 4.3: Incidence matrices corresponding to 1- and 2-stage 4 × 4 shuffle-exchange OMIN.

Proposition 1. The following holds:

$$|P| = \begin{cases} (2^{m+1} - 1)N, if\ 1 \leq m < n \\ \frac{1}{2}(m - n + 4)N^2 - N, if\ n \leq m < 2n - 1, \end{cases}$$

and

$$|T| = \begin{cases} \frac{1}{3}(4^m - 1)N, \qquad if\ 1 \leq m < n \\ \frac{1}{4}\left(m - n + \frac{4}{3}\right)N^3 - \frac{1}{3}N, if\ n \leq m < 2n - 1. \end{cases}$$

Proof. We calculate $|P|$ seperately for the intervals $1 \ll m < n$ and $n \ll m < 2n - 1$. When $1 \ll m < n$, $|P|$ is represented calculated as the sum of the elemants of geometric series:

$$\sum_{k=1}^{m+1} |P_k| = \sum_{k=1}^{m+1} 2^{k-1}N = (2^0 + 2^1 + \cdots + 2^m)N = \frac{1 - 2^{m+1}}{1 - 2}N = (2^{m+1} - 1)N.$$

For $n \ll m < 2n - 1$ the number of places is calculated as follows:

$$\sum_{k=1}^{m+1} |P_k| = \sum_{k=1}^{n} |P_k| + \sum_{k=n+1}^{m+1} |P_k| = (N^2 - N) + \sum_{k=n}^{m+1} 2^{n-1} N$$

$$= (N^2 - N) + \frac{1}{2}(m - n + 2)N^2 = \frac{1}{2}(m - n + 4)N^2 - N.$$

Combining the above two formulas in one we obtain the claim regarding $|P|$.

Proof of the claim regarding $|T|$ also dissipates into intervals $1 \ll m < n$ and $n \ll m < 2n - 1$.

$$\sum_{k=1}^{m} |T_k| = \sum_{k=1}^{m} 4^{k-1} N = (4^0 + 4^1 + \cdots + 4^{m-1})N = \frac{1 - 4^m}{1 - 4} = \frac{1}{3}(4^m - 1)N.$$

$$\sum_{k=1}^{m} |T_k| = \sum_{k=1}^{n-1} |T_k| + \sum_{k=n}^{m} |T_k| = \left(\frac{1}{12}N^3 - \frac{1}{3}N\right) + \sum_{k=n}^{m} 4^{n-1} N = \left(\frac{1}{12}N^3 - \frac{1}{3}N\right)$$

$$+ \frac{1}{4}(m - n + 1)N^3 = \frac{1}{4}\left(m - n + \frac{4}{3}\right)N^3 - \frac{1}{3}N.$$

The claim regarding $|T|$ is reassured by bringing together the above two formulae.

■

Memory allocation is a problem of primary interest. Memory capacity needed to store a Petri net dramatically increases with increase of the number of places and transitions. It is often the case when a Petri net model that is exponential in the size of the original problem is created even for the problems of the modest size [16]. Such

Petri net models lead to inefficient use of the memory frequently causing memory overflow. Given $2^n \times 2^n$ OMIN, Proposition 1 expresses $|P|$ and $|T|$ as functions of m. An important conclusion is that both $|P|$ and $|T|$ are in polynomial dependence on the number on the OMIN size. This assures that memory capacity of the modern computers are indeed sufficient to run our task.

A Petri net based analysis of permutaton admissibility problem is considered in [5]. According to the method proposed in [5] we firstly create a CP-net model of the system then convert the resulting CP-net into complete unfolding. The size of the complete unfoldings obtained in [Bashirov et al] are represented by polynomial functions $|P_{unf}| = \frac{(m+2)N^2}{2} + \frac{mN}{2}$ and $|T_{unf}| = \frac{mN^3}{2}$. In the present research we compare the functions $|P_{unf}|$ and $|T_{unf}|$ with $|P|$ and $|T|$, and thus assess the relative merits of the Petri nets created in the present work and those proposed in [5] regarding their complexity. This is done in the following proposition.

