Laboratoire d'Informatique
Fondamentale de Lille

numéro d'ordre : **3475**

Thèse présentée pour obtenir le titre de Docteur de l'Université des Sciences et Technologies de Lille (Label Européen), spécialité Informatique

# Une Méthodologie Multi-Critères Pour l'Évaluation de Performance Appliquée aux Architectures de Réseaux d'Interconnexion Multi-Étages

Ahmad Chadi ALJUNDI

Laboratoire d'Informatique Fondamentale de Lille
Université des sciences et Technologies de Lille

Soutenue le 9 juillet 2004
Après avis de Daniel ETIEMBLE, Jack DONGARRA et Miroslaw MALEK
Devant la commision d'examen formée de

Isaac SCHERSON (Président)
Daniel ETIEMBLE (Rapporteur)
Jack DONGARRA (Rapporteur)
Miroslaw MALEK (Rapporteur)
Jean-Luc DEKEYSER (Directeur)
M-Tahar KECHADI (Examinateur)
Pierre BOULET (Examinateur)

## Une Méthodologie Multi-Critères Pour l'Évaluation de Performance Appliquée aux Architectures de Réseaux d'Interconnexion Multi-Étages

Dans cette thèse, nous nous intéressons à l'évaluation de performances des réseaux d'interconnexion multi-étages. Le travail présenté couvre deux aspects essentiels : la définition d'une méthodologie multi-critères pour l'évaluation et la comparaison de réseaux d'interconnexion. Cette méthodologie est basée sur la définition d'une fonction de distance dans un espace multidimensionnelle, où chaque dimension représente un facteur de performance. La fonction peut être utilisée dans un contexte d'optimisation Pareto ou dans le contexte d'une classification. Le deuxième aspect concerne la proposition d'une nouvelle famille de réseaux d'interconnexion multi-étages baptisée les réseaux d'interconnexion Delta surdimensionnés. Cette famille de réseaux fournit des performances meilleures que celles des réseaux Delta au prix d'une complexité plus élevée. La méthodologie est utilisée pour comparer les performances de deux familles en prenant en compte cette complexité plus élevée.

## A Multi-Criteria Methodology for the Performance Evaluation of Multistage Interconnection Network Architectures

In this thesis, we are interested in the performance evaluation of multistage interconnection networks. The presented work covers two essential aspects: the definition of a multi-criteria methodology for the evaluation and comparison of interconnection networks. This methodology is based on the definition of a distance function in a multidimensional space, where each dimension represents a performance metric. The function can be used in a Pareto optimisation context or in the context of a classification. The second aspect is the proposition of a novel family of multistage interconnection network called over-sized Delta interconnection networks. This family of networks provides better performance than Delta networks but has a higher complexity. The methodology is used to compare the performance of the two families considering this complexity difference.

# Remerciments

Je tiens à remercier tout d'abord les membres de mon jury de thèse:

- Isaac SCHERSON pour avoir accepté la présidence du jury, mais également pour m'avoir soutenu tout au long de cette thèse et notament dans les moments difficiles.

- Jack DONGARRA, Daniel ETIEMBLE et Miroslaw MALEK pour avoir passé de nombreuses heures à lire mon manuscrit et pour la justesse de leurs remarques.

- Pierre BOULET qui a toujours été là pour répondre à mes question et qui a une fois de plus accepté de me faire profiter de ses critiques pertinentes en tant qu'examinateur.

- Tahar KECHADI qui a été à l'initiative de ce projet; les nombreuses discussions que nous avons eues au cours de mes visites à Dublin ont toujours profité à l'avancement de mes travaux.

-Jean-Luc DEKEYSER qui m'a encadré pendant quasiment cinq ans et qui par son expérience, ses connaissances et ses conseils a su mener à bien mon initiation dans le monde de la recherche tout en me laissant une large autonomie.

Au cours de mes recherches j'ai bénéficié de l'aide de nombreuses personnes et notamment: El-ghazali TALBI, David SIMPLOT, Alexandre SIDOGLAVIC, mais également Nicolas JOZEFWIEZ.

J'ai bénéficié en outre du soutien de tous les membres de mon équipe de recherche: l'équipe WEST du LIFL. Je tiens à remercier principalement:

- Philippe MARQUET pour sa disponibilité pendant ces années.

- Julien SOULA, Philippe DUMONT, Stéphane AKHOUN et Mickael SAMYN qui m'ont aidé à comprendre les arcanes de Linux et de Latex et avec qui j'ai noué des liens d'amitié qui, je l'espère, perdureront au-delà de cette thèse.

- Abdelkader AMAR à qui je souhaite toute la réussite et tout le bonheurqu'il mérite.

- Je n'oublie pas tous les autres membres de l'équipe: Alexandre HESSEMANN, Eric PIEL, Ouassila LABBANI, Linda KAOUANE, Ashish MEENA, Arnaud CUCCURU, Lossan BONDÉ, Javed DULLOO, Cédric DUMOULIN, Samy MEFTALI, ainsi que tous les autres membres associés.

Je n'oublie pas non plus tous mes autres amis thésards et docteurs: Amar ALJER, Iyad ALSHABANI, Emmanuel RENAUX, Omran ALCHIHABI, Hazem ISSA, Julien CARIGNY et Violeta FELEA.

En cette fin de thèse, je ne peux que remercier Imad MOUSTAPHA qui a été l'origine de ma vocation pour la recherche.

Comme me l'a dit un jour Kathleen « C'est l'une des merveilles de cette vie que des gens venant de territoires éloignés puissent se rencontrer, trouver des points communs et parler un langage commun ». C'est vrai que j'ai fait des connaissances merveilleuses au cours de ma thèse. Je pense notamment à Kathleen, à Karol, à la famille BALHAWAN qui est devenue au fil de temps « ma famille en France ».

Mais bien sûr, mes remerciments vont en premier lieu à ma propre famille pour toutes leurs affections.

# Contents

# List of Tables

# List of Figures

# Synthèse

## 0.1 Introduction

L'utilisation actuelle des ordinateurs ne se limite plus aux domaines scientifiques, commerciaux et industriels, mais ils sont utilisés aussi dans le domaine des communications. Le nombre croissant d'applications de traitement automatique de données, de communications, ainsi que l'augmentation du nombre d'utilisateurs conduisent au besoin de moyens de calcul automatique de haut niveau.

Ce niveau élevé de calcul peut être établi par plusieurs moyens logiciels et matériels. Beaucoup de travail et de recherche scientifique sont produits sur de nombreux domaines logiciels pour aboutir à ce niveau élevé de performances. A titre d'exemple, on peut citer le travail sur les algorithmes d'optimisation pour des buts spécifiques tels que les applications temps réel.

L'autre domaine susceptible d'améliorer les performances des machines parallèles est celui du matériel. Dans ce cadre, les travaux de recherche sont quasiment dominés par des multinationales tels que IBM, DELL, SUN, Intel, etc. Cependant, cette domination n'empêche pas des laboratoires de recherche académique à continuer à travailler sur de nouvelles idées et aspects matériels.

Dans notre équipe, nous nous intéressons au parallélisme et aux architectures parallèles. Le parallélisme est un des moyens utilisés pour l'amélioration de capacités de calcul des ordinateurs. Il peut être défini de façon simple par l'utilisation coopérative de plusieurs processeurs pour résoudre un seul problème. Cette coopération de processeurs exige un certain niveau de communication parmi les processeurs et des moyens pour la gestion de leurs accès concurrents à la mémoire. Beaucoup de travail est consacré pour répondre à des questions posées par ces situations. Un des domaines de cette recherche est l'étude du système de communication dans la machine, connu par les informaticiens comme étant le réseau d'interconnexion.

Les réseaux d'interconnexion multi-étages (RIMs) sont un exemple de ces systèmes de communication. Ils sont largement utilisés dans les ordinateurs parallèles et réputés pour avoir un large débit avec un niveau de complexité acceptable. Cette compensation entre le débit et la complexité, qui peut être exprimé aussi comme une compensation de qualité / coût, empêche, d'une part, la définition d'un meilleur réseau pour toutes applications. D'une autre part, il n'existe pas une définition universellement acceptée des performances d'un ordinateur parallèle puisque de nombreuses métriques de performances peuvent être considérées. Dans la littérature on peut trouver un grand nombre de propositions de RIM, qui pour un algorithme de routage approprié peuvent avoir un comportement idéal pour une application et une métrique spécifique. Néanmoins, les études d'évaluation de performances de réseaux d'interconnexion qui considèrent le problème en tant qu'un problème de décision multicritères ne sont pas nombreuses. Les quelques exemples existants sont soient informelles ou ils ne sont pas universelles.

## 0.2 Le problème de l'évaluation de performances de RIMs

L'évaluation de performances et la comparaison d'ordinateurs parallèles a été présenté dans [62]. Le problème entier est déjà très clairement résumé dans le chapô du papier : « *L'efficacité de réseau change considérablement en fonction d'applications et d'environnements d'opération. Cependant, même si ces variables sont fixées, les métriques de coût et de performances doivent être choisis. Scientifiquement, la détermination du meilleur réseau est aussi difficile que d'être certain qu'un animal est meilleur qu'un autre.* »

En informatique, comme dans beaucoup d'autres domaines scientifiques, économiques et industriels, des décisions multi-critères sont souvent nécessaires. Un problème de décision multi-critères a plus qu'un objectif, chacun correspondant à une solution optimale différente. Ces solutions optimales individuelles sont souvent contradictoires. Ainsi, la solution d'un tel problème n'est pas unique mais elle serait l'une des solutions optimales [26].

La plupart des recherches établies sur l'évaluation de performances de RIMs ne considère pas le problème en tant que problème multi-objectifs. De nombreuses études ont évalué le débit, la latence et d'autres métriques comme des facteurs séparés. Deux exemples d'une évaluation multi-objectifs sont présentés dans la suite.

Dans [20], Cheemalavague et Malek ont étudié l'effet du degré des RIMs Banyans sur leurs performances. Le débit et la latence ont été mesurées en utilisant une stratégie de commutation de paquets. Le rapport débit/latence a été utilisé comme facteur d'évaluation. L'utilisation de ce rapport a été justifiée par l'importance de deux métriques et par le fait qu'elles peuvent donner une indication du nombre moyen de « paquets qui pouvaient sortir par le réseau ».

Cette étude de Cheemalavague et Malek est limitée à l'évaluation de deux métriques. Comme nous l'avons dit, une évaluation de performances de RIMs nécessite l'évaluation de plus que deux facteurs, et donc cette possibilité aura besoin d'une évaluation plus formelle que l'utilisation du simple rapport de deux métriques.

Un autre exemple de l'approche multi-objectifs de l'évaluation de performances de RIMs est le travail de Lakamraju *et al.* qui ont présenté un méthode pour la synthèse de réseaux d'interconnexion [57] . Le but des auteurs était de pouvoir synthétiser des réseaux qui satisfont un certain nombre de conditions. Le travail a été appliqué sur des réseaux aléatoires réguliers, qui étaient passés successivement à travers des filtres qui écartaient les réseaux qui ne correspondaient à des certains critères. A la fin de la procédure, le résultat est un groupe de *bons* réseaux parmi lesquels un peut être choisi. L'importance de cette technique vient de sa capacité à considérer un nombre quelconque de facteurs de performances.

La technique est basée sur la spécification des mesures de performances intéressantes et la création de réseaux aléatoires réguliers qui passent par la suite à travers une *banque* de filtres. « *Chaque filtre est associé à un facteur de performances désiré. Les filtres identifient un sous-groupe de réseaux qui possèdent les caractéristiques demandées par rapport aux mesures spécifiées. Ce sous-groupe constitue une petite liste de réseaux parmi lesquels le concepteur peut en choisir un* ».

Le filtre se compose de deux parts: une évaluation qui calcule la valeur associée de la métrique de performance et un part qui vérifie et compare les valeurs mesurées à un certain seuil. Les sorties d'un filtre, qui sont un sous-groupe de l'entrée, sert comme entrée au filtre suivant. « *Les filtres sont rangés séquentiellement l'un après l'autre dans une priorité décroissante des mesures à évaluer* ».

Cette technique de filtrage est intéressante comme une approche multi-objectifs de l'évaluation de performances de réseaux d'interconnexion, mais elle n'est pas facile à utiliser car elle manque de formalisation mathématique. En outre, l'évaluation par rapport à différents seuils nécessite plusieurs phases d'évaluation, et la comparaison ne donne pas une vue globale des réseaux comparés.

## 0.3 La contribution de cette thèse

Le travail de cette thèse a été initié comme une étude de performances d'un nouveau RIM. L'étude des caractéristiques de ce réseau a conduit à la proposition d'une nouvelle classe de RIMs, baptisé les réseaux Delta sur-dimensionés. Cette classe de réseaux contient des RIM généralement plus complexes, mais qui semblent plus performants que les réseaux Delta. De ce fait, la question principale sur laquelle cette thèse est basée peut être résumée par la justification de l'utilisation de réseaux plus complexes pour obtenir plus de performances.

Cette étude a été confrontée par la non-existence d'une méthodologie multi-critères pour l'évaluation de performances précédemment discutée. Par conséquent, nous avons essayer de trouver un moyen pour l'évaluation de performances multi-critères, autrement dit, pour évaluer plusieurs métriques de performances simultanément. Le résultat était une méthodologie d'évaluation de performances multi-critères qui projette les métriques sur un espace multi-dimensionnelle et qui transforme de cette manière le problème à un problème d'évaluation d'une seule fonction de distance.

Alleyne et Scherson ont prouvé dans [7] qu'un réseau Delta précédé par un étage de permutation aléatoire peut simuler tous les réseaux de même taille pour des performances équivalentes. Toutefois, certaines architectures, telles que les systèmes SoC qui utilisent de réseaux sur puce, pour lesquels un tel étage ne peut pas être ajouté avec un coût raisonnable, sont des applications potentielles de la méthodologie proposée.

### 0.3.1 La méthodologie multi-critères d'évaluation et de comparaison de réseaux d'interconnexion

Dans cette thèse, le problème de l'évaluation et de comparaison de RIMs est considérée comme un problème de décision multi-critères. La solution proposée à ce problème est une fonction de distance baptisée UPF (Universal Performance Factor). Cette fonction peut être utilisée soit pour la proposition d'un groupe de solutions optimales, soit pour la comparaison de plusieurs réseaux par rapport à leurs fonctions de distance. La méthodologie est basée sur des mesures obtenus pas des simulations.

Plusieurs techniques sont utilisées pour la solution des problèmes de décision multi-critères. Toutes ces techniques consistent à transformer le problème de décision multi-critères pour en trouver une représentation mono-valeur de chaque solution. Cette tâche n'est pas toujours facile. Cependant, les fonctions de distance, utilisée lors qu'une solution idéale est connue ou lors qu'une solution plus commode est cherché, sont souvent utilisées.

En supposant que le choix des facteurs à évaluer ainsi que leurs importances est un processus de conception, nous divisons les facteurs de performances en deux groupes principales, un groupe de facteurs à maximiser $p^{max} = \{p_1^{max}, p_2^{max}, \dots, p_k^{max}\}$ où $k$ est le nombre de facteurs à maximiser, et les facteurs à minimiser $p^{min} = \{p_1^{min}, p_2^{min}, \dots, p_l^{min}\}$, où $l$ et le nombre de facteurs à minimiser. Par conséquent, l'UPF peut être défini par:

$$UPF = \sqrt{\sum_{i=1}^{l} \left( \frac{p_i^{min}}{MAX(p_i^{min})} \right)^2 w_i + \sum_{j=1}^{k} \left( 1 - \frac{p_j^{max}}{MAX(p_j^{max})} \right)^2 w_j} \qquad (1)$$

avec la définition suivante:

**Definition 1.** *Ayant deux réseaux $\mu_1$ et $\mu_2$ et leurs UPFs, $UPF_1$ et $UPF_2$. On dit que $\mu_1$ et plus performant que $\mu_2$ si $UPF_1 < UPF_2$.*

Cette fonction peut également être utilisé dans un contexte d'optimalité Pareto pour chercher un ensemble de réseaux optimaux par rapport aux métriques considérées.

### 0.3.2 Les réseaux Delta sur-dimensionés

Les réseaux Delta sont des très bons moyens de communication utilisés dans les ordinateurs parallèles. Cependant, des techniques peuvent être y appliquées pour améliorer leurs performances, tout en continuant de bénéficier de leurs caractéristiques telle que la simplicité. Ces techniques comprennent, entre autres, l'augmentation, la bufférisation et les réseaux optiques.

Nous proposons dans cette thèse une nouvelle technique que nous appelons le sur-dimensionnement. Deux différences principales existent entre les réseaux Delta et les réseaux Delta sur-dimensionés. Premièrement, les commutateurs du premier étage dans un réseau sur-dimensionés sont des commutateurs de taille $1 \times r$ et ceux du dernier étage sont de taille $r \times 1$, autrement dit, ce réseau n'est pas uniforme. L'autre différence est que le réseau sur-dimensioné se compose de plusieurs copies de réseau Delta qui se connecte par un étage ayant la propriété Delta.

Tant que ce réseau reste un réseau Banyan, il est capable de mieux distribuer la charge de communication sur les commutateurs qu'un réseau Delta simple. Néanmoins, cette nouvelle architecture est plus complexe que ce dernier.

## 0.4 Les résultats de l'évaluation de performances

Deux réseaux sont principalement évalués: le réseau Omega, un membre de la famille Delta, et le réseau MCRB, un membre de la famille Delta sur-dimensioné. Le réseau MCRB a été proposé par Kechadi dans [50]. Trois évaluations différentes sont présentées: une évaluation générale, une étude de l'effet de degré du réseau sur ses performances et enfin une étude de la complexité de réseaux d'interconnexion multi-étages.

# Chapter 1

# Introduction

## 1.1 Motivation

Today, computers are not only used as machines capable of data treatment in scientific, commercial and industrial domains, but they are also used for communication purposes. The increasing number of automatic data processing applications as well as the expanding number of users, especially for communication purposes, are the main reasons behind the increasing need for high performance computing.

This high performance computing can be achieved by different means and a lot of work is being done on all software and hardware levels.

A huge amount of scientific research is being carried out in numerous areas of software development in order to achieve this high computing capacity. Just as a simple example we can cite all the work on optimization algorithms for special purposes such as real time applications and the enormous improvement in operating systems.

While great advances in software development are being made in computer science research laboratories all over the world, one has to acknowledge that research into hardware develoment is almost dominated by multinational companies such as IBM, DELL, SUN, Intel, etc.

This domination would not, by any means, prevent some research teams from continuing to work on and develop novel ideas in this field.

In our team, we are concerned with parallelism and parallel architectures. Parallelism is one method used to improve the calculation capacities of today's computers. It can simply defined as: the use of more than one processor for the solution of a unique problem in cooperative ways. This cooperation of processors requires a certain level of communication between them and/or some methods for allowing the concurrent access to memory. A lot of work is devoted to answering the questions raised in trying to fulfil these requirements. The study of the machine's communication system, known to computer scientists as an interconnection network, represents one field of research work in this area.

Historically, these communication systems were based on telephone switching systems which had to be able to maximize the possibility of connecting any two idle terminals. Communication systems in parallel computers have the same objective, which is to maximize the memory accessibility of processors and their capacity to communicate with one another.

One communication system for parallel computers based on this kind of telephone switching system is a class of interconnection networks called the multistage interconnection network. Multistage interconnection networks are widely used in parallel computers as they have a relatively high throughput with an acceptable complexity. This throughput complexity trade-off, which can also be described as a quality cost trade-off is the reason behind the impossibility of defining a best interconnection network for all applications. Moreover, the performance depends largely on the running application and in addition, no one definition of the performance of a parallel computer exists because many different performance metrics can be considered. Many multistage interconnection networks are proposed in the literature. With an appropriate routing algorithm, all of these networks are proved to be ideal for certain applications and for a certain performance metric. However, studies that take into account the performance evaluation of multistage interconnection networks as a multiple criteria decision making problem are not numerous, and the few existing in the literature are either not universal or not formal.

In this dissertation we propose a multi-criteria performance evaluation methodology for multistage interconnection networks which will be applied on two classes of networks: the Delta network and another class that we propose which is the over-sized Delta network.

## 1.2 Contribution of this dissertation

This thesis began as a study of the performance of a novel multistage interconnection network architecture. The study of the characteristics of the proposed network lead to the proposition of a whole new class of multistage interconnection networks, that is, over-sized Delta networks. This is a class of networks that are in general more complex than normal Delta networks, but seem to be more powerful, thus the main question on which this dissertation is based can be summarized as the justification of the use of more complex networks in order to increase performance.

This study was faced with the problem of the above mentioned lack of a methodology for multi-criteria performance evaluation. Thus, while evaluating different performance metrics, we tried to find a way to evaluate different performance factors simultaneously. This resulted in a multi-criteria performance evaluation methodology that projects the performance metrics into a multi-dimensional space and transforms the problem into an evaluation of a one-dimension distance function.

Alleyne and Scherson proved in [7] that a Delta network preceeded by a random permutation stage can simulate any equal size network at the same performance. However some architectures, such as SoC systems using networks on chip, for which such a stage cannot be added at reasonable cost, are a potential application for the use of the proposed methodology. It is in choosing cost-effective networks in cases like this one that the methodology presented here is most applicable.

## 1.3 Organization of the thesis

This thesis presents a multi-criteria performance evaluation methodology used principally for the comparison of two multistage interconnection network classes. One of the classes is the Delta network represented by its subclass the Omega network, and the second networks is the over-sized Delta network proposed in this dissertation and represented by its special case the MCRB network.

Chapter 2 starts with different classifications of parallel architectures with respect to different aspects and by a brief study of conflicts in parallel machines. After this introduction, some examples of existing parallel architectures are given: SMP machines as well as recent network on chip architectures. While the presentation of these two architectures is not meant to be a survey, the communication system in these kinds of architectures is the main interest of this dissertation, thus a survey of interconnection networks is presented, with a special interest in multistage interconnection networks. Finally, the chapter ends with a review of the multistage interconnection network performance evaluation studies found in the literature. The case of multi-criteria performance evaluation is not often treated in the literature and the few examples found are either informal or not universal.

This lack of multi-criteria methodology for the evaluation of performance together with the comparison of interconnection networks in parallel computers leads to the proposition of such a methodology in chapter 3. This chapter starts with a presentation of the interconnection network evaluation problem and the multiple-criteria decision making problem. The multistage interconnection network performance evaluation is then shown to be based on multiple criteria decison making, with the possibility of dealing with the problem as one of multi-criteria optimization. This is based on the definition and the comparison of a distance function called the universal performance factor. This universal performance factor is simulation based and utilizes different measured performance factors simultaneously as a comparison factor. The simulation tool used as well as the performance metrics that are used as examples are presented.

In chapter 4, the over-sized Delta networks are proposed as an improvement technique for Delta networks. The chapter starts with an introduction to the different improvement techniques studied in the literature. The architecture of the over-sized Delta network is then presented then with an implementation example which uses the MCRB network. Some special cases of this last network are presented at the end of the chapter.

In chapter 5 the proposed methodology is applied to the proposed network class which is then compared with the Delta network. This had to be based on a mathematical validation of the simulation results. The networks are then tested according to some performance metrics in a mono-criterion and multi-criteria evaluations. Results are presented for a general case study and for two special cases: the study of the effect of the degree of a multistage interconnection network on its performance and the scalability of these networks.

In the conclusion we present an assessment of the work and discuss some future directions and perspectives that we propose as extensions to our work.

An appendix is added concerning some results obtained on the AMCRB network.

# Chapter 2

# Multistage interconnection networks in parallel computers

## 2.1 Introduction

Parallel computing, or parallelism, can be simply defined as the use of more than one processing unit in order to solve **one** particular problem. Duncan defines a parallel architecture as "*an explicit, high-level framework for the development of parallel programming solutions by providing multiple processors, whether simple or complex, that cooperate to solve problems through concurrent execution*" [29].

Communication systems play a main role in today's parallel computers, which in turn are becoming a very important utility for solving today's scientific problems which need more and more computational capability and speed. This principal role leads to the necessity of developing efficient techniques for the analysis of their performance.

Multistage interconnection networks are widely used in parallel multiprocessor systems to connect processors to processors and / or memory modules. Their popularity is due to the high switching cost of crossbar networks. Various topologies of multistage interconnection networks have been proposed and studied in the last few decades. Most of these topologies are derived from well known undirected graph topologies including mesh, star, shuffle exchange, tree networks, and cube-connected cycles, among others.

Because the data exchange between processors and memory is an important factor that drastically affects the performance of multiprocessor systems, all the studies on these topologies have the same goal, that is: how to design an interconnection network that provides the processors with maximum bandwidth and fast access to a global shared memory multiprocessor system.

Different types of problems need parallel computers with different characteristics. Therefore, a classification is needed to define different architectures having comparable characteristics. One problem of studying parallel architectures and evaluating their performance is that there is no one universal classification for their specification.

One of the most “*popular*” [82] and widely accepted taxonomies of functional environments for parallel architectures is the classification of Flynn [34]. In this classification, Flynn divided “*information streams*” into data and control (instruction) streams and then proposed a taxonomy based on the concurrent or serial execution of these streams [82]. This results in four basic classes: the SISD (Single Instruction stream, Single Data stream), SIMD (Single Instruction stream, Multiple Data stream), MISD (Multiple Instruction stream, Single Data stream), and MIMD (Multiple Instruction stream, Multiple Data stream) classes.

Basic performance criteria in SIMD environments are different from those in MIMD ones [6]. When studying the routing capacity of a communication system in parallel computer, throughput is the important performance factor in MIMD environments, while for SIMD architectures, the important criterion is the network permutation capability [90]. This is due to the fact that, in general, workloads running on different classes of machines are different. SIMD machines usually run permutation workloads. However, for the performance evaluation of a general purpose parallel architecture, both kinds of criterion can be evaluated.

In this chapter, after a small introduction to some architecture related classifications, the next two sections are devoted to those architectures needing more and more powerful communication systems. These architectures are presented as examples of parallel architectures in use today, and are not, by any means, supposed to serve as a survey. Interested readers are referred to some references treating these subjects. Interconnection networks of different varieties have been and are used as communication systems in parallel computers. Thus, a survey of interconnection networks, their characteristics, static and dynamic architectures is presented. A classification of multistage interconnection networks is proposed and some networks of special interest are discussed. Finally, before concluding the chapter, the problem of the performance evaluation of communication system in parallel computers is presented. The main interest of this dissertation is the consideration of the performance evaluation of multistage interconnection networks as a multi-criteria decision making problem. Excepting some rare, nonformal studies that are presented at the end of the section, this kind of study has not, as far as we know, been previously undertaken.

The organisation of memory in a parallel system is a key factor in its design. In fact, many design decisions depend totally on this organisation. In the following paragraph, some important basic memory organisations for parallel computers are presented.

## 2.2 Parallel architectures and memory organisation

In a multiprocessor system, also called shared memory, all processors share the same memory space. In order to allow parallel access to this shared memory, it is divided into several memory modules. The granularity of the memory system is defined by the size of the memory modules. Granularity is an essential issue when designing a parallel architecture. The sharing of the memory by the processors is provided by a medium that must be capable of transfering data among all processors and memory modules. Shared memory parallel computers are distinguished by their programming facility.

Every PE in a distributed memory system, also called multi-computer, has its own memory, and data access to another memory node is achieved by communicating with the processor connected to it.

Figure 2.1 shows the difference between the two systems. Note that in modern shared memory systems each processor has a small cache memory which is not accessible directly by other processors.

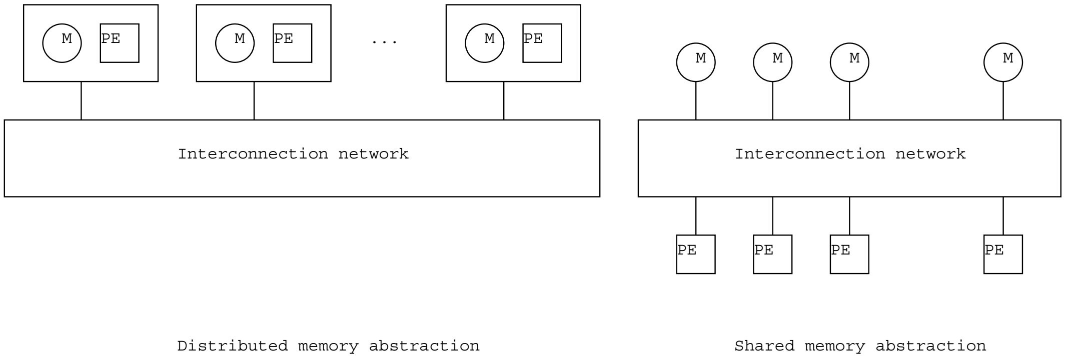

Figure 2.1: Shared and distributed memory systems. M stands for memory

In both architectures a communication medium (interconnection network) is used to connect different nodes of the system. In the distributed memory system, it links the different processors using a message passing network, and for the shared memory architecture, it connects processor to processor and / or to memory modules.

When more than one PE needs to access a memory module for a read or write operation, conflicts might take place. In a parallel system, conflicts can also occur in the communication system. The following is a brief discussion of conflicts in parallel computers.

## 2.3 Conflicts in parallel architectures

In a parallel computer communication system, a conflict occurs when more than one message tries to utilize the same communication resource. We call a communication resource a link or a Switching Element (SE) output and in a buffered communication system, an input buffer. When a conflict occurs in a buffered system, one message passes to its destination and the others are queued in order to be routed in the following cycles. In unbuffered systems, or in the case of a full buffer, in case of a conflict, only one message passes and all others are discarded and can be re-emitted later.

Three types of conflicts can occur in parallel computers [49]: Network conflicts, bank busy conflicts, and simultaneous bank conflicts. These last two types of conflict can be grouped to form memory conflicts.

One way frequently used to avoid simultaneous bank conflicts in multiprocessors is data arrangement in memory also called data skewing [16, 89, 58, 100]. This is done by allocating data in such a way that the number of memory conflicts is minimized and often results in a certain unused space in the memory. Figure 2.2 depicts a skewing of an $4 \times 4$ array into a $4 \times 8$ memory space. Assuming that the processors are located as a line above the memory system, this new alignment of the array elements enables concurrent access to lines, columns, diagonals and backward diagonals. However, it can be seen that such an alignment leaves memory holes or unused memory space.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| a(0,0) | | a(0,1) | | a(0,2) | | a(0,3) | |
| | a(1,3) | | a(1,0) | | a(1,1) | | a(1,2) |
| a(2,1) | | a(2,2) | | a(2,3) | | a(2,0) | |
| | a(3,0) | | a(3,1) | | a(3,2) | | a(3,3) |

Figure 2.2: Skewing a $4 \times 4$ array into a $4 \times 8$ memory space

The trade-off between unused space of the memory and a maximum utilization of PEs is an important issue for the programming of a parallel computer. Skewing schemes are one technique to optimize to an acceptable level the utilization of memory space and PEs [56]. Skewing elements of data structures means storing them in memory modules in an order that might be different from the original order so that they can be accessed concurrently by different processors. Lawrie defined a $\delta_i$ skewing scheme in [58]. The following is a mathematical generalization of the two dimensional scheme defined by Lawrie.

**Definition 2.** *In order to skew a $D$ dimensions table, a skewing distance $\delta_i$ must be defined for each dimension $i$ so that element $e_{d_0,d_1,\ldots,d_D}$ will be stored in memory location $\sum_{i=0}^{D} d_i \times \delta_i$.*

A skewing scheme is linear when the data arrangement function is linear.

When a memory conflict is unavoidable, consistency rules are used. One rule is the EREW (Exclusive Read, Exclusive Write) where only one processor can execute an R/W operation on the same memory bank at the same time. On the other hand, CRCW (Concurrent Read, Concurrent Write) enables more than one processor to read/write date from/to the same memory module at the same time. Conflicts on write requests can be solved by special algorithms. A practical solution is the use of the CREW (Concurrent Read Exclusive Write) mechanism.

Practical parallel systems use different techniques in order to avoid or resolve conflict problems. A list of some practical existing architectures would seem to be useful in order to have an idea about what really causes differences among machines belonging to the same family or having the same architecture model. The following two sections present two architecture families: SMP machines and MPoCs.

## 2.4 Examples of some current shared memory architectures

Today, SMP (Symmetric MultiProcessors or simply Shared Memory Processors in some references) architectures as well as their NUMA extensions are used to build almost all parallel servers. The SMP architecture model is presented in Figure 2.3.

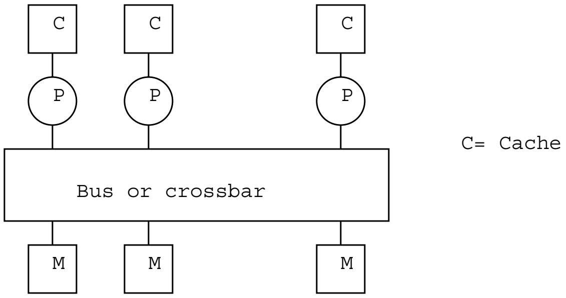

Figure 2.3: SMP architecture model, P stands for Processor, M for Memory module and C for Cache memory

Advantages of shared memory systems, such as SMP machines, include symmetry, unique addressing space, and low communication latency.

In an asymmetric parallel system, a master/slave paradigm is used. Not all processors have the same rights and capacities. While asymmetry can resolve some cache coherence problems, the unavailability of a master processor at the moment of an I/O operation may cause performance degradation or even total system blocking [1]. Symmetric systems do not suffer from such problems as there is no need for the use of a master/slave paradigm.

The fact that a symmetric system has a unique addressing space means that it has a unique copy of the OS, databases, application, etc. which gives the capacity to support "*single system image*" which leads to a scheduling and dynamic load balancing established by the OS itself.

Communications in shared memory systems are simple load/store operations. Cache coherence is controlled, in general, by the hardware. In multicomputer systems, these communication tasks have to take place among different processors, thus, they are more difficult to deal with. Such systems must also have mechanisms for the management of different versions of variables in different processors' memories.

An example of an SMP architecture performance evaluation study can be found in [73]. In their performance evaluation the authors used static simulation [70, 19, 30] and [72] to evaluate issues like instruction distribution, memory dependencies and data localisation by observing instruction throughputs.

The work of Nussbaum and Smith [73] consists of extending a previously developed model used for the evaluation of single processor computers in order to be able to evaluate SMPs. This single processor model is based on a statistical simulation. In order to construct a statistical model simulation, a detailed simulation of a benchmark was run. As a result of this detailed simulation, a stream of instructions, called a dynamic instruction trace, is generated. This trace is analyzed in order to create a statistical image for this particular benchmark. "*To carry out a statistical simulation, the tables are used to drive the random generation of a synthetic instruction trace*" [73]. Sequential consistency is assumed. Critical sections and barrier synchronizations are used.

In order to simulate the cache coherence, a probability of owning the cache line is added to the single processor model. Synchronization issues are simulated using three instruction types: *acquire* and *release* for the simulation of processor entering and leaving a critical section, and *barrier* for the barrier synchronization technique.

The main goal of the authors was not to test and compare SMPs but to prove that statistical simulation can be used in order to test this kind of architecture. Thus, the effect of the architectural characteristics of the communication systems in SMPs was not considered.

In the following some examples of the intercommunication systems of some SMP architectures are given. This is not meant to be a survey of existing architectures, and is thus not exhaustive.

### 2.4.1 Bus SMPs

These are SMP machines where the communication system is a simple or an improved bus. Two systems are presented; the SHV server board and the AViiON25000 based on the SHV server board.

The general configuration of the Intel Standard High-Volume (SHV) server board is presented in figure 2.4. Only the communication system used to connect the processors and the memory modules is depicted. This system supports hardware cache coherence. It is the result of the cooperation between Intel and Microsoft that created an operating system enabling the former to build Windows NT SHV systems with more than 4 processors [76]. At that time, NT suffered from scalability problems and the fact that it could not address more than 32 processors. In 2002, Windows 2000 had better processor and memory managing capabilities [83]. However, due to the scalability problem caused by the operating system and the traffic bottleneck on the bus, most of the systems supporting this technology are limited to 4 or 8 processors.

The AViiON25000 [35], presented in figure 2.5 is built up of Intel SHV motherboards connected by 2 counter rotating rings relaying up to 8 nodes to reach a 32 processor system (their latest system contains 64 processors).

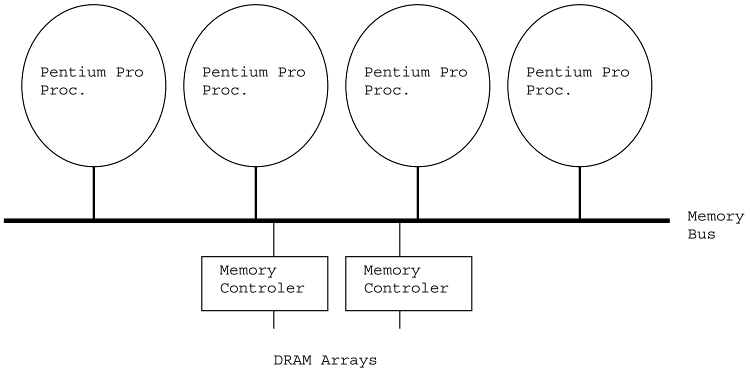

Figure 2.4: The Intel SHV SMP board architecture

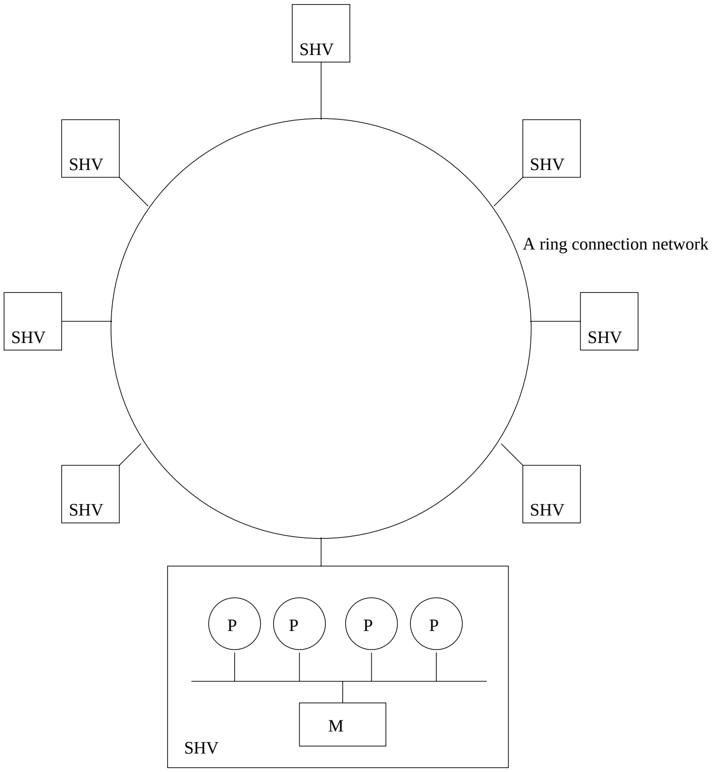

Figure 2.5: AViiON 25000 architecture

## 2.4.2 Crossbar-bus hybrid SMPs

In a bus based SMP machine with caches, a bus is used for different communication activities. In order to override bottleneck caused by the use of the bus, IBM proposed an architecture with a bus for the snoopy and a switch (crossbar) used for the interconnections among the processors as well as their communication with the memory [36]. We call an architecture using both a crossbar and a bus a hybrid system. The switch allows multiple parallel communications which are faster than those that can be routed on a bus. The use of a crossbar allows to increase the memory bandwidth by providing multiple buses.