Proposition 2 The following is valid:

$$|P| < |P_{unf}| \text{ and } |T| < |T_{unf}|.$$

Proof. We prove the claim through evaluating the ratios: $|P|/|P_{unf}|$ and $|T|/|T_{unf}|$. For $1 \leq m < n$ we obtain

$$\left|\frac{|P|}{|P_{unf}|}\right| = \frac{2(2^{m+1}-1)N}{(m+2)N^2+mN} = \frac{2(2^m-1)}{(m+2)N+m} \leq \frac{2N-2}{(m+2)N+m} < 1 \qquad (2)$$

and

$$\left|\frac{|T|}{|T_{unf}|}\right| = \frac{2(4^{m}-1)N}{3mN^{3}} < \frac{2N^{3}-2N}{3mN^{3}} = \frac{2-2/N^{2}}{3m} < 1. \tag{3}$$

Likewise for $n \leq m < 2n-1$ we obtain

$$\left|\frac{|P|}{|P_{unf}|}\right| = \frac{(m-n+4)N^{2}-2N}{(m+2)N^{2}+kN} = \frac{m-n+4-2/N}{m+2+k/N} < 1 \tag{4}$$

and

$$\left|\frac{|T|}{|T_{unf}|}\right| = \frac{(3m-3n+4)N^{3}-4N}{6mN^{3}} < \frac{3m-3n+4-4/N^{2}}{6m} < 1. \tag{5}$$

This ascertains that $|P| < |P_{unf}|$ and $|T| < |T_{unf}|$, as claimed.

■

P/T-nets considered in the present work are created from scratch based on the OMIN specifications. When creating a P/T-net we remove unnecessary components such as 0-bound places, false-guarded transitions, etc. consequently keeping OMIN components as minimal as possible. For each OMIN, resulting P/N-net is indeed an optimized complete unfolding. This is why each optimized complete unfolding is inevitably more compact than related complete unfolding.

Proposition 3 $PTNOMIN_{2^n,m}$ is an acyclic net.

Proof. The proof is straightforward from the Definition 15 and the way the complete occurrence sequence (1) is created.

■

Table 4.3: Results of computer tests for 4 × 4 OMIN.

Permutations	$\boldsymbol{m_{min}}$	**Time/sec**
$\pi_1 = (0132)$	1	0.01
$\pi_2 = (0321)$	2	0.01

Table 4.4: Results of computer tests for 8 × 8 OMIN.

Permutations	$\boldsymbol{m_{min}}$	**Time/sec**
$\pi_3 = (76230145)$	4	0.06
$\pi_4 = (10235476)$	1	0.01
$\pi_5 = (03461257)$	2	0.02
$\pi_6 = (70152436)$	3	0.03
$\pi_7 = (12460257)$	2	0.03

Table 4.5: Results of computer tests for 16 × 16 OMIN.

Permutations	$\boldsymbol{m_{min}}$	**Time/sec**
$\pi_8 = (15\ 7\ 14\ 6\ 13\ 5\ 12\ 4\ 11\ 2\ 10\ 3\ 9\ 1\ 8\ 0)$	4	0.11
$\pi_9 = (0\ 3\ 7\ 5\ 8\ 10\ 13\ 15\ 1\ 2\ 4\ 6\ 9\ 11\ 12\ 14)$	2	0.04
$\pi_{10} = (14\ 3\ 9\ 10\ 11\ 15\ 1\ 2\ 13\ 0\ 7\ 8\ 5\ 6\ 12\ 4)$	6	0.34
$\pi_{11} = (7\ 3\ 15\ 11\ 6\ 2\ 14\ 10\ 5\ 1\ 13\ 9\ 4\ 0\ 12\ 8)$	3	0.08
$\pi_{12} = (13\ 14\ 15\ 12\ 9\ 5\ 11\ 3\ 7\ 6\ 10\ 4\ 1\ 0\ 3\ 2)$	3	0.10

Given a marking M, solution of the system of algebraic equations, that is discussed in the previous chapter, is a neccessary but not sufficent condition for reachability of the marking M from the initial marking M_0. The method of P-invariants may in general lead to spurious solutions [20]. This means that even though system of linear algebraic equations has a solution, the given marking M is not necessarily reachable from the itital marking M_0. For acyclic P/T-nets the above condition is also sufficient. This is the reason why claim of the Proposition 3 is crucial. Proposition 3 allows us to implement the mehod of P-invarinats for the optimized complete unfoldings considered in the prosent work.