The Compaq 8-way [2] is another example of this hybrid class of SMPs. Its architecture is presented on figure 2.6. Only one communication system is used, but it is composed of a combination of buses and a crossbar.

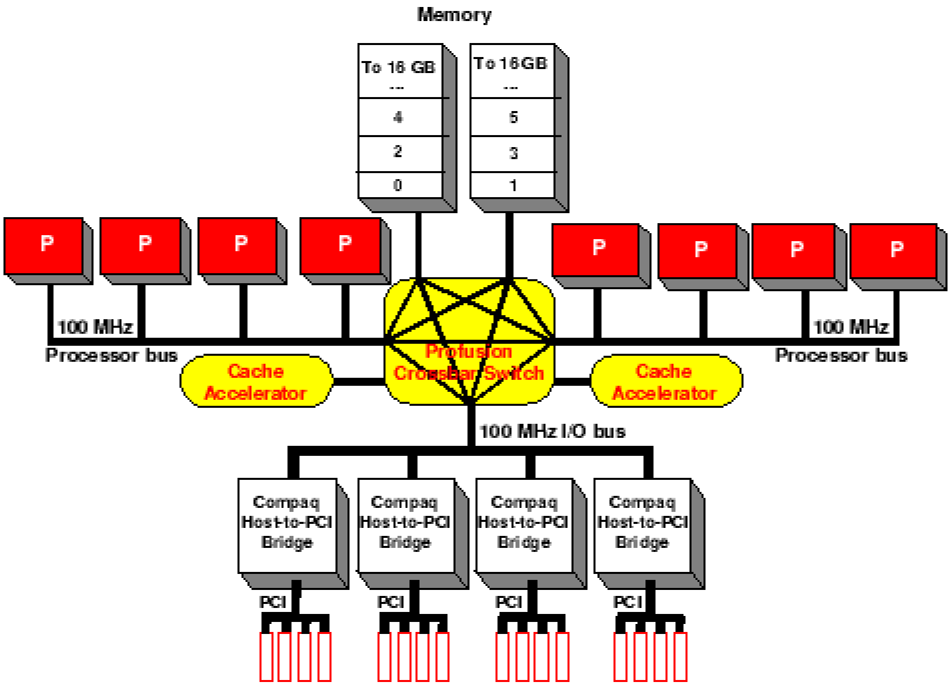

Figure 2.6: The architecture of Compaq 8-way, figure copied from [2]

## 2.5 Multiprocessors-on-chip (MPoC)

A system-on-chip (SoC) is widely defined as the integration of a complete system on only one silicon chip. It is accepted that independence is one important aspect provided by a SoC. No exterior software or hardware components are expected to interfere in the task execution of a SoC.

SoCs are an interesting possibility for the construction of computing systems because they solve a key problem from which traditional systems suffer, that is, memory latency. Even if the memory access time of a SoC is still much higher than a processor timing cycle, it is remarkably less than that of a traditional computer.

The technological progress being made in the manufacture of semiconductors and the electronic integration capability, make it possible for two improvements on simple SoCs: increasing the performance of the processors and /or multiprocessing. Both kinds of improvements are technologically possible to be realized at the time of writing.

The study of the communication system of an MPoC is a recent research branch dealing with what is called Networks on Chip (NoC) [47]. In his dissertation [10], Baghdadi listed a number MPoC architectures used for general purposes or for specific applications. In a section about the switching networks used on these systems, he listed and widely analysed the PROPHID [59], a crossbar based architecture, the SPIN [40], a Fat-Tree based architecture, as well as many other different architectures.

While buses are attractive because of their simplicity and crossbars because of their performance, neither are practical solutions for large scale computer systems. One solution is interconnection networks. They are presented in the next section.

## 2.6 A survey of interconnection networks

As we have seen in the previous section, communication between the different PEs themselves and/or communication with the memory system must be carried out by means of a medium. In fact, inter-connecting processors and linking them efficiently to the memory modules in a parallel computer is of a paramount importance. However, this task is complicated and difficult because of the complexity/performance trade-off that has to be made. The use of bus architectures, as in some of the previously mentioned examples, is not an enchanting solution as buses do not supply a satisfying level of parallelism and bandwidth. On the other hand, a crossbar, which provides a full connection between all the nodes of the system is very complex, expensive, and hard to control. Therefore, hybrid solutions have been proposed. Interconnection networks (INs) [87] constitute a good communication medium for parallel systems. In fact, an IN is a tool that restricts the paths between the different communicating nodes in order to minimize the switch complexity, while giving a certain level of parallelism which is superior to that of a bus.

Functionally, the role of an IN in a parallel system is to transfer *information* between source and destination nodes. This section begins by listing the most important characteristics of an IN from a high level architecture point of view. and then presents a small survey. An architectural classification of INs is proposed and some networks, of special interest for the dissertation, are presented in some detail.

### 2.6.1 The Characteristics of INs

In general, an interconnection network is characterized by its topology, communication (switching) strategy, synchronisation philosophy [14], control strategy, and routing mechanism [24]. In the following, informal definitions of these properties are provided.

### Topology

The physical structure of an interconnection network is defined by its topology. The topology of an interconnection network is defined mathematically by a graph $G = (V, E)$, where $V$ is a set of nodes (processors, memory modules, computers and/or intermediate SEs) and $E$ is a set of links. It is clear that the routing algorithm, which defines the path of a message to be routed between a source and a destination, depends largely on the network topology.

### Switching strategy

Basically, two switching strategies are used: circuit switching and packet switching. In the former, the whole path between the source and the destination of a message has to be reserved before the communication takes place and this reservation has to be valid until the message reaches its destination.

In the latter, a message is divided into a number of information sequences, of the same or of different sizes called packets. These packets are routed individually to their destinations. The transmission is established by steps. Only the path between intermediate nodes must be reserved at each step of a communication.

While modern telephone switching systems use packet switching [63], circuit switching was largely used for this purpose [92]. Today, modern optical parallel and communication systems use circuit switching because of technical difficulties imposed by packet switching in optical systems [75]. Improved communication strategies based on the two previously mentioned basic strategies can be found in the literature.

### Synchronization

In a synchronized interconnection network, a central clock controls the operation of SEs and I/O nodes. Handshaking strategies are needed in asynchronous systems.

### Control strategy

The control of a network can be centralized or distributed. In a centralized control strategy, a central controller must have at each moment all the information concerning the global state of the system. It will generate and send control signals to different nodes of the network according to this collected information. Obviously, the complexity of such a system increases rapidly with the increase of the number of nodes and its break-down causes the whole system to halt. Conversely, routed messages on non-centralized networks (called also self-routing) contain necessary routing information. This information is added to the message and will be read and used by the SEs [92].

#### Routing algorithm

The routing algorithm defines, depending on the source and destination of the message, the links to be used while traversing the network. Routing can be adaptive or deterministic. Paths with deterministic routing mechanism can not be changed according to the existent traffic in the network.

Before studying some examples of the architecture of INs, some definitions are necessary. These will be presented in the following paragraph.

### 2.6.2 Various related definitions

In order to consider the functionality of an IN, some classifications and definitions should be recalled.

An alignment IN is a network capable of providing access to a certain number of data structures with maximum performance.

A permutation IN is a network in which all $N!$ permutations can be realized, where $N$ is the number of inputs and outputs of the network.

An IN is characterized by its size and its degree.

**Definition 3.** *The size of a graph is the number of the vertices of the graph. For an IN it is the number of its nodes or, in other words, the number of the network's inputs/outputs.*

**Definition 4.** *The degree of a graph is the number of inputs/outputs of its vertices.*

Interconnection networks can be static or dynamic. In the following paragraphs we will present these two network families.

### 2.6.3 Static interconnection architectures

In a static interconnection network, links among different nodes of the system are "*passive*" [32] and "*only graph theoretically adjacent processors can communicate in a given step*" [15]

The simplest static network is the bus. It is evident that the use of a simple bus network is not a good choice for parallel computers as they can not transfer more than one message at a time and improved bus architectures, such as hierarchical buses, cannot afford acceptable level of parallelism. Other static INs can contain, among others, linear arrays, rings, meshes, trees, etc. In a linear array, each processor is connected to its two neighbors.

For the purpose of this dissertation, chordal rings are of particular importance. This architecture can be presented as the basis of one of the network examples to be tested later. In the following, a small review of this network is given.

A ring network can be obtained by the connection of the two ends of a linear array. This leads to a network having the same simplicity of the linear array while being capable of providing a higher level of parallelism. A much higher level of parallelism can be acquired by using bidirectional links between the PEs. Figure 2.7 shows a ring topology.

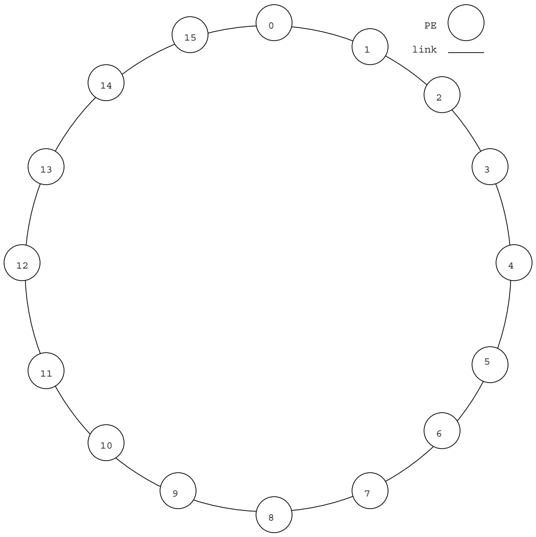

Figure 2.7: A ring IN architecture

The diameter[1] of a ring is $(N-1)$ for uni-directional rings and $N/2$ for bidirectional rings. In order to decrease the diameter of a ring, additional links may be added to every node of the network, bypassing a certain number of nodes for communication between two distant PEs. The resulting network is called a chordal ring. Figure 2.8 depicts an example of a chordal ring.

The chordal ring of this figure is irregular in nature. In general, regular topologies are easier to control, i.e. their routing algorithms are simpler. Thus, it is accepted that when talking about a chordal ring, we assume that it is regular. Figure 2.9 shows a regular chordal ring. The following definition of a chordal ring is given [77]:

**Definition 5.** *Consider an N-node ring with nodes labeled* $0, 1, \ldots, N-1$*, and ring connection going from each node* $v$ *to node* $v+1(mod\ N)$*. Let there also be unidirectional skip links or chords from each node* $v$ *to nodes* $v+s_1, v+s_2, \ldots, v+s_{r-1}(mod\ N)$*, with*

$$1 < s_1 < s_2 < \ldots < s_{r-1} < N$$

*Such an augmented ring is known as degree-r chordal ring with the skip set* $\{s_1, s_2, \ldots, s_{r-1}\}$*. For convenience* $s_0 = 1$ *and* $s_r = N$ *must be also defined.*

[1] The diameter of a network is the shortest possible path between the two most distant nodes. For a static network, the diameter is an important performance metric. The smaller the diameter, the better the network.

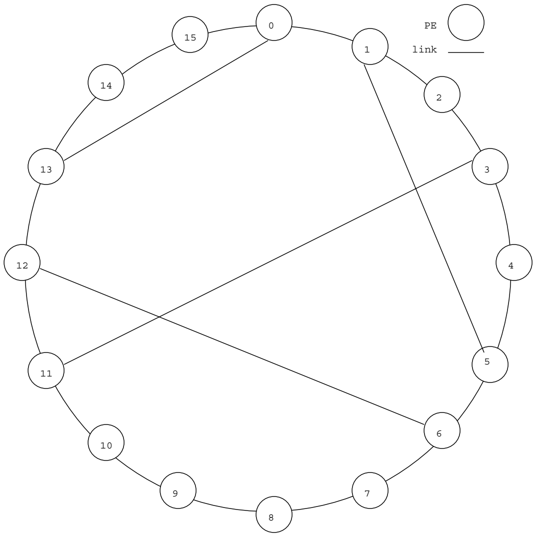

Figure 2.8: The architecture of a chordal ring

A particular case of this definition is of a special importance for this dissertation as a basis for an implementation of a multistage interconnection network. This case occurs when a chord between two nodes is expressed as a power of $r$. Formally the definition of this chordal ring is the following [50]:

**Definition 6.** *The chordal ring which forms the base of a multistage chordal ring based network implementation is a chordal ring of size $N$, with a set of chords $s = \{s_0, s_1, \ldots, s_N\}$ having the property: $\forall s_i \in s, s_i = r^i$, where $r$ is the degree of the network.*

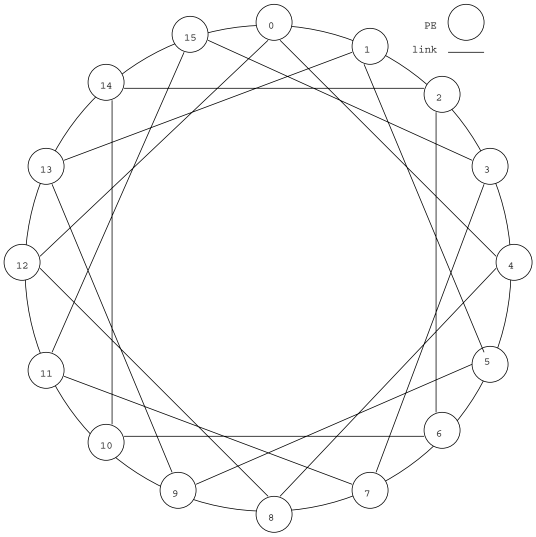

Figure 2.9: A regular chordal ring with chords of the type $s_i = r^i$

## 2.7 Dynamic and Multistage Interconnection Networks

Unlike a static network, in which the links between the nodes are passive, the linking configuration of a dynamic IN is a function of the SEs states. In other words, the paths between the graph nodes of a dynamic IN change as the SEs states change.

### 2.7.1 Single-stage INs

Dynamic INs might be built as single-stage or Multistage INs (MINs). A single-stage IN is a dynamic network composed of one linking stage and two end SE stages. Note that in some references a single-stage IN is composed of only one switching stage and one linking stage. Figure 2.10 shows a general schematic of a single stage IN. Crossbars, which provide a full connection between all nodes of the system, are considered as non-blocking single-stage intercommunication networks for parallel computers [102].

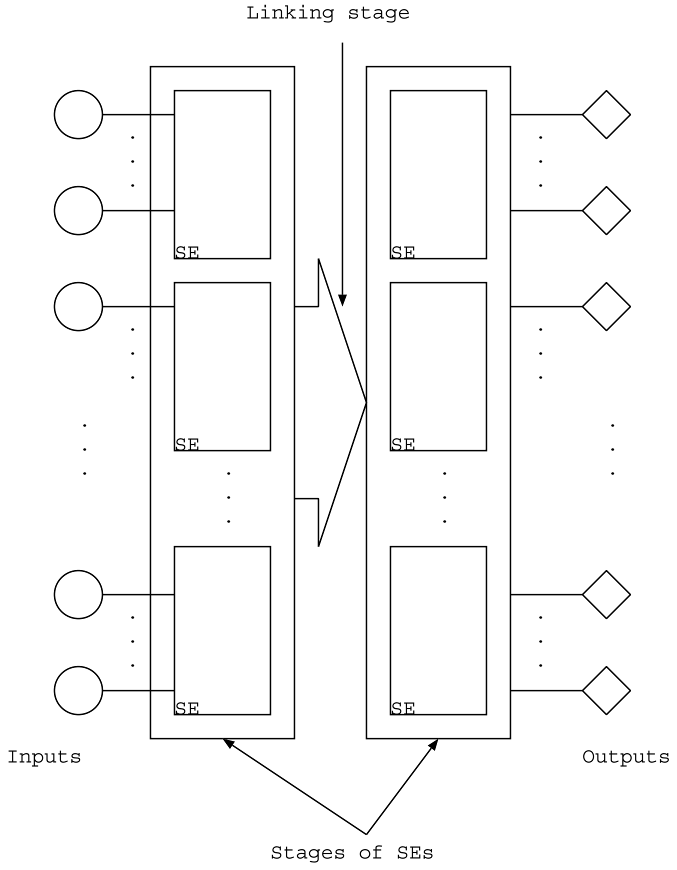

Figure 2.10: The schematic of a single stage IN

The linking stage in figure 2.10 is a permutation function connecting the outputs of the SEs of the stage furthest to the left to the inputs of the other SE stage.

In a single stage IN, more than one path through the network might be necessary to allow communication between a source and a destination. Note that not all permutation configurations can lead to a *connected* network; i.e. capable of connecting any source node to any destination. This can be clarified by the examples given in figure 2.11. In the configuration to the right, it can be proved that at most 3 cycles are needed for a message to reach its destination. For the configuration of the network to the left, some destinations will never be reached by some sources (example: a communication between SEs 0 and 1, in other words, no message emitted from 0 can reach 1 no matter how many paths it makes through the network), thus such a configuration can not be used for the construction of a single stage network. A study of such single-stage INs performance can be found in [17].

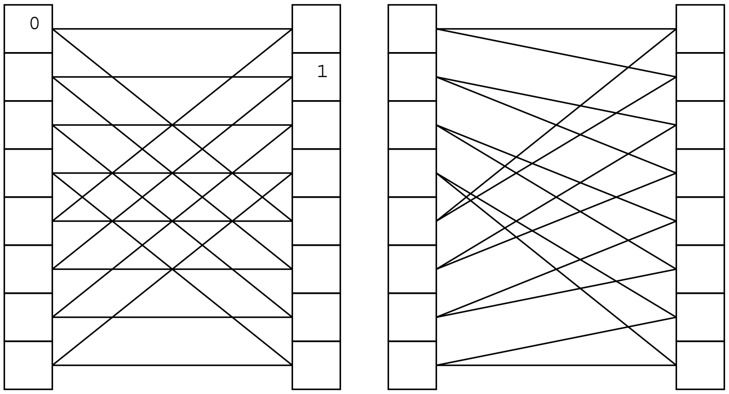

Figure 2.11: Two configurations of single stage INs

### 2.7.2 Generalities on Multistage INs (MINs)

A MIN can be defined as a network used to interconnect a group of $N$ inputs to a group of $M$ outputs through several intermediate stages of small size SEs followed (or leaded) by linking stages. These linking stages are a key factor for the definition of the network topology. Siegel [90] defines a MIN as a series of interconnection functions representing the communication pattern of the linking stage.

More formally, a MIN is a succession of stages of SEs and interconnection links. SEs in there most general architecture are themselves small size interconnection networks. The most used SEs are hyperbars [6] and more specifically crossbars. If $N$ is the MIN's size and $k$ is the SE's size, the minimum number of switches in a stage must be $\frac{N}{k}$.

Linking stages are *interconnection functions* [90], each function is a bijection of the group of the previous stage switches' address which connects all SEs outputs from a given stage to the inputs of the next stage.

In a multi-processor environment, the first stage of links is connected to the sources (usually processors) and the last stage is connected to the destinations (Memory modules). The minimum number of stages of a MIN must provide a full connection of input nodes to output nodes. Formal definitions of generalized self-routing MINs are given in [90] and [6].

A general multi-processor interconnection network block diagram is shown in figure 2.12. The SEs in MINs may have input and / or output buffers. The buffers serve as temporary storage for blocked messages when conflicts occur. In this case the MIN is called a buffered MIN. Only MINs in their simplest configuration, without buffers, are considered in this dissertation.

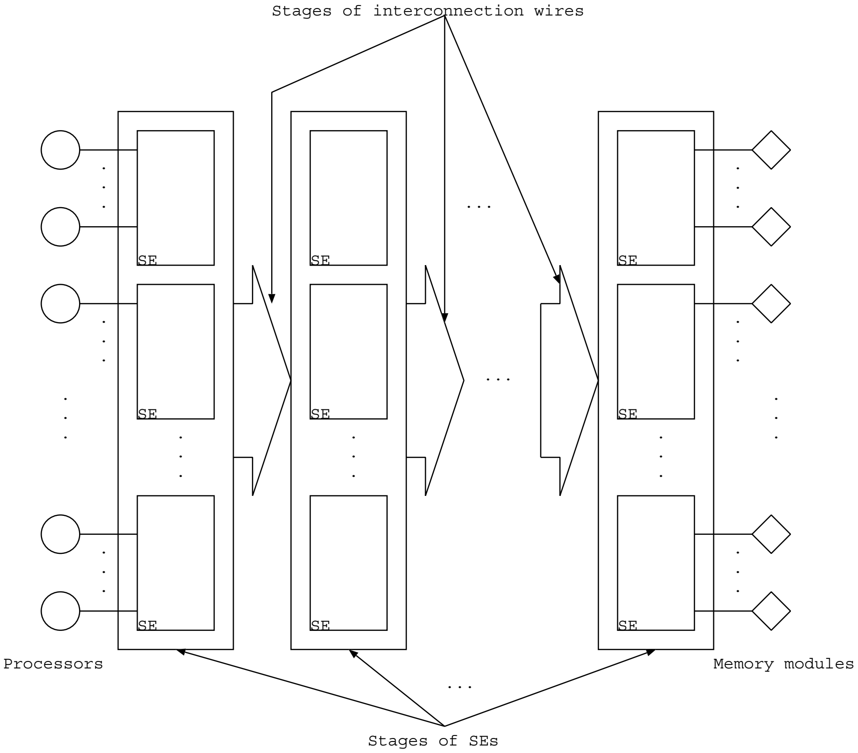

Figure 2.12: A general block diagram of a MIN

It is easy to prove that a permutation on $N = r^n$ elements needs at least $n$ interconnection functions to be implemented. This is why an $N$ size multistage interconnection network contains at least this number of stages, otherwise, some sources may never be connected to some destinations.

In general, a clock cycle is defined as the "*basic internal unit of time for the system*" [93]. For an interconnection network we can give the following definition of a cycle.

**Definition 7.** *An interconnection cycle is the time needed to end a direct communication task through a MIN. A direct communication task is one that can pass through the MIN without need to be bufferized or to be postponed because of a conflict or a faulty link.*

### 2.7.3 A classification of MINs

In the following we list some necessary definitions of MINs for the proposed classification.

One of the most important issues concerning an IN topology is the existence or absence of multi-paths. A uni-path network is also called a Banyan network.

**Definition 8.** *A Banyan network [37] is a Hasse diagram of a partial ordering in which there is one and only one path from any input node to any output node.*

In a uniform MIN, all switching elements of a stage are of the same degree, and a square MIN of degree $r$ is built from SEs of size $r$ [96]. A rectangular network is one that has the same number of inputs and outputs.

We propose in figure 2.13 a topological classification of MINs. In the following we explain each of the branches of the classification tree.

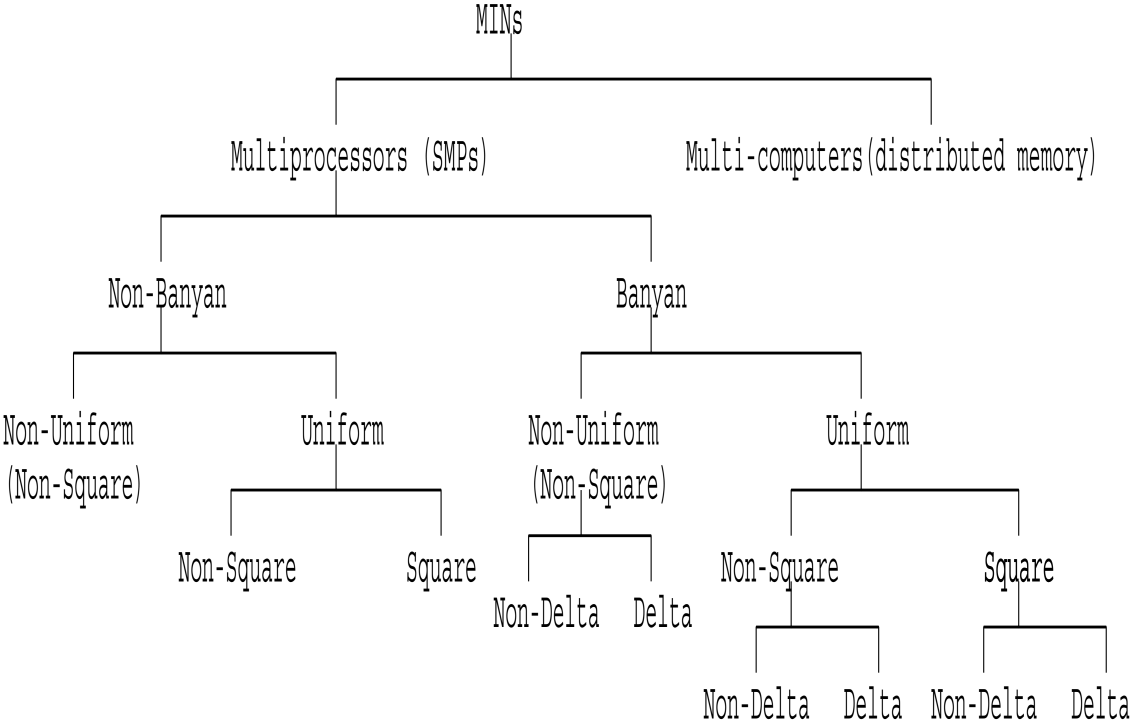

Figure 2.13: A topological classification of MINs

As stated above, MIINs can be used as communication systems in multiprocessor or multi-computer machines. In this dissertation we are interested in MINs in multiprocessor environments. Here, MINs can either be Banyan or non-Banyan.

Banyan MINs, which are the main interest of this dissertation, may or may not have the *delta property* or not. Delta networks, proposed by Patel [79], are built of $a \times b$ crossbars. The Delta property is defined as follows: let $o_i;\ i = 0, 1, \ldots, b-1$

be an output of index $i$ of a crossbar in a MIN. If an input of a crossbar in stage $j$ is connected to an output $o_i$ of another crossbar in stage $j-1$, then all its other inputs must be connected to outputs of the same index $i$ of crossbars in the previous stage. We propose the following mathematical generalization of Patel's definition of the delta property.

**Definition 9.** *For a Banyan MIN of size $N$ and degree $r$*[2]*, suppose that the switch's inputs and outputs are presented in the base $r$, of the form $d_0, d_1, \ldots, d_{r-1}$. Let the inputs and outputs of the SEs in the network have the same indexes, then digits $d_0$ of all inputs of a switch must be equal. If a stage has this characteristic then it has the Delta property.*

**Definition 10.** *A Banyan network is called a Delta network if all its stages have the Delta property. In this case, it is said that network is having the Delta property.*

Note that a network having the Delta property possesses some kind of regularity so that the network routing algorithm can be simply defined [54]. Thus, Non-Delta Banyan networks are not of interest for this dissertation.

Delta networks will be studied in more detail in the next section because they constitute one case study to be tested later.

According to the proposed classification, uniform Banyan MINs can be square or non-square. Note that considering the above definitions, a non-uniform network is also non-square.

A DSUB (Delta, Square (also called SW in [37]), Uniform, Banyan) network is a Delta network with all its SEs being of the same size.

The Over-Sized Delta (OS Delta) network, proposed in a later chapter, is an example of the DnSUB (Delta, non-Square, Uniform Banyan) class. In a network of this class, switching elements of the first and the last stages are multiplexers and demultiplexers as will be explained, and thus, all SEs in a same stage are of the same size, while different stages have SEs of different sizes.

A Delta network of a size which is not a power of 2 can also be built as a DnSUB MIN. Figure 2.14 demonstrates an Omega network of size 6.

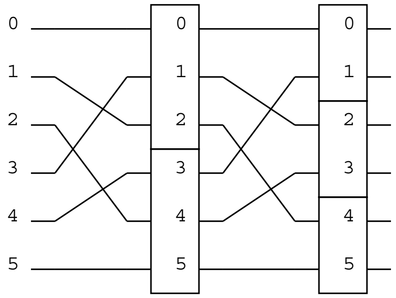

Figure 2.14: A Delta network of size 6

Non-Banyan MINs are, in general, more expensive than Banyan networks and more complex to control, still, they often are fault tolerant and capable of using

[2] In this dissertation Network(N,r) will present a MIN of size $N$ and degree $r$.

rerouting strategies to solve some conflicts that may occur in the network. Networks of this class can be constructed either by the *augmentation* of a Banyan network or by the construction of a *multipath* network such as the Clos MIN [21].

Kruskal and Snir studied in [54] two augmentation strategies: replication and dilation which are defined by the authors as follows:

**Definition 11.** *The d-dilation of a network $G$ is a network obtained from $G$ by replacing each edge (link) by d distinct edges.*

**Definition 12.** *The d-replication of a network $G$ is a network consisting of $d$ identical distinct copies of G.*

Augmented networks, when built on the base of DSUB, form an example of the SUnB MINs, while Clos network is one example of the nSUnB class.

### 2.7.4 Delta networks

In its formal definition given by Patel [78, 79], a Delta network (Delta-MIN) is a Banyan MIN built using $a \times b$ digit controlled crossbars of which no input and output ports can be left unconnected. The total number of crossbars required to construct a Delta-MIN is:

$$\begin{aligned} \sum_{1 \leq i \leq n} a^{n-i} n^{i-1} &= \frac{a^n - b^n}{a - b}; a \neq b \\ &= nb^{n-1} \quad ; a = b \end{aligned} \tag{2.1}$$

In order to simplify the construction of a Delta-MIN as well as the design of a routing algorithm, Patel proposed using a *regular* link pattern: the $q$-shuffle function.

A $q$-shuffle of the elements of a group of $qa$ elements is a permutation of these elements defined by:

$$S_{q*a}(i) = \left( qi + \left\lfloor \frac{i}{a} \right\rfloor \right) mod\ qa; 0 \leq i \leq qa - 1 \tag{2.2}$$

Furthermore, applying the q-shuffle function on a number represented in base $q$ corresponds to the application of a cyclic shift on said number. This leads to the construction of a class of MINs called "*shuffle-exchange MINs*" [95, 79, 45].

Informally, a shuffle-exchange MIN is a succession of q-shuffle linking stages and crossbar stages. Crossbar stages are defined as an implementation of an "*exchange*" function. An exchange function is defined by complementing the least significant digit of the r-based representation of the input [45].

The $q$-shuffle is the general case of which the perfect shuffle is a special case. In his paper on parallel processing with perfect shuffle [95], Stone proved the appropriateness of the perfect shuffle for a number of well know and important scientific applications, in particular FFT (Fast-Fourier Transform), polynomial evaluation, sorting, and matrix transposition [94].

Lawrie called the family of shuffle-exchange MINs Omega MINs [58] and studied the conditions and the capability of this network to allow processors to access sub-structures of interest. In the remainder of this dissertation, Omega MINs will be used to refer to shuffle-exchange MINs. Figures 2.15 and 2.16 show schematics of Omega(16,4) and Omega(8,2) respectively.

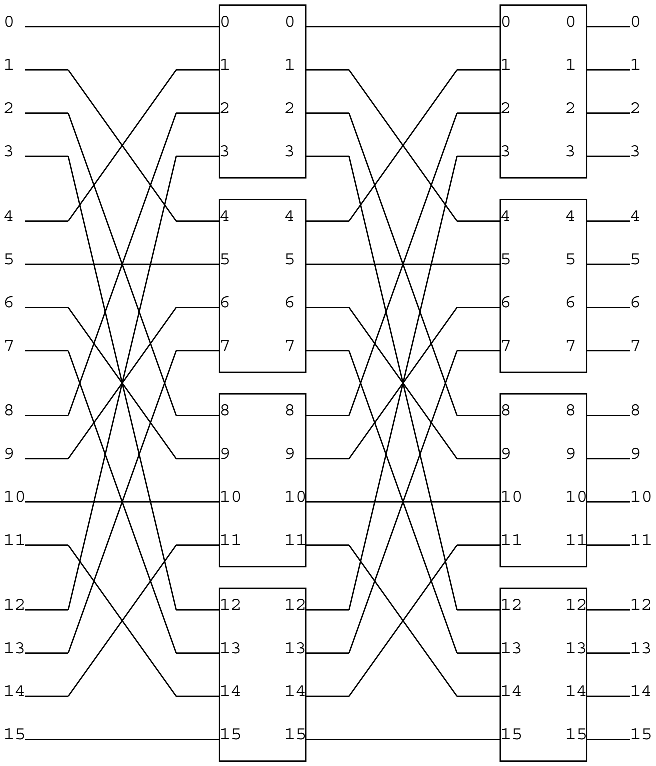

Figure 2.15: A block diagram of $\Omega(16, 4)$

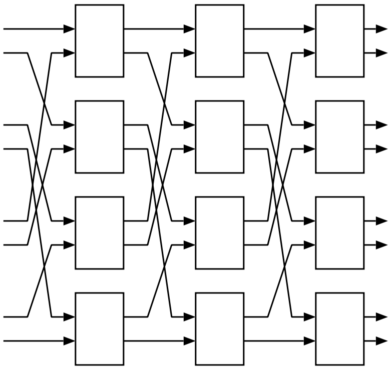

Figure 2.16: A block diagram of $\Omega(8, 2)$

The routing algorithm in an Omega MIN is very simple. The destination of a message is represented in base $r$, it serves as the control sequence to lead the message through the SEs of the network. In other words, the message will be lead to output $i$ if the corresponding digit of the control sequence is equal to $i$.

Omega MIN was effectively implemented in the NYU Ultracomputer, a shared memory MIMD parallel computer relaying thousands of processors and memory modules [39]. Some enhancements were applied on the network [38] in order to achieve a larger bandwidth and to prevent the network bottleneck due to the large number of I/O operations and to certain communication patterns.

It is proved in [28] that all Delta MINs of the same size and degree have the same throughput. Informally, a MIN is equivalent to another if by a simple modification to its architecture, such as adding a permutation stage, the two networks can implement the same permutations [81].

### 2.7.5 Clos networks

In 1952 Clos published a paper studying a non-blocking network to be used in telephone switching systems [21]. In this paper, Clos studied the non-blocking conditions of a non-square switching network of an optimal degree for a certain size.

A Clos network of size $N$, denoted $v(m, n, r)$ is built of three switching stages. The input stage contains $r$ SEs of size $n \times m$, the middle stage is composed of up to $m$ SEs of size $r \times r$ and the $r$ SEs in the output stage are of size $m \times n$, with exactly one link between every two switches in two consecutive stages [103]. Large scale Clos networks can be built recursively using smaller size networks as SEs in the middle stage, this gives the network excellent scalability. Figure 2.17 shows the $v(3, 3, 4)$ Clos network.

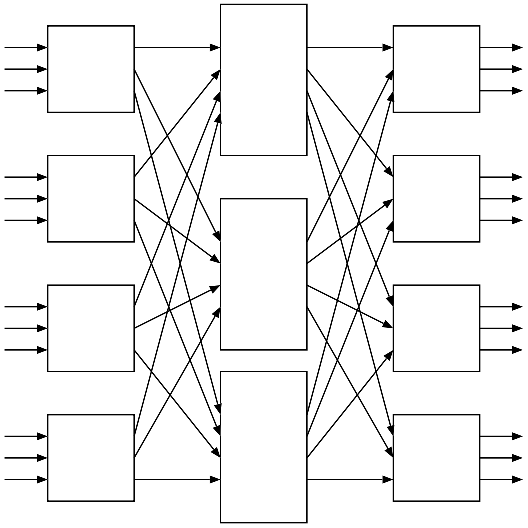

Figure 2.17: A block diagram of $v(3, 3, 4)$ Clos network

A comparison between figure 2.17 and figure 2.15 shows why a Clos network is considered to be an "*extended Delta network*". An extended Delta network is a Delta network with extra switching stages in order to transform the original network into a multipath one. In fact, a square $v(n, n, n)$ Clos network is an Omega network of $n \times n$ with an extra input switching stage.

Clos MIN is a strictly non-blocking network for $m \geq 2n - 1$ [18], i.e. if this condition is respected, two idle nodes (an input and an output terminals) can be connected no matter how many connections already exist.

Benes proved that using less complex configurations, i.e. less SEs in the middle stage, and by choosing the paths carefully, only a number of $m \geq n$ is necessary. This network is called a rearrangeable network. A rearrangeable network is a nonblocking network when suitable routing algorithms are used [13, 103]. Much research has been devoted to the study of routing algorithms. A recent example is presented in [64]. Another important study was presented by Lenfant in [60], in which the author proved that general routing algorithms for Benes MINs, which are very complex, can be simplified if the studied permutations are restricted exclusively to families of frequently used permutations.

### 2.7.6 The plus-minus $2^i$ (PM2I) network

Feng used this network, also called a barrel shifter network, in order to implement data manipulating functions (DMFs) [33]. He also presented a logic ports architecture of the SE used in it. The interconnection function defining this MIN of degree 3[3] is described as follows: the outputs of a switch $j$ in stage $i$ are connected to inputs $(j \pm x \times 2^i) mod\ N$, where $N$ is the MIN size, $x = 0, 1$ and $0 \leq i \leq N - 1$. Figure 2.18 shows the static network which is the basis of the dynamic implementation of the barrel shifter. Siegel listed the parallel architectures in which this network, or similar MINs were used [90, 43, 22, 99]. Feng claimed that this MIN is of a relatively small complexity. In fact, this might be true when using logic ports in order to build centralized controlled SEs. However, the claim will be revised when studying the distributed controlled version as a special case of the OS Delta-MIN. Note that the Illiac network is a subset of the PM2I MIN [90].

In the following section a survey of MIN performance evaluation methods is presented. The problems faced in such a study as well as the solutions proposed in the literature are presented. An indication of how the main problem studied in this dissertation, that is the multiple criteria approach, was addressed in the literature is also given.

## 2.8 MINs performance evaluation-related work

The variety of proposed architectures for intercommunication networks in multiprocessor systems makes the choice of one specific network a difficult task. There-

[3] Note that Feng mentioned also the use of PM2I MIN of degree 5.