4.4 Results of Computer Experiments

Correctness of the approach was tested through series of computer experiments. Minimum number of stages m_{min} was calculated for randomly selected permutations π_1 through π_{12}, that are presented in Table 4.3 – 4.5. The tests were conducted for 4×4, 8×8, and 16×16 shuffle-exchange OMINs. We used Matlab on Windows XP platform with 3 GHz CPU and 4 GB RAM to solve linear algebraic equations.

The main idea behind of the computer tests is as follows. It is known that $2^n \times 2^n$ $2n - 1$ shuffle-exchange OMIN is rearrangeable. This particuarly means that $(2n - 1)$-stage shuffle-exchange OMIN can perform arbitrary 2^n-permutation. We test 1- through $(2n - 1)$ -stage shuffle-exchange OMINs. Test for realizability of the given permutation terminates as soon as the minimum number of the stages m_{min} that is sufficient to perform the permutation is found. The resuts of computer tests are illustrated in Table 4.3 – 4.5.

In fact, the incidence matrices and firing vectors that have been used in the present research are sparse in nature. A typical example of the OMIN incidence matrix is illustrated in Figure 4.3. We are aware, that there exist dozens of efficient methods for solving the system of equations with sparse matrix of coefficients. We did not attempt, however, to use any of known methods for sparse systems since standart Matlab involved solution of linear equations for the most time consuming case of π_{10} was completed in 0.34 sec. It must be noticed a result of "no solution" was obtained at no time. We used a simple interface to convert data into suitable for Matlab format – a retangular table of elements. This is also important that time span indicated in the Tables 4.3-4.6 is the cummulative time spent required to check the task for the stages 1- through m_{min}. For instance, considering still same example of π_{10}.

Comparison of the time spans spent to perform the computer tests in the present research and in [5] clearly illustrates the time improvement achieved in the current approach.

Chapter 5

CONCLUSIONS

In this thesis we have applied Petri net based methods and techniques to minimize path dependent loss and number of switch crosstalks in multistage optical interconnection networks. The main conclusion is that both path dependent loss and number of switch crosstalks can be reduced essentially through realizing a permutation on an optical network employing m_{min} the minimal number of stages.

Other results and observations include:

- Petri nets are adapted to the structure of both problems;
- it is rather easy to create Petri net model;
- Petri nets that we deal with throughout the research are in polynomial dependence on the problem size, which alleviates the memory consumption;
- Petri nets that we create in this thesis are acyclic, which enables us to use P-invariants method;
- complexity results obtained in the previous research [1] have improved;
- the time-effectiveness has been proved through performing a series of computer experiments.

REFERENCES

[1] Bashirov, R., Rearrangeability of $(2\log N - 1) -$ stage networks employing a uniform connection pattern, Calcolo, Springer Verlag, 38, pp. 85-95, 2001.

[2] Bashirov, R., Crespi, V., Analyzing the permutation capability of multistage interconnection networks with colored Petri nets, Information Sciences, 176, pp. 3143-3165, 2006.

[3] Bashirov, R., Karanfiller, T., On path dependent loss and switch crosstalk reduction in optical networks, Information Sciences, 180, pp. 1040-1050, 2010.

[4] Bashirov, R., Karanfiller, T., Lort, H., An approach to reduce path-dependent-loss in optical networks implementing Petri nets, In: Prococeedings of 8th International Conference on Application of Fuzzy Systems and Soft Computing, Quadrat Verlag, Helsinki, 1-3 September, pp. 91-96, 2008.

[5] Bashirov, R., Kordon, F., Lort, H., Exploiting colored Petri nets to decide on permutation admissibility, Acta Informatica, 46, pp. 43-55, 2009.

[6] Beneš, V. E., Mathematical Theory of Connecting Networks and Telephone Traffic, Acdemic Press, New York, 1965.

[7] Benhadddou, D., Chaudhry, Photonic switching techniques and architecture for next generation optical networks, Cluster Computing, 7, pp. 281-291, 2004.

[8] Das, N., Bhattacharya, B. B., Bezrukov, S., Menon, R., Sarkar, A., Permutation routing in optical MINs with minimum number of stages, Journal of Systems Architecture, 48, pp. 311-323, 2003.

[9] Du, Y. Y., Jiang, C. J., Zhou, Y. F., Modelling and monitoring of E-commerce workflows, Information Sciences, 179, pp. 995-1006, 2009.

[10] Duato, J., Yalamanchili, S., Ni, L., Interconnection Networks, IEEE Computer Society, NJ, 1997.

[11] Hinton, H. S., A nonblocking optical interconnection network using directional couplers, In: Proceedings of IEEE Globecom'84, Atlanta, GA, Nov. 1984, Vol.2, pp. 885-889.

[12] Jajszczyk, A., A class of directional-coupler-based photonic switching networks, IEEE Transactions on Computers, 41, pp. 599-603.

[13] Jiang, X., Sheng, H., Khandker, M. R., Horiguchi, S., Blocking behaviors of crosstalk-free optical banyan networks on vertical stacking, IEEE/ACM Transactions on networking, 11, pp. 982-993, 2003.

[14] Kao, K. C., Hockham, G. A., Dielectric-fibre surface waveguides for optical frequencies, Proc. I.E.E., 113, pp. 1151-1158, 1966.

[15] Katangur, A. K., Akkaladevi, S., Pan, Y., Analyzing the performance of optical multistage interconnection networks with limited crosstalk, Cluster Computing, 10, pp. 241-250, 2007.

[16] Khomenko, V., Kondratyev, A., Koutny, A., Vogler, W., Merged processes: a new condenced representation of Petri net behaviour, Acta Informatica, 43, pp.307-330, 2006.

[17] Lea, C., Crossover minimization in directional-coupler-based photonic switching systems, IEEE Transactions on Communications, 36, pp.355-363, 1988.

[18] McCarthy, D. C., Photonic switches: fast, but functional, photonics spectra, March, pp.141-150, 2001.

[19] Midwinder, J. E., Photonics in switching: The next 25 years of optical communications?, IEE Proceedings, 139, pp. 1-12, 1992.

[20] Murata, T., Petri nets: properties, analysis and applications, Proceedings of IEEE, 77, pp.541-580, 1989.

[21] O'Connor, I., Optical solutions for system level interconnect, In: Proceedings of SLIP'2004, Paris, pp.14-15, 2004.

[22] Padmanabhan, K., Netravali, A. N., Dialated networks for switching, IEEE Transactions on Communications, 35, pp.1357-1365, 1987.

[23] Pan, Y., Optical multistage interconnection networks: New challenges and approaches, IEEE Communications Magazine, 37, pp. 50-56, 1999.

[24] Sahni, S., Matrix multiplication and data routing using a partitioned optical passive stars network. IEEE Transactions on Parallel and Distributed Computing, 11, pp.720-728, 2000.

[25] Savastano, L., Maier, G., Martinelli, M., Pattavina, A., Performance comparison of guided-wave architectures for space-division photonic switching, In: Proceedings of the 1st International Conference on Broadband Networks, IEEE Press, 2004.

[26] Shen, X., An optimal $O(NlgN)$ algorithm for permutation admissibility to extra-stage cube-type networks, IEEE Transactions on Computers, 44, pp.1144-1149, 1995.

[27] Shen, X. Y., Pan, Y., Equivalent permutation capabilities between time-division optical omega networks and non-optical extra-stage omega networks, IEEE Transactions on Networking, 9, pp.518-524, 2001.

[28] Shen, X., Xu, M., Wang, X., An optimal algorithm for permutation admissibility to multistage interconnection networks, IEEE Transactions on Computers, 44, pp.604-608, 1995.

[29] Shen, X., Yang, F., Pan, F., Equivalent permutation capabilities between time-division optical omega networks and non-optical extra-stage omega networks, IEEE Transactions on Networking, 9, pp.518-524, 2001.

[30] Tian, H., Katangur, A. K., Zhong, J., Pan, Y., A novel multistage architecture with multicast and broadcast capability, The Journal of Supercomputing, 35, pp.277-300, 2006.

[31] Vaes, M. M., Lea, C. T., Strictly nonblocking directional-coupler based switching networks under crosstalk constraint, IEEE Transactions on Communication, 48, pp.316-323, 2000.

[32] Vaes, M. M., Lea, C.-T., Wide-sense nonblocking Banyan type switchnig systems based on directional couplers, IEEE Transactions on Communications, 16, pp. 1327-1332, 1998.

[33] Yang, Y., Wang, J., Pan, Y., Permutation capability of optical multistage interconnection networks, Journal of Parallel and Distributed Computing, 60, pp.72-91, 2000.

[34] Wong, T.-S., Lea, C.-T., Crosstalk reduction through wavelength assignment in WDM photonic switching networks, IEEE Transactions on Communication, 49, pp.1280-1287, 2001.