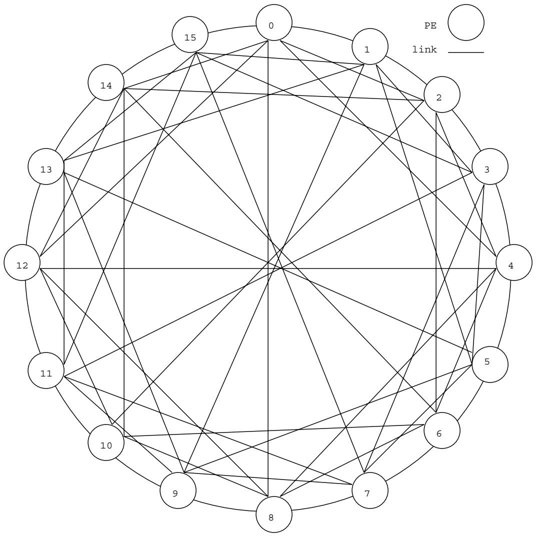

Figure 2.18: A static barrel shifter of size 16

fore, a lot of work was devoted to performance evaluation and the comparison of communication systems in parallel computers.

Basically, two tools are used to evaluate the performance of interconnection networks: mathematical (analytical) methods [79, 90, 101], and simulation [20]. However, real performance evaluation needs some kind of hybridization. Analytical models are frequently solved by simulation and simulators are usually based on some mathematical inputs e.g. the workload as well as some mathematically accepted assumptions.

Analytical models provide a certain level of performance prediction. They must be based on a certain number of assumptions. These assumptions are seldom realistic. On the other hand, solving a model built without considering some simplifying assumptions is very difficult and even impossible.

Simulation allows more flexible characterization of networks than an analytical model as it permits better control over communication patterns and routing algorithms, so, real and popular communication patterns as defined by [60, 69] can be analysed. In addition, simulation is becoming less costly in hardware, software and time terms [91].

In [91], simulation techniques are classified into four major categories: "*synthetic workload-driven simulations*" where the simulated application is represented by a simple probabilistic model, "*abstraction-driven simulations*" in which the model is a representation of the application behavior, "*trace-driven simulations*" which use results obtained from previous simulations, and "*execution-driven simulations*" where the execution of the application is simulated.

In addition to these two evaluation techniques, after effectively building a system, its *performance tuning* can be established by a third performance evaluation technique, either measurements [44] or experimentation [91]. A comparison between the three techniques can be found in the latter reference.

### 2.8.1 The problem of MINs performance evaluation

In [62] the problem of evaluating and comparing parallel computer performance was presented. The whole problem is already very clearly summarized in the overview of the paper: "*Network effectiveness differs considerably across applications and operating environments, but even if these variables were fixed, cost and performance metrics must be chosen. Scientifically determining the best network is as difficult as saying with certainty that one animal is better than another*".

Among other questions, the problem of choosing performance metrics was discussed in the paper. The authors treated the importance of the relation between the performance factors and the application the machine is supposed to run as well as the relative importance of each chosen metric in relation to the other ones. In addition, the definition of one metric depends on the application, the context and the architecture. For example, when considering a metric, its average, maximum, minimum or other describing value can be used.

### 2.8.2 Probabilistic analysis

The blocking characteristics of crossbar networks as well as Delta-MINs were analysed and compared by Patel [79]. This study is based on a probabilistic model and is aimed at evaluating the throughput of the networks[4].

Patel's study is basically based on an approximation of a stage of crossbars to a single crossbar. A justification of such an approximation can be found in [28] (lemma 2). In this study the "*probability of acceptance*" is defined as the probability that an arbitrary request will be accepted, i.e. will reach its destination.

The probability of acceptance of an $M \times N$ crossbar is given by:

$$P_A = \frac{N}{mM} - \frac{N}{mM}\left(1 - \frac{m}{N}\right)^M \tag{2.3}$$

Where $m$ is the probability that a request exists on a crossbar input.

The previously mentioned approximation allows the calculation of "$m_i$ *the rate of requests on an output line of stage* $i$" in a MIN built of $a \times b$ crossbars, which is given by:

$$m_i = 1 - \left(1 - \frac{m_{i-1}}{b}\right)^a \text{ and } m_0 = m \tag{2.4}$$

### 2.8.3 Multi-criteria MINs performance evaluation

In computer science, as well as many other scientific, economic, and industrial fields, multi-objective decision making is frequently necessary. A multi-objective

[4] The term used in Patel's paper for the "*number of memory requests accepted per cycle*" is bandwidth not throughput.

optimisation problem is one having more than one objective, each corresponding to a different optimal solution. These individual optimal solutions are often contradictory. Hence, no one optimal global solution exists and the solution of the problem would be one of a number of optimal solutions [26]. More formally, in a multi-objective optimisation problem, rather than a single optimization function, we find a vector of such functions.

For the performance evaluation of a parallel system, selecting the metrics to be evaluated is not an easy job [44]. This is due to the large number of factors that must be evaluated, their importance and relative importance as well as the non existence of standardization. This point will not be studied in this dissertation.

The majority of the work done on the performance evaluation of MINs did not consider it as a multi-objective problem. A lot of these studies evaluated throughput, latency, or other performance metrics as separate factors. In the following two examples of multi-objective evaluation are presented.

In [20] Cheemalavagu and Malek studied the effect of the degree of Banyan MINs on their performance. The throughput and delay were measured using a packet switching strategy. After defining the throughput as the total number of packets received during a time interval, and the average delay as the average time taken by a packet to reach its destination, the authors used the throughput-delay ratio as a *figure of merit*. This is defined as the ratio of throughput to the average delay. The use of this ratio was explained by the importance of both metrics, and by the fact that it gives an *indication* of the average number of "*packets being outputted by the network*". Simulations were used to measure throughput and delay.

One of the applications presented in this dissertation is an improved study of the effect of the degree of a MIN on its performance.The methodology proposed here validates the results obtained by Cheemalavagu and Malek and enables the use of more than the two performance metrics they proposed for higher degree networks. This possibility of evaluating more than two metrics needs a more formal evaluation than a simple use of a ratio. This is considered in our proposed methodology. Cheemalavagu and Malek's paper ([20]) will be further analysed later on.

Another example of a simulation based multi-objective approach to the performance evaluation of MINs is the work of Lakamraju *et al.* which presented an interconnection networks synthesis method [57]. The goal of the authors was to be able to synthesize networks satisfying "*a set of desired properties*". The technique was applied on random regular networks, which later were successively passed through filters which discard networks that do not fulfill certain criteria. At the end of the procedure, the output is a set of "*good*" networks from which one can be chosen. The important feature of this technique is that any number of desired performance factors can be considered and that it is useful "*when seeking to synthesise a network that performs well with respect to multiple performance measures*".

The technique is based on the specification of the performance measures of interest and the generation of a number of random regular networks which are then passed through a *bank* of filters. "*Each filter is associated with a performance requirement. The filters identify a subset of networks which have the desired performance with respect to the specified measures. This subset constitutes a short-list of networks from which the designer can choose*".

The filter consists of two parts: an evaluation phase which calculates the value of the associated performance metric, and the checking phase that compares measured values with a certain threshold. The output of one filter, which is a subset of the input set of networks serves as the input of the next filter. "*Filters are arranged sequentially one after the other in decreasing priority order of the measures they represent*".

This filtering technique is an interesting approach for multi-objective performance evaluation of interconnection networks. However, the mathematical non-formalization makes it confusing to use. For example, the meaning of priority order is an essential point that has to be defined. Also, different thresholds need different evaluation phases and the comparison does not give a global view about the compared networks.

## 2.9 Conclusion

The purpose of this chapter was to give an overview of communication systems in parallel architectures from a theoretical and practical point of view. Some classifications as well as some practical multiprocessor architectures were presented. A survey of interconnection networks for multiprocessor systems was also presented. The chapter included a review of research into the evaluation of the performance of interconnection networks.

In fact, to our knowledge, no efficient formal study of multiple criteria MINs performance evaluation exists. In the following chapter we present a multiple criteria simulation based performance evaluation and comparison methodology for MINs based on the definition of a distance function as a figure of merit.

# Chapter 3

# Multi-criteria evaluation and comparison methodology of interconnection networks

## 3.1 Introduction

A fair performance evaluation of MINs needs to take multiple metrics into account simultaneously. As far as we know, the performance evaluation of MINs has seldom been dealt with as a multiple criteria problem. The few existing examples either need more formalisation or the capacity to deal with more than two metrics at a time. In this chapter, the MIN performance evaluation and comparison problem is presented as a multiple-criteria decision making problem. The proposed solution to this problem is a distance function called UPF. This function can be used either for the proposition of a group of optimal solutions or to compare different networks with respect to their distance functions or UPFs. The chapter starts with a presentation of the interconnection network evaluation problem and gives a general idea of the multiple-objective decision making problem followed by a discussion of the possibility of application of this latter in order to solve the former.

The proposed methodology is simulation based. Thus, the simulation tool is presented as well as the considered simplifying assumptions. The performance metrics selected as examples for the evaluation of tested MINs are then defined before the conclusion of the chapter.

## 3.2 The interconnection network evaluation problem

As stated before, the evaluation and comparison of interconnection networks are not an easy job. A fair evaluation needs to take into account all hardware and software factors having an effect on performance. While the software environment (OS and running application) is an essential factor for this comparison, the main interest of this dissertation is the hardware related evaluation. In fact, "*even for a fixed application domain and a fixed operating environment, selecting the best network may be difficult because many cost and performance metrics could be used*" [62]. However, the performance evaluation can never be totally isolated from software issues and obtained results are related to the communication patterns routed on the network.

The performance of an interconnection network is often characterized by the latency, universality, fault tolerance, partitioning ability, permuting ability, control complexity, and cost and hardware complexity of the network and its components, throughput, scalability, dimensions, weight, power consumption, the length of buffers and delays for buffered networks, reliability, transmission time, link bandwidth, topology regularity, etc. [28, 42, 53, 55, 62, 78]. Some of these metrics will be defined later.

This dissertation studies the interconnection network performance problem stated as follows: For a certain environment, running a specific application, how can a number of MIN architectures be compared with respect to a number of performance metrics simultaneously.

In order to answer this question the problem is studied as a multiple criteria decision making (MCDM) issue.

## 3.3 The MCDM problem

Real world decision making problems are seldom single-objective and multiple objectives are often conflicting. One very famous example is the contradiction between performance and cost for almost all engineering systems, including parallel communication systems and networks. It is known that more powerful networks are more complex and thus more expensive. Thus, when multiple objectives are considered, improving one objective, the performance for example, leads to lawering of quality in the others, the increase of cost for example.

An MCDM problem refers to making decisions considering multiple, usually conflicting criteria. The output of an MCDM problem can be used to chose a solution from a number of feasible solutions, classify a number of solutions, or sort a number of solutions according to a certain order.

While single-objective decision making problems are often easily modeled and solved, MCDM problems suffer from a number of difficulties. Croce *et al.*'s paper is an important study of these difficulties [27].

The main difficulty with MCDM problems is the absence of an "*objective definition of optimal solution*" because of the conflicting optimal solutions from different criteria. In order to solve this problem, many techniques have been proposed. All these techniques consist of transforming the MCDM problem into a single-criterion one. In other words, the solution consists of finding a single value representing each single solution. While it is not always easy to find such a function, distance functions are usually used when an ideal value is known, and when the aime is to chose the most *most convenient* solution.

For a classification application, a distance function is useful to measure how close a tested system is to a certain class representation. For an optimization problem, a distance function serves as a measure of how close a solution is to a certain optimal one.

The following is the definition of a distance function.

**Definition 13.** *A function $d(p,q) \in R^{+*}$ is called a distance function between two points $p$ and $q$ if it satisfies the following conditions:*

- $d(p,q) \geq 0$;
- $d(p,q) = d(q,p)$;
- $d(p,r) \leq d(p,q) + d(q,r)$.

One particularly important distance is the Minkowski distance, it is given for an $n$ dimensions vector space by the following equation:

$$d(p,q) = \sqrt[\lambda]{\sum_{j=1}^{n} |p_j - q_j|^{\lambda}} \tag{3.1}$$

When $\lambda = 1$ the distance is called Manhattan distance, and when $\lambda = 2$ it is called Euclidean distance.

As the goals of an MCDM problem are often in contradiction, the definition of a function representing its whole state space can not give one *best* solution. In other words, no one optimal solution exits and a whole trade-off of solutions is given as a function of considered parameters. As a result the MCDM problem can be treated as a multi-objective optimization problem for which a solution *"is more of a concept than a definition"* [65]. Still, the last decision from the set of optimal solutions is a *subjective* choice. The following is basically a summary from [65].

**Definition 14.** *A multi-objective optimization problem is defined by:*

$$Min/Max(f(x) = [f_1(x), f_2(x), \ldots, f_k(x)])$$

*subject to:*

$$g_j(x); j = 1, 2, \ldots, m$$

*where $k$ is the number of objective functions $f_i(x)$ to be minimized or maximized, $m$ is the number of constraints and $x$ a vector of design variables.*

For a multi-objective optimization problem, one approach to defining the set of optimal solutions is Pareto optimality. A set of such optimal solutions is called the Pareto optimal, which is theoretically an infinite set of solutions. A Pareto optimal solution is defined as follows.

**Definition 15.** *A solution $x^* \in X$, where $X$ is the states space, is Pareto optimal iff there does not exist another point $x \in X$ such that $f_i(x) \leq f_i(x^*)$, with strict inequality for at least one index.*

In the following section, the MCDM and the multi-objective optimization will be applied to the problem of evaluating and comparing MINs.

## 3.4 Treating the problem of performance evaluation of MINs as an MCDM problem

First, the motivation behind the proposition of a methodology for the evaluation of MINs is presented and then a distance function called the Universal Performance Factor (UPF) is defined and its suitability for the performance evaluation of MINs is studied.

### 3.4.1 Motivation

The proposed methodology aims to introduce a mathematical function regrouping a number of performance metrics in order to compare MINs. The number as well as the choice of the metric is supposed to be a conceptual decision, i.e. before the use of the function, evaluated factors are chosen.

The proposed function tries to answer the following question: given a number of MINs with different architectural characteristics, how can they be compared with respect to a number of different performance metrics?

After defining the evaluation function it can be utilized in two ways: simply as a distance function used to compare different MINs, or as an optimization function aimed at defining a set of Pareto optimal networks.

### 3.4.2 The Universal Performance Factor

In this section, the proposed methodology used to combine a number of performance factors in order to get a universal performance factor is explained.

In order to define the UPF, let us suppose that the factors to be evaluated as well as the importance attended to them are a part of the designing process (i.e. the performance factors to be evaluated are chosen). In general, performance evaluation factors can be divided into two major groups: factors to be maximized and factors to be minimized. We call the group of factors to be maximized $p^{max} = \{p_1^{max}, p_2^{max}, \ldots, p_k^{max}\}$ and the factors to be minimized $p^{min} = \{p_1^{min}, p_2^{min}, \ldots, p_l^{min}\}$, where $k$ is the number of factors to be maximized and $l$ is the number of factors to be minimized.

The beginning of this study of the universal factor will concentrate on factors to be minimized only. The definition will be generalized later for the case where the two types of factors are involved. For a group $p^{min}$ of performance metrics, the universal performance factor is defined by the Euclidean distance of the performance projection value into an $n$ dimensions vector space, where dimensions represent different performance factors.

**Definition 16.** *Given a MIN and a groups of performance factors $p^{min}$. The universal performance factor (UPF) can be broadly defined by:*

$$UPF = \sqrt{\sum_{i=1}^{l}(p_i^{min})^2} \tag{3.2}$$

**Definition 17.** *Given two networks $\mu_1$ and $\mu_2$ and their UPFs, $UPF_1$ and $UPF_2$ respectively. We say that $\mu_1$ is more powerful than $\mu_2$ if $UPF_1 < UPF_2$.*

By this definition, the MINs multi-criteria performance evaluation and comparison is transformed into the evaluation of a unique function, for which the parameter of comparison is the distance between the value of the UPF and an ideal (non-realistic) network, for which the UPF value is equal to 0. In order to clarify the idea behind this definition consider an example concerning the performance evaluation of two MINsusing two performance factors $p'$ and $p''$. We assume that these two factors are both to be minimized. Figure 3.1 presents in two dimensional space the performance of the two MINs $\mu_1$ and $\mu_2$. Let $p1'$(resp. $p2'$) and $p1''$(resp. $p2''$) be the calculated values of these factors for $\mu_1$(resp. $\mu_2$). From Figure 3.1 one can notice that $\mu_1$ is more powerful than $\mu_2$ as $\mu_1$ gives smaller values than those of $\mu_2$. Note that the UPF is the length of the vector having for coordinates $(p_i', p_i'')$. One can notice that the smaller the value for UPF, the better the network performance.

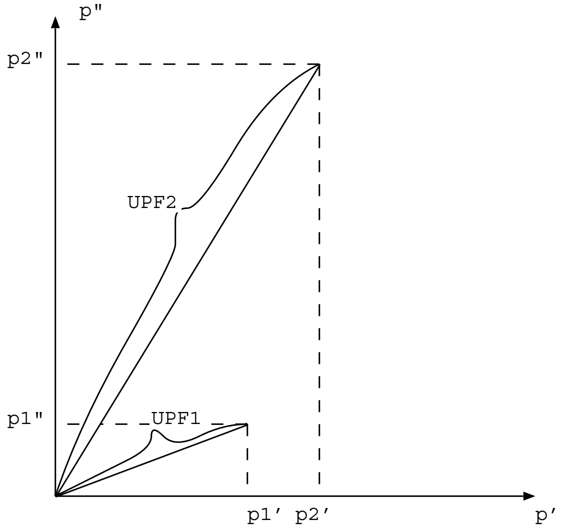

Figure 3.1: An example of the use of the UPF factor.

Usually different parameters have different types of measures and scaling. In order to solve this problem, values can be normalized to a certain value, this may be the average value, or the maximum value for each factor. The normalization procedure using the maximum value provides the possibility of giving all factors the same importance, and thus of being able to compare different scaling metrics. The equation 3.2 can be replaced by the following equation:

$$UPF = \sqrt{\sum_{i=1}^{l} \left( \frac{p_i^{min}}{MAX(p_i^{min})} \right)^2} \tag{3.3}$$

Equation 3.3 can be further improved by including the importance aspect of each factor in the design process of a MIN. This can be done by multiplying each term by a factor called the *weight* ($w$). The weight $w_i$ expresses the importance of the performance parameter $p_i$. This leads to the following equation.

$$UPF = \sqrt{\sum_{i=1}^{l} \left( \frac{p_i^{min}}{MAX(p_i^{min})} \right)^2 w_i} \tag{3.4}$$

Now, the UPF formula (equation 3.4) will be generalized to the case where both factors to be maximized and factors to be minimized are to be evaluated simultaneously. To introduce this aspect it is a good idea to note that normalizing the performance factors using the maximum value permits us to consider the maximizing of a factor as being equivalent to the minimizing of $1 - \frac{p^{max}}{MAX(p^{max})}$. This leads to the following final formula for the UPF:

$$UPF = \sqrt{\sum_{i=1}^{l} \left( \frac{p_i^{min}}{MAX(p_i^{min})} \right)^2 w_i + \sum_{j=1}^{k} \left( 1 - \frac{p_j^{max}}{MAX(p_j^{max})} \right)^2 w_j} \tag{3.5}$$

Two conditions have to be considered in order to use the UPF as a performance factor. First, it is assumed that $MAX(p) \neq 0$. Second, the measured factors are assumed to be inter-independent.

Defining the UPF, measurable metrics that it can take in account have to be evaluated. The evaluation of these factors is done by the simulation of the networks and the routing of certain communication patterns. The simulation tool is presented in the following section.

## 3.5 The simulation tool

Considering the complexity of a detailed mathematical model of a general purpose multiprocessor architecture for a multi-criteria evaluation and comparison, the impossibility of the construction of a general purpose mathematical model of a generic MIN, as well as our initial purpose of concentrating on the functionality of the MIN isolated from any other software or hardware factors, a simulation technique was privileged. Even if such a testing technique is not perfectly accurate, it remains a very useful tool especially for our purpose of comparing MINs functioning in same contexts.

While in a statistical simulation [72], information about the application running on the tested machine is collected by a detailed simulation to be used later in the testing phase, the use of a similar technique based on the utilization of special well known and defined frequently used workloads presented as permutations was preferred. Also, rather than using benchmarks for which testing and simulation will give results applicable only to similar programs [72], real permutation workloads that are met frequently in real word parallel computations are used. Therefore, we will call our simulation tool a semi-statistical simulator.

### 3.5.1 The Multidimensional Queue Management (MQM) simulation approach

We give here some essential definitions related to simulation and simulators [31]. Generally, simulators can be **discrete** or **continuous**. Input variable values in the discrete systems can be changed only in discrete intervals, while in the continuous systems, these values can be changed at any time. In addition, a simulator is **dynamic** if the input variable values can be changed during the simulation and **static** if these values cannot be changed. Finally, the simulation is **random** if at least one of the input variables is random. Otherwise, it is **deterministic**.

The simulator used is a discrete static one used to simulate message circulation on MINs. The general block diagram of the simulator is given in figure 3.2. We explain here the most important elements of this block diagram.

Besides the definition of the architecture, the simulator takes as inputs: $N$ the network size, $r$ its degree, the switching strategy (circuit or packet), and a number representing the message generation mode. Two generation modes are available: Manual mode, in which the number of messages to be routed is entered before the sources and destinations of these messages. Inputs in the automatic mode are the message workload that represents the number of processors generating messages during each cycle and the number of times (cycles) the simulation will be tried. When this latter mode is chosen, a communication pattern, such as one of the frequently used permutations defined by Lenfant [60] is tested.

The simulator gives as outputs three tables: the passed messages table, the conflicts table, and the non-reroutable conflicts table. Each of these three tables contains the corresponding information of each SE in the network. For example, for an MCRB(64,2) network, three $64 \times 6$ tables will be constructed. Each element in the passed messages table represents the number of messages passed by the corresponding SE. The Conflict table's elements are the numbers of conflicts produced by the different SEs, and finally, the number of conflicts that SEs could not reroute is presented by the non-reroutable conflicts table. This last table is used with networks possessing rerouting strategies to solve conflict situations.

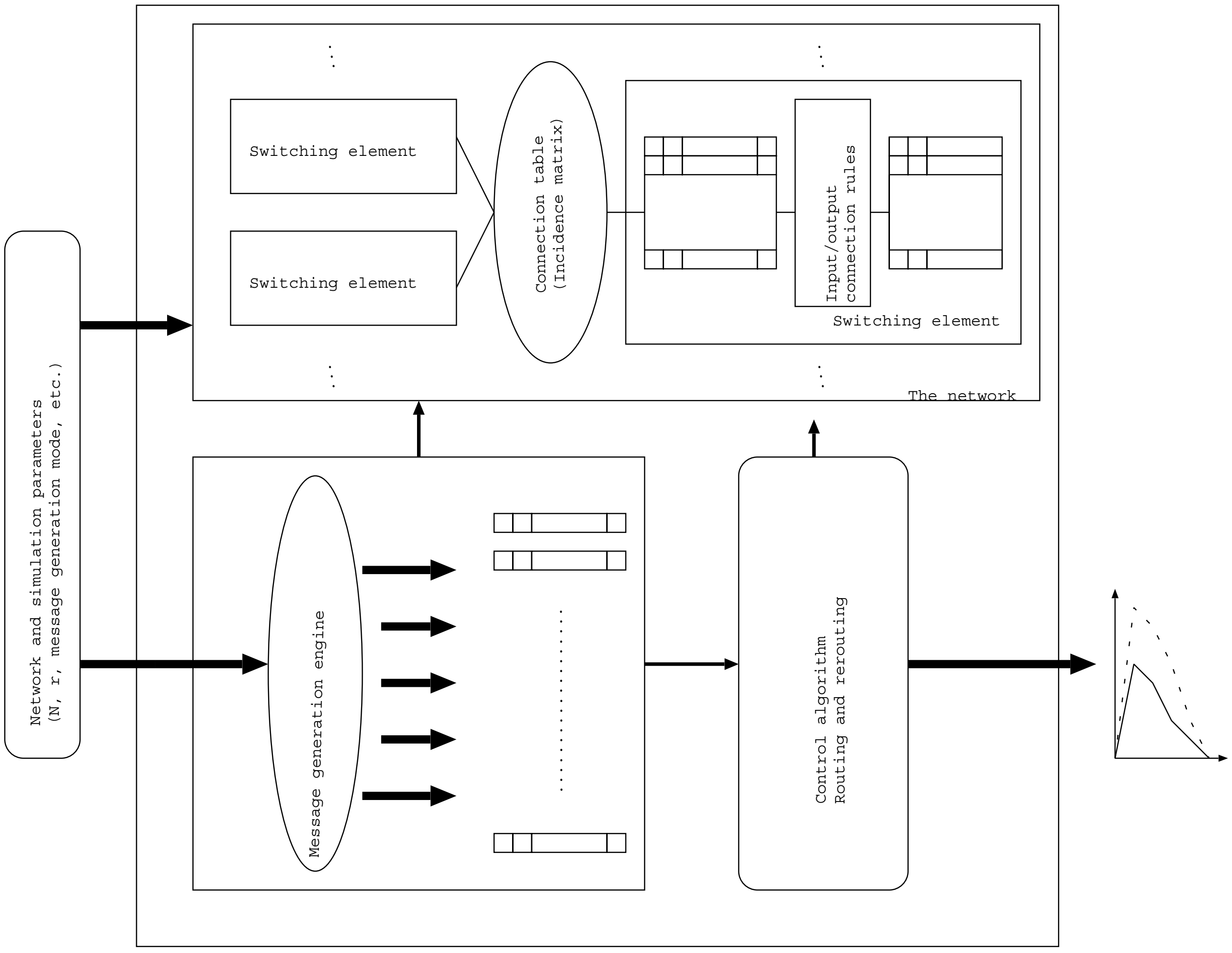

Figure 3.2: The simulator block diagram

Given the input information, the simulator constructs the network, which, naturally, is a table of SEs and a table of connections between them.

A SE is an object in the sense of the software definition of the term, which normally has two tables of messages of size $1 \times r$. Each element of a table corresponds to a link between a switching element and a switching element existing in the previous or in the next stage.

A relation between the two tables can be defined. In our study the SE cases examined are crossbars, however, other relations can be defined. A place in an output table must be empty in order to be occupied by a message. The case where a message tends to occupy a previously occupied place is called a conflict case. If the network has a rerouting technique and the messages correspond to its conditions, the rerouting algorithm will be applied. If not, one of the messages will be destroyed.

Note that tables can be easily modified into buffers in order to study buffered multistage interconnection networks performance. Only unbuffered networks are studied in this dissertation.

After constructing a table of switching elements, a table of connections must be constructed. This table is the numerical representation of the network's topology. The connection procedure consists of linking up output buffers with input buffers in the next stage, in other words, creating a table of switching element positions in the stage $i + 1$ linked to switching elements in stage $i$. This table is used to guide the message to the right switching element in the next stage.

A message is defined, mainly, by its source and destination. In our simulation technique, a message is an object which, using the definitions of the simulated interconnection network, calculates its control sequence.

Here, the engine of the simulator is defined. According to the message generation mode, the simulator creates the messages and puts them in the corresponding input buffers of the network's input stage. The routing process consists of transforming existing messages from input buffers to output buffers in switching elements, and from the output buffers of a stage to the input buffers in the next stage via the connection tables. In fact, from the simulation engine point of view, a multistage interconnection network is, a multidimensional system of input buffers or queues. A first dimension will present a place in a buffer, a second one presents the switching element in a stage, and a third dimension corresponds to the stage number.

Because the graph that gives rise to any multistage interconnection network lies in a space of certain dimensionality, we dub this system a multidimensional queuing system. The management of the queues in this space defines the efficiency of the corresponding network [86]. Figure 3.3 shows schematic diagram of the MQM system.

What will happen when starting the simulation, is that the simulator will look for messages in the input buffers of the switching elements in the network's input stage, starting with the first one, and continuing one by one. For a packet switching strategy, if a message is found, the simulator will try to route it. Routing a

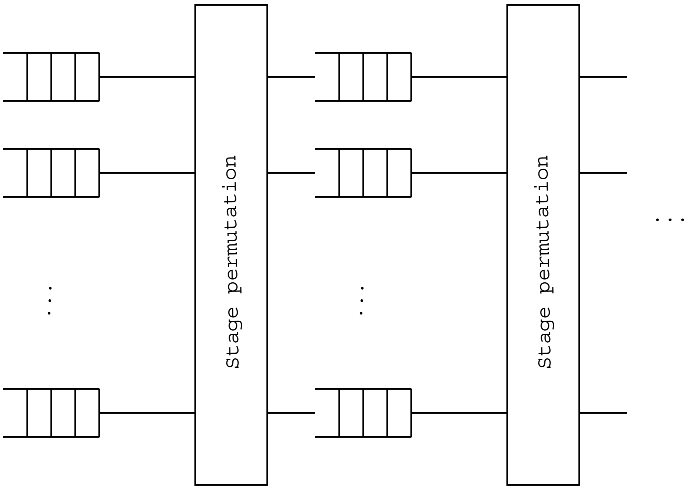

Figure 3.3: Multidimensional queuing system

message corresponds to puting it in the output buffer of the switching element. If the place where it is to be outputted is empty, the message will be placed there. If it is not, and the network has a fault tolerance strategy that can be applied to one of the messages in conflict, this strategy will be applied and the message will be placed in another place in the output buffers table of this switching element.

When Finished placing all possible routed messages in the output buffers of the input stage, the simulator transfers them via the connecting table to input buffers of the next stage and restarts the procedure. This continues until the last stage of the network.

In the case of a circuit switching strategy, whole paths from source to destination of each message are reserved one by one.

The MQM simulation approach is used in order to compare the performance of different size and degree MCRB and Omega networks. In order to do this, SEs with connection tables, message structures, and control strategy for both networks must be defined.

The switching element in our current definition of the MCRB network and in that of Omega network used as compared networks is a crossbar. This means that no control strategies are to be defined for the SE, and every input buffer in a SE has a direct connection to all output buffers. Connection tables for both networks must reflect the networks' topologies.

A simulated message in the MCRB network is composed of $n+3$ digits, where $n$ is the control sequence number of digits, or in other words, the number of network stages. The block diagram of an MCRB network message is given in figure 3.4.

The performance of considered MINs is to be evaluated with respect to permutation workloads. Analytical studies proved that the simulation of completely random permutation is not efficient for the performance evaluation of MINs

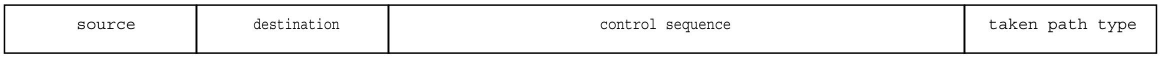

Figure 3.4: Block diagram of a message

[60, 7]. The following paragraph explains the simulated workload.

### 3.5.2 Simulated workload

Among the main characteristics of the simulated workload is its ability to represent real workload and it must be easily to be produced, reproduced and modified [44].

It is important to note here the difference between commercial and scientific applications. Commercial applications use large amounts of data shared among many users or programs. This leads to a small amount of data locality which means high level data traffic between the different system nodes. In addition, while commercial applications usually treat strings and integers, scientific applications are employed on floating point data and use loops more frequently [51].

In order to define efficient routing algorithms for Benes networks, Lenfant defined a certain number of permutation families used frequently in scientific applications. He proved also that these *useful* permutations form a small set of all *possible* permutations [60]. These permutation families were called FUB (Frequently Used Bijections).

Nassimi and Sahni proposed a permutation family called Bit-Permute-Complement (BPC) permutations [67]. This forms a general case that contains a number of the families defined by Lenfant [68].

**Definition 18.** *A BPC permutation [67] can be defined as a vector* $A = [A_{p-1}, A_{p-2}, \dots, A_0]$ *where*

1. $A_i \in \{\pm 0, \pm 1, \dots, \pm(p-1)\}, 0 \leq i < p$
2. $[|A_{p-1}|, |A_{p-2}|, \dots |A_0|]$ *is a permutation of* $[0, 1, \dots, p-1]$

*Considering the binary representation of a source, the result of applying a BPC permutation will be a number for which the binary representation is* $r_{p-1} r_{p-2} \dots d_0$ *and for* $i = 0, 1, \dots, p-1$

$$d_{|A_i|} = \begin{cases} m_i \, if \; A_i \geq 0, \\ \overline{m_i} \, if \; A_i < 0 \end{cases}$$

*and* $\overline{m_i}$ *is the one'th complement of the considered bit.*

For the simulation tool, random BPC permutations are to be generated with a specific seed. This seed is useful for the reproduction of the permutations in order to test more than one architecture with the same set of tests.

### 3.5.3 Simplifying assumptions

As stated earlier, the performance evaluation of MINs need some simplifying assumptions. The following assumptions were considered for the construction of the simulator.

1. The uniformly distribution of the requests on memory modules.
2. The independence of memory requests. This means that in case of conflict, only one message passes and all other messages are discarded and will not be reissued in next interconnection cycle. It is claimed by Patel [79] that the difference of evaluation resulting from the omission of this assumption can be ignored.
3. Processors can generate at most one request at a time. Requests are randomly generated BPCs and thus, are uniformly distributed on memory modules.
4. A request is a generic message containing addresses of source and destination nodes. No data is contained in it.
5. The only requests considered are "read / write" requests. This means that the only messages considered by the simulation are those leaving processors in the direction of the memory modules. This assumption is possible as practically all parallel systems based on MINs have two separate networks for read / write requests and read responses.
6. Memory modules are fast enough to accept a request per cycle [61]. In other words; Memory access time is equal to one processor read cycle time. This allows us to concentrate on the role and performance of the interconnection network.
7. SEs do not have buffers. Messages that can not be rerouted are destroyed.

The simulation approach as well as the simulation tool will be validated later. The following section defines in some detail some performance factors. These performance factors will be the evaluated factors for the test of the networks. These metrics were chosen as in the literature they are the most often considered when evaluating MIN performance. However, in this study, they are evaluated simultaneously using one function.

## 3.6 Some selected performance factors

### 3.6.1 Complexity

When studying a MIN, the first evaluation to be undertaken is that of hardware complexity. The hardware complexity of a MIN can be calculated using two means: the number of connection points and the number of links or wires needed to construct the MIN. Liu [63] defines the hardware complexity of a MIN as the maximum of the two values. The hardware complexity of a MIN in term of crosspoints is equal to the total number of crosspoints of all crossbars used to build it. The complexity in terms of connections is the sum of the links or wires in all stages.

**Definition 19.** *consider a MIN of size $N$ and degree $r$, that has $X$ stages of $x$ SEs each. The stages are connected with $Y$ inter-stage links. The integration complexity of the MIN will be defined as $C = max(r^2Xx, Yr)$.*

As in [21], crossbar complexity is considered as a complexity upper limit. This is explained by the *perfect passability* of a crossbar: A crossbar is capable of passing any permutation in one cycle.

Because the crossbar has the highest complexity, all MINs of complexity higher than the crossbar are generally excluded. The crossbar network is non-blocking as it has the capacity to connect directly any input to any output in one cycle [18, 20, 21]. However, the construction of larger size crossbars is very expensive in term of cost and technology. This is why less complex inter-communication means, such as MINs, are studied.

### 3.6.2 Throughput

This is defined as the number of messages delivered to their destinations per unit of time [66, 79]. Many analytical studies of MIN's throughput can be found in the literature [78, 79, 53, 96]. Simulation is used frequently when more realistic results are needed. It allows more flexibility in network characteriZation in order to make it possible for the network to analyse real-world and popular communication patterns. In fact, in order to study the throughput of an unbuffered network, messages leave sources en route to their destinations and in the case of a conflict, only one message goes through and the others are discarded. The throughput is calculated as the ratio of the number of messages having arrived at their destinations over a certain number of trials to the number of initialized messages.

**Definition 20.** *In an unbuffered MIN, We define the throughput as the number of messages delivered to their destination per unit of time knowing that only one message goes through when more than one message is assigned the same interconnection source. All other messages are discarded. In this dissertaion throughput will be calculated as the*

### 3.6.3 Universality

Another important performance parameter is the network's universality, which is defined below. The network universality analysis depends directly on the maximum number of cycles needed to route a certain number of permutations to their destinations. We use the same previously explained simulation to measure the analysed MINs' latency.

**Definition 21.** *The universality of a MIN is defined by the number of network cycles needed for all messages of a permutation to arrive to their destinations.*

### 3.6.4 Permutation capability

The permutation capability of MINs has been studied in depth in the literature, examples: [4, 96].

Permutation capability [4] refers to the MIN's capacity to route message permutations. By message permutations we mean groups of messages of which destinations are permutations of all inputs.

In order to study the permutation capacity of a MIN, one might try to route a certain number of random permutations on the network and calculate the number of cycles needed to route all message permutations to their destinations. Analytical studies [60] as well as our experience proved that routing random permutations is not efficient for the permutation capability study. Therefore, frequently used permutations [60] must be used for such an analysis. To establish this comparative study, a certain number of these permutations can be routed under the assumption that the accumulated number of permutations routed per cycle can be a comparison factor between them.

**Definition 22.** *The permutation capability of a MIN is a factor that measures the number of message permutations that can arrive in their totality at their destinations in a certain number of interconnection cycles.*

### 3.6.5 Scalability

In the literature, one can find many definitions of scalability. Many publications aime to present surveys, e.g. [25]. In fact, while it is common to rely cost, resources capacities and performance in order to define scalability, the non-existence of a standard definition of the performance of a parallel computer makes it difficult to formalize a standard definition.

A general informal definition of scalability can be found in [25] where it is defined as "*the change of performance measures of a system as particular characteristics of that system are varied*".

Some authors define scalability as the capability of adding nodes to the graph of the network, without "*drastic changes in such properties as diameter or average node-pair distance*" [57]. Small increases of size must be supported by a scalable interconnection network. However, other authors [71] estimate such a definition to be inaccurate as structural characteristics of different size systems are not the same.

More formally, the scalability of a parallel system is defined by a relation between the size of the machine on one hand, and the performance and the cost on the other hand [46]. In other words the performance of a parallel machine must increase with the an increase of size, and its cost must decrease with a decrease in size. Another aspect of scalability discussed in [46] is that of compatibility, this implies the use of the same components when some part of the system is to be upgraded.

Hwang and Xu treated scalability from three different points of view: resource, application, and technology. The main focus of their paper is resource scalability and application scalability. Resource scalability refers to the performance improvement due to an increase in the system's resources such as the number of processors, the size of memory, etc. In particular, size scalability is defined as the "*maximum number of processors a system can accommodate*". This is different from scalability in machine size, classified by the authors under the application scalability dimension. Scalability in machine size measures the improvement in performance relative to the increase in system size.

Hwang and Xu discussed three parallel system scalability metrics. They are based on maintaining a fixed efficiency, per-processor speed, and utilization.

After defining asymptotic speedup as the best speedup of an algorithm that can be obtained as a function of the size only, scalability is defined in [71] as the "*fraction of the parallelism inherent in a given algorithm that can be exploited by any machine of that architecture as a function of problem size*", This is mathematically represented as the ratio of the asymptotic speedup of the tested machine to the speedup of an ideal EREW PRAM (Exclusive Read Exclusive Write Parallel Random Access Machine). This leads to the following mathematical definition of scalability:

$$\Psi(s) = \frac{T_I}{T} \tag{3.6}$$

Where $T_I$ and $T$ are the execution times on the ideal, and tested architectures respectively.

Royo *et al.* proposed user-oriented scalability evaluation methods [85]. User-oriented scalability evaluation is more useful for the user than for the system architect. Their methodology is based on the definition of a factor $\Delta F(p,m)$ representing the increment of the studied scalability metric $F(p)$ as the size of the system increase by $m \times 100$ times.

$$\Delta F(p,m) = \frac{F((1+m)p) - F(p)}{F(p)} \tag{3.7}$$

And the scalability is defined by the function $H(p,m)$:

$$H(p,m) = \frac{\Delta F(p,m)}{m} \tag{3.8}$$

Scalability is then calculated for a fixed-problem size, with time or memory constraints.

## 3.7 Conclusion

In this chapter, the evaluation methodology for the performance evaluation of MINs is presented. The problem is considered as a multiple-objective decision making issue that can be solved either as a Pareto optimal optimization problem or by the comparison of the value of the distance function of each value. Information about the used simulation tool as well some example metrics were presented.

This methodology is used to compare different MINs with different architectural characteristics. One of the example networks is the over-sized MIN, which is proposed and studied in detail in the following chapter.

# Chapter 4

# OS Delta MINs

## 4.1 Introduction

The original idea behind the proposition of a new performance evaluation methodology for MINs was the evaluation of a recent MIN proposed in [50] called MCRB. It is proved that this network belongs to the family of improved Delta MINs, which we introduced, i.e. over-sized Delta (OS Delta) networks. In this chapter some improvement techniques that can be applied on Delta networks are presented and then the over-sized Delta networks are proposed. The MCRB network is defined as an example of an over-sized Delta network, and its characteristics are presented. Finally an improved version of the MCRB network is presented before the conclusion of the chapter.

## 4.2 Previous work on the improvement of Delta-MINs

Delta MINs are very attractive networks for the communication systems of parallel machines. However, improvement techniques have been proposed in order to increase the performance of the network, without sacrifying too many of its characterizations such that of simplicity.

One example of an improvement on Delta-MINs is the rearrangeable Benes network previously presented in chapter 2. Benes networks are considered to be improvement on Delta-MINs as they have an architecture very close to the improved version of Delta-MINs proposed in this dissertation, that is, OS Delta-MINs. Some of the improvement techniques are listed below.

### 4.2.1 Augmentation

The term *network augmentation* was used by Kruskal and Snir in order to group together two techniques used to improve the performance of a Delta network

[53, 97]. These techniques are known as $d$-dilation [88] and d-replication [37].

A $d$-dilated Delta network is a Delta network as defined by Patel[1] with the links being divided in space. This can be established by replacing each single link in the MIN by $d$ links.

A $d$-replicated network is built by the superposition of $d$ copies of the network with the inputs and outputs of all the copies connected.

This kind of augmentation considerably improves the performance of the network "*without sacrificing much of its structure.*" [53]. A replicated network is presented in figure4.1

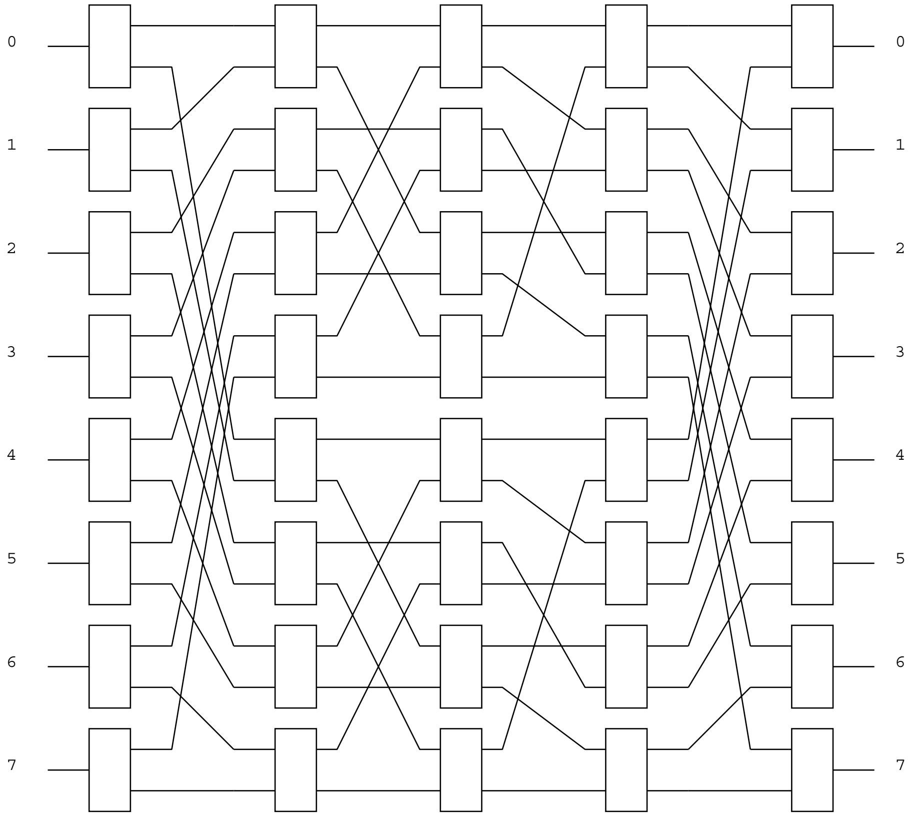

Figure 4.1: A 2-replicated Delta(8,2)

### 4.2.2 Bufferization and Bypassing

Buffered Delta MINs have been proposed in order to solve conflict problems. SE outputs in buffered MINs have FIFO type buffers where messages can be stored when a conflict occurs and in the following cycle the message continues its path to its destination. This MINs family was first analysed by Ramachandani [84]. The use of buffered SEs in interconnection networks leads to a larger bandwidth and throughput comparable to crossbars [28], with a hardware complexity comparable to that of unbuffered networks.

[1] See chapter 2.

Dias and Jump [28] studied the performance of buffered Delta-MINs and compared it to the performance of unbuffered Delta-MINs as well as to the performance of crossbars. It was shown that buffered networks have better performance with respect to throughput and turn-around-time[2]. Note that cost, and software and hardware complexity issues were not discussed by the authors.

Even if buffered MINs are not more complex with respect to our definition of complexity than buffered ones, the hardware needed to build them is considerably more expensive than that of unbuffered networks. On the other hand, unbuffered switches do not suffer from time dilation due, in buffered networks to message waiting time in case of a failing link or a conflict [80]. However, other different techniques have to be considered in order to re-send this type of message in unbuffered networks.

In fact, this dissertation is limited to unbuffered MINs as it proposes a comparison methodology among MINs, and buffers or other performance improving factors can be considered when needed.

### 4.2.3 Optical MINs

Using optical MINs helps to increase the communication speed between processors and memory modules in a parallel system. However, the input and output interfaces of optical networks are still in practice, electronic, which largely limits their effective speed [98].

Optical MINs are similar to electronic ones in many aspects. However, optical MINs suffer from new problems that do not exist in electronic networks, for example crosstalk and path-dependent loss [75].

The Cray C90 can be cited as an example of a computer using an optical communication system [104].

Note that in order to further improve their performance, modern optical MINs can be dilated in order to resolve problems such as crosstalk [74].

## 4.3 The OS Delta-MIN architecture

In this section we propose a new Delta-MIN improvement technique, called over sizing. The definition of an over-sized network is given and its characteristics are studied using an example from this family called the MCRB network.

[2] Turn-around-time is defined in [28] as the "*average time interval between the time a packet is placed in a buffer at a network input link and the time at which it is placed in a buffer at a network output link.*"

### 4.3.1 The OS Delta-MIN topology

There are two main differences between Delta networks and over-sized Delta-MINs. First, SEs of the first stage in the former are $1 \times r$ and those at the former are $r \times 1$ switches. The other difference is that an OS Delta is composed of more than one copy of a Delta network connected together with a linking stage having the Delta property. Here is a formal definition of the OS Delta-MIN.

**Definition 23.** *We call an over-sized MIN of size $N$ a Banyan Delta MIN composed of more than one copy of a Delta MIN gathered together by an interconnection stage having the Delta property. Furthermore, SEs in the input and output stages are both demultiplexers and multiplexers.*

Figure 4.2 shows an OS Delta(8,2) network. While the resulting architecture is still a Banyan Delta network, it offers a more appropriate distribution of the workload on the SEs of the network than a simple Delta-MIN. However, this architecture is more complex than that of a simple Delta-MIN, and the question that must be answered is whether the trade-off between complexity and performance is acceptable, in other words, does the performance of the OS Delta-MIN compensate for this increase in complexity.

the comparison between this figure and figure 4.1 gives an idea of the similitude of the two networks.

### 4.3.2 The MCRB-MIN: an example of an OS Delta-MIN

The MCRB network was initially proposed by Kechadi. For complexity reasons that will be discussed later, the definition given here of the network is somewhat different to the one given by Kechadi. Nevertheless, significant part of the following is a summary of [50]. After the architectural definition of the MCRB network, it will be proved that it is an OS Delta network. Then its characteristics will briefly introduced.

#### The architecture of the MCRB-MIN

The MCRB-MIN is a multistage dynamic implementation based on the static chordal ring architecture. In the following, the architecture of the MCRB network is defined.

Informally, a chordal ring is defined as an augmented ring with "*shortcut*" links between certain nodes [5]. Chordal rings were graphically proposed by Coxter [23] and first proposed as an interconnection network by Arden and Lee [8] who defined a chordal ring to be a ring graph of degree 3, i.e. each node is connected to only one node other than the two nearest neighbours. Chordal rings of degree 3 and higher have been studied in, among others, [52, 12, 11, 9, 105, 77].

A formal definition of bidirectional chordal rings can be found in [52]:

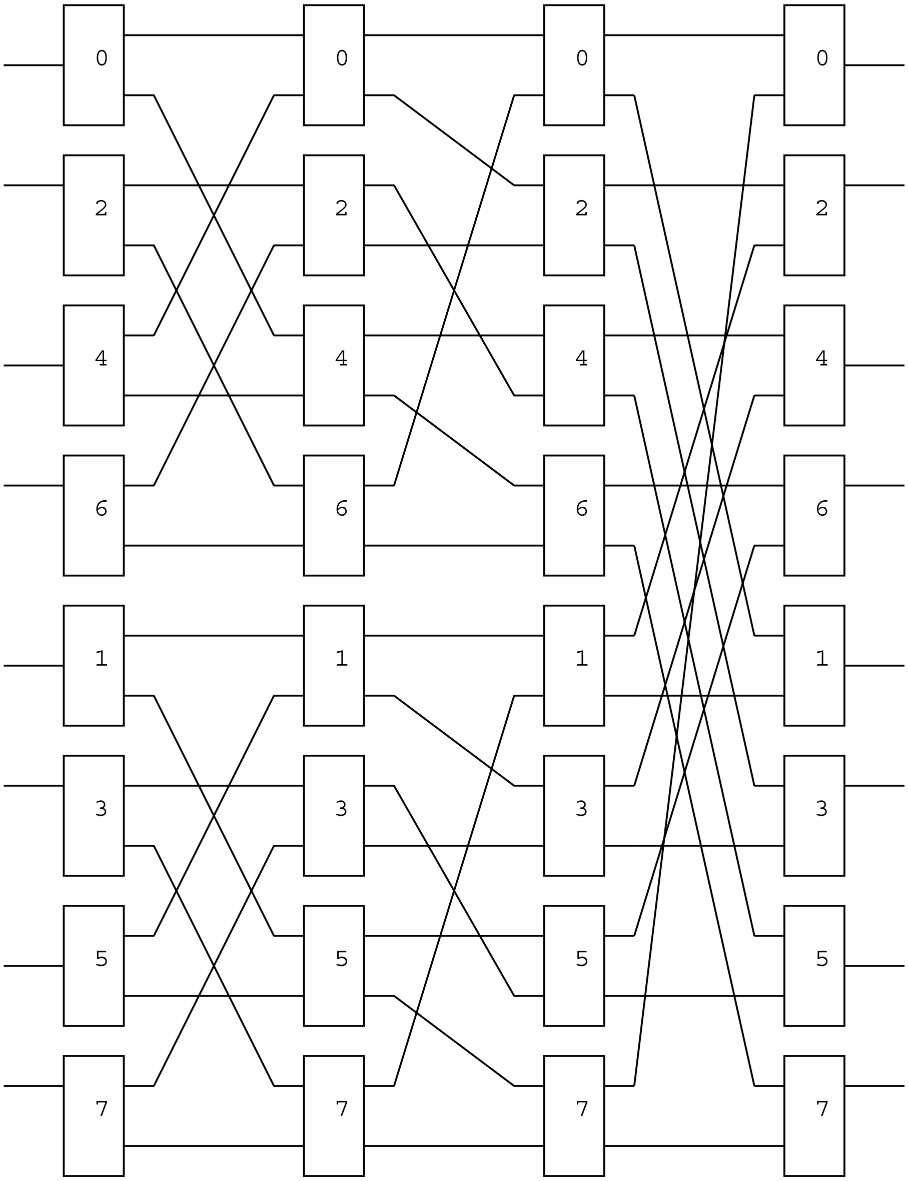

Figure 4.2: The scheme of the OS Delta(8,2)

**Definition 24.** *Given a ring $G = (V, E)$, where $E = \{(v_i, v_{(i+1) mod\ n} | i = 0, 1, \ldots, n-1)\}$, and $V = \{v_0, \ldots, v_{n-1}\}$, and given a set $I \subseteq \{2, \ldots, n-2\}$, a* **chordal ring** *$C = (V, E', I)$ is an undirected graph with $E' = E \cup \hat{E}$, where $\hat{E} = \{(v_i, v_{(i+j) mod\ n}) | i = 0, \ldots, n-1, j \in I\}$. The edges in $\hat{E}$ are called chords. For reasons of compatibility with the definitions of MINs given previously and that will be given later, we have $|V| = N$ and the degree of the graph, i.e. the number of edges connected to each node $r$.*

It can be simply noted that the mesh used in the ILLIAC IV parallel computer can be considered as a special case of the chordal ring.

In a symmetric chordal ring, that is, a chordal ring of which each vertice has the same number of edges, elements of $\hat{E}$ may represent the distances of nodes between which a shortcut exists, i.e. each edge $v_i$ connects every pair of nodes that are at distance $v_i$.

In the case of a complete chordal ring, i.e. a ring for which $N = r^n$, the ring is divided into $r^{n-1}$ groups of $r$ nodes each, and $n$ is defined as the number of dimensions in the graph.

The “goodness” of the chordal ring leads to the definition of its multistage implementation.

Let $d(x, y)$ be the distance along the shortest path and let $\delta_k(x, y)_{0 \leq k \leq n}$ be the number of chords of type $c_k$ included in the shortest path, then the distance between nodes $x$ and $y$ can be defined as $d(x, y) = \delta_0 + \delta_1 + \ldots + \delta_n$. This means that at most $n$ chords are needed to transform a message from the source $x$ to the destination $y$.

Note that on chordal ring networks, the chords along the shortest path can be traversed in any order. If the chords are crossed in a certain order, such as descending order, the resulting path is called a **standard** path. The mathematical representation of the standard path, which consists of a certain number of digits, is called **control sequence**.

**Lemma 1.** *The defined routing sequence for a chordal ring network specifies a unique path between two nodes*

*Proof.* There is a unique representation of a certain base $r$ of the distance $d(x, y)$ between two nodes.

**Proposition 1.** *A chordal ring network with an $r$ based control sequence is a Banyan network.*

*Proof.* This is a direct consequence of lemma 1

**Proposition 2.** *For any two nodes, $x$ and $y$ of a chordal ring network, there exists a unique standard routing sequence $\alpha = \delta_n \ldots \delta_0$, where $0 \leq \delta_i \leq r - 1$ and the number $\delta_n \ldots \delta_0$ is exactly the distance between $x$ and $y$ expresse in the degree of the network $r$.*

*We will call this control sequence an $r$ based control sequence, denoted as $d(x, y)_r = \delta_n \delta_{n-1} \ldots \delta_1 \delta_0$, where $\delta_i$ is the number of chords along the i-dimension of the path.*

*Proof.* This is also a direct consequence of lemma 1.

As stated above, a multistage interconnection network consists of a number of stages, a number of SEs per stage, a connectivity relation between stages, and a routing algorithm. The chordal ring based multistage implementation is given in the following.

**Definition 25.** *Let $N$ be the number of input/output nodes labelled from 0 to $N-1$ in a MIN. A chordal ring based multistage interconnection network, called a* **Multidimensional Chordal Ring Based network** *and denoted MCRB(N,r), has $n$ stages, defined such that for any stage $S_i, 0 \leq i \leq n, S_i$ implements all paths defined only along the i-type dimension of a chordal ring network of size $N$ and degree $r$.*

Structures of MINs are usually influenced by the choice of the routing strategies. Note that the MCRB(N,r) obtained using different configurations of stages are equivalent. After fixing the position of each stage $S_k$, the different i-dimension chords are traversed in the same order as the network stages. The proposed MCRB network is based on the standard routing defined above. Without loss of generality, the stages are implemented in descending order.

**Proposition 3.** *An $MCRB(r^n, r)$ network is a MIN built of $r \times r$ SEs and contains $n$ stages of $r^n$ (SEs) each. Let SE$_{ij}$ be the switching element $j$ of the stage $i$ of the $MCRB(r^n, r)$, then SE$_{ij}$ is connected to SE$_{i-1,k_d}$ such that $k_d = (j + \ d\, r^i) \bmod N$, for $0 \leq i \leq N-1, 1 \leq j \leq r-1$, and $0 \leq d \leq r-1$.*

*Proof.* The SEs of the MCRB network correspond to the nodes of the chordal ring network. Since the MCRB($r^n, r$) network conforms to the standard routing of the proposition 2, the stage $n-1$ implements the dimension $n-1$, the next stage implements the dimension $n-2$, and so on. The SE $i$ of stage $n-1$ is connected to SE $j = (i+\delta_{n-1}r^{n-1})$ of stage $n-2$. This corresponds to the same connectivity of a chordal ring as defined above. The shortest path is defined by an $r$ based control sequence. The mapping of the dimensions defined in the chordal ring network onto stage defines the new MCRB-MIN.

As an example, the configuration of the $MCRB(8,2)$ is shown in Figure 4.3.

MCRB networks can be seen as a special case of Delta networks built by replacing every switch in the Delta network with $r$ switches of the same size. We will now explain the procedure of obtaining the corresponding Delta network of an MCRB network.

**Proposition 4.** *Let $\mu(N, r)$ be an MCRB network of size $N$ and degree $r$. In order to derive the corresponding Delta network, switches of distance $r$ in $\mu$ must be regrouped in order to form a Delta(N,r) network. Moreover, the $r$ corresponding Delta networks can be connected by a last stage. The overall resulting network is a Delta network.*

*Proof.* The proof will be discussed in two steps: we prove in step one that the topology of the first $n-1$ stages of an MCRB network is equivalent to the topology of $r$ Delta networks. In the second step we prove that relaying the $r$

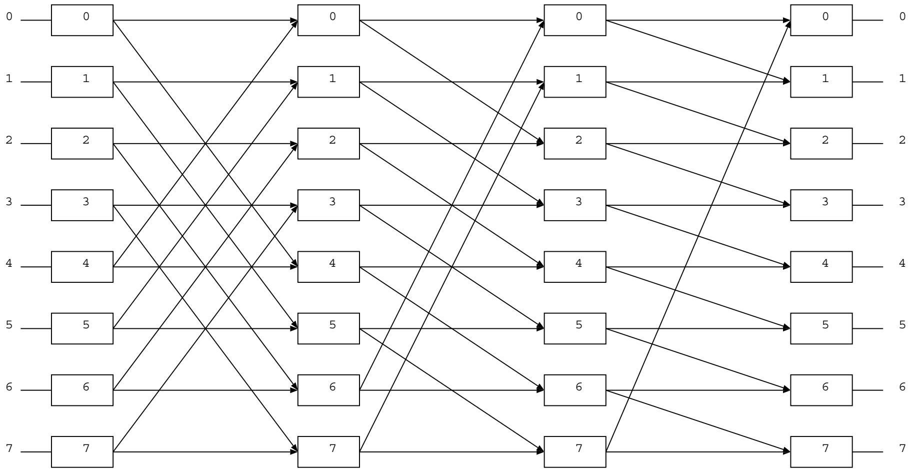

Figure 4.3: $MCRB(8,2)$ Network

delta networks with a stage having the topology of stage 0 of an MCRB network results in a Delta network of size $N$ and degree $r$. Figure 4.4 shows the links of the first four switches on all stages of an MCRB(64,4). Note that, except for the last stage, only one link can replace all $r$ links connecting indexes of the same distances. In other words, only one link and one switch of size 4 are necessary to link outputs 0,1,2,3 of stage 2 to inputs 0,1,2,3 of stage 1, another link and another switch to link the same output to inputs 16,17,18,19. Similar for inputs 32 to 35 and 48 to 51. This gives the schematic of figure 4.5.

**Step 1.** From proposition 3 we see that a switch $j$ in a stage $i$ is connected to switches of which indexes are $j + dr^i$ in stage $i + 1$. In other words we can say that switch $x$ in stage $i$ is connected to switches of which the indexes are of a difference equal to multiplications of $r^i; i > 0$ which yields indexes having the same digits to the base $r$ and so corresponding to definition 9.

**Step 2.** As the MCRB network is divided into $r$ identical Delta networks, it is clear that if a switch in the last stage receives a link from an output of index $i$, all the other inputs will receive links from outputs of index $i$.

Applying this procedure to the MCRB(8,2) shown on figure 4.3 gives the equivalent Delta network of figure 4.2.

**Definition 26.** *In [86] two MINs are defined to be equivalent if and only if they can realise the same permutations by adding a wired permutation stage to one of them.*

**Proposition 5.** *Every over-sized Delta MIN has an MCRB equivalent network.*

*Proof.* As every MCRB network is composed of $r$ Delta networks and a Delta linking stage, we can say: For all values of $N$ and $r$ for which $r$ is a power of 2 and $\log_r N = n, \exists\, \Delta(N/r, r)\&\Delta_s$ where $\Delta_s$ is a linking stage having the Delta property, which can be combined as an MCRB network.

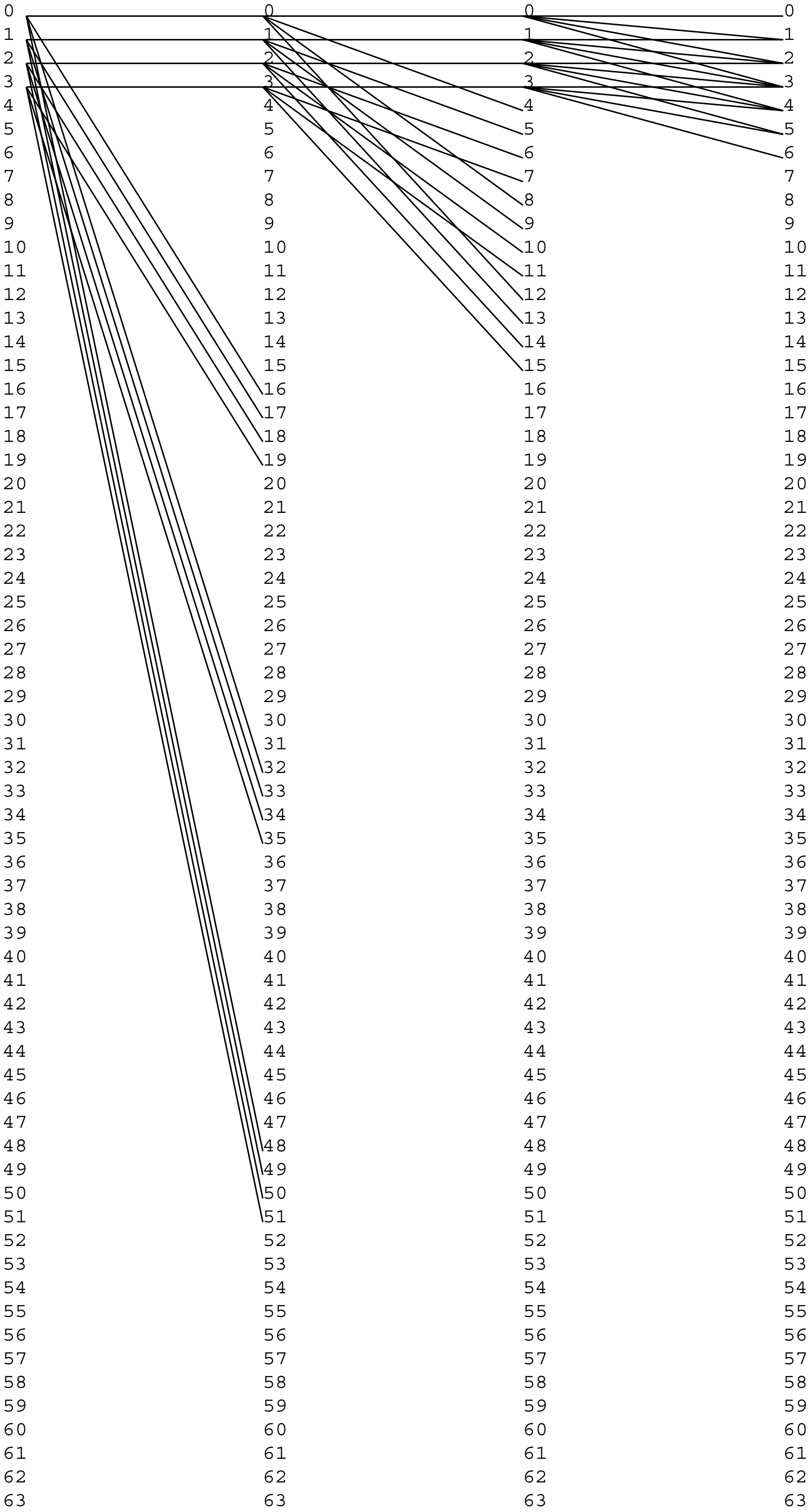

Figure 4.4: A schematic of links in an MCRB(64,4)

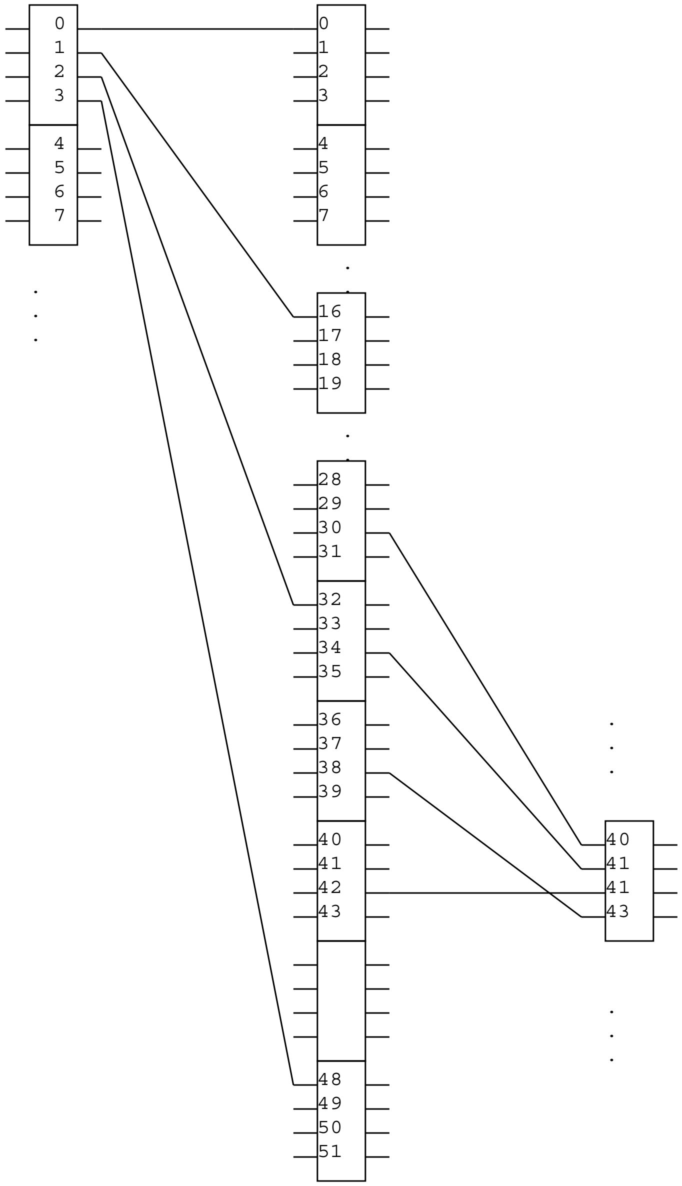

Figure 4.5: A schematic of the first links of a Delta network based on an MCRB network topology

### Characteristics of the MCRB-MIN

Kechadi studied the routing characteristics of the MCRB-MIN in [50]. In this paper, the existence of conflict free routing conditions was proved. For a set of concurrent requests $R = \{(A_i, B_i)|0 \leq i \leq N-1\}$ with $A_i$ the set of sources and $B_i$ the set of destinations the following conditions were given and proved for a conflict free routing in $R$, assuming that each source can issue at most one request.

$$
\begin{aligned}
d(A_i, A_j) &\neq ((d^i_{n-1} - d^j_{n-1})r^{n-1} + \ldots + (d^i_{k+1} - d^j_{k+1})r^{k+1}) mod\ N && (4.1)\\
d(B_i, B_j) &\neq ((d^i_{k-1} - d^j_{k-1})r^{k-1} + \ldots + (d^i_0 - d^j_0)) mod\ N && (4.2)\\
d^i_K &= d^j_k, \forall k \in \{0, 1, \ldots, n-1\} && (4.3)\\
d^i_K &\neq d^j_k, \forall k \in \{0, 1, \ldots, n-1\} && (4.4)\\
gcd(d(A_i, A_j), r) &= 1 && (4.5)
\end{aligned}
$$

Further more, it is proved that the MCRB-MIN is capable of implementing linear skewing schemes. Conditions for such an implementations can be found in [50].

In the following two sections, two interesting special cases of the MCRB network are presented. The first one is the MCRB($r^2, r$) which is proved to be more complex than an identical size crossbar, and the AMCRB, which is a non-Banyan network corresponding to the initial definition of the MCRB network proposed by Kechadi.

## 4.4 The MCRB($r^2, r$)

While networks MCRB($r^2, r$) are more complex than the crossbar and thus are not the subject of this dissertation, they deserve to be mentioned as they provide some interesting characteristics.

These networks, of which the size is a square of the degree, such as MCRB(16,4), MCRB(9,3), etc., seem to be interesting as they are non-blocking. They can route all possible permutations without any conflict. Figure 4.6 shows an MCRB(9,3).

In fact, as shown in figure 4.6, all MCRBs of a size which is a square of the degree have 2 linking stages and 3 SEs stages. As the considered workload is composed of permutations, no conflicts will occur on the first SEs stage, because each processor sends only one request at each cycle. In addition, no memory conflicts can occur, and finally as each SE of the last stage is a demultiplexer and is connected to only one destination node, no conflicts can occur at the stage just before it. This means that no conflicts can occur on the three SE stages of these kinds of networks and also that they are non-blocking.

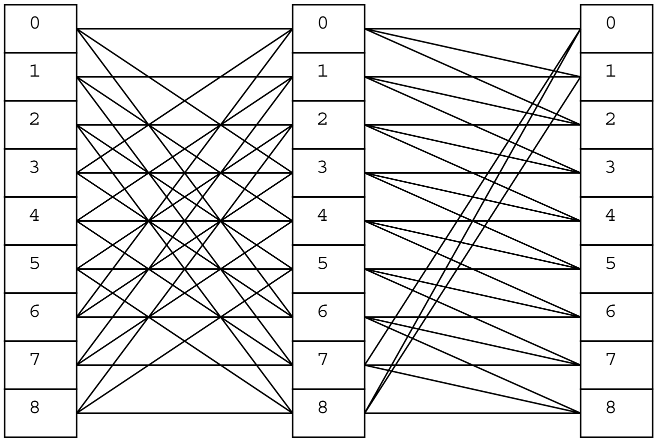

Figure 4.6: The topology of an MCRB(9,3)

## 4.5 AMCRB (Augmented MCRB)

The following definition of the AMCRB-MIN corresponds to the initial definition of the MCRB-MIN given by Kechadi in [50]. As indicated above, for complexity reasons that will be presented later, the definition of the MCRB network in this dissertation is somewhat different from this definition. However, Kechadi's definition is still a good choice for the performance improvement of the network and can be considered as an augmentation technique. This is why the network defined in [50] is called Augmented MCRB (AMCRB).

The AMCRB network consists of two superposed networks, an MCRB-MIN and its r-complement, which is equivalent to the MCRB-MIN. In fact, the superposition of the networks results in a non-Banyan network that is more complex than the MCRB-MIN. However, the AMCRB-MIN can implement rerouting algorithms for the resolution of a conflict or a non availability of a link. Figure 4.7 presents the topology of an AMCRB(16,2). Here are the propositions and definitions necessary for the implementation of this network.

**Proposition 6.** *For any two nodes $x$ and $y$ in a chordal ring network with bidirectional links, there are two standard routing sequences $\alpha$ and $\overline{\alpha}$, its r-complement defining two distinct routing paths between $x$ and $y$.*

*Proof.* $y = (x - d(x,y)) mod\ N$ can be written as $x = (y + d'(x,y)) mod\ N$. $d'(x,y) = -d(x,y)$, which is the complement of $d(x,y)$ and can be represented by r-complement of $d(x,y)$.

The previously defined standard routing strategy uses the shortest path to transmit a message from a source to a destination. The path as it is defined, specifies the number of links of each dimension. In a chordal ring network the links of the shortest path can be traversed in any order. However, the standard routing defined in proposition 2 fixes the order in which the links of different dimensions are traversed.

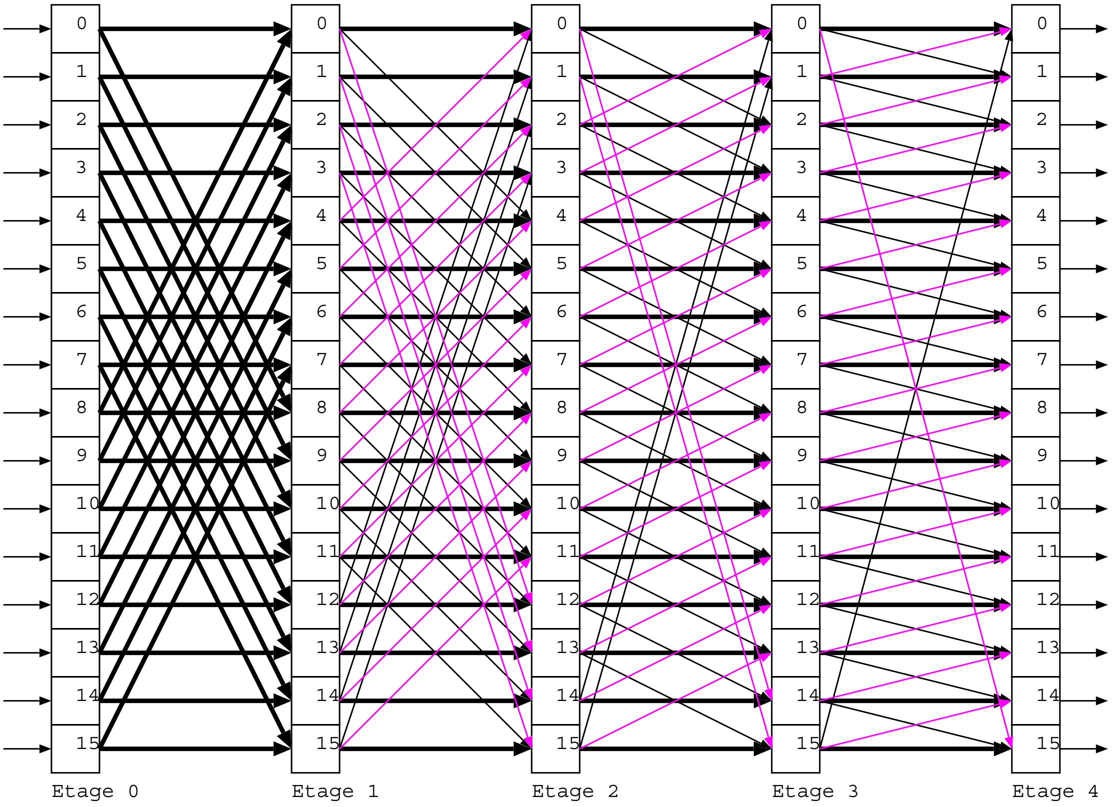

Figure 4.7: The topology of the AMCRB(16,2)

Routing a message from a source to a destination in an MRCB-MIN consists of performing at any stage $k$ the label $k$ of the next SE of the stage $k-1$. If the routing sequence is $d(x,y)_r = \delta_n\delta_{n-1}\ldots\delta_1\delta_0$, then the label $i_{k-1}$ of the next SE is $i_{k-1} = (i_k + \delta_{k-1}) mod\ N$. This can be done locally at every SE. However, in the case of a conflict in an AMCRB-MIN, this routing algorithm can be improved by rerouting the conflicting messages. A rerouting policy for a standard routing is described in the following.

**Definition 27.** *Two control sequences $\alpha(a,b)$ and $\beta(c,d)$ are said to be equivalent iff $a = c$ and $b = d$.*

**Theorem 1.** *Given an AMCRB-MIN, a control sequence $\alpha$ and its r-complement $\overline{\alpha}$ are equivalent.*

*Proof.* Follows directly from proposition 2.

**Theorem 2.** *Given an AMCRB(N,r), and a message with a routing sequence $\alpha(x,y)$ that presents a conflict at a switching element $SE_{i}j$. If $d(SE_{i-1j}, y \neq 0$ and $\delta_i < r-1$ then the new rerouting sequence from $SE_{ij}$ to $y$ will be $(\delta_i + 1)\overline{\alpha}(SE_{i-1j}, y)$.*

*Proof.*Assume that the control sequence $\alpha(x,y) = (\delta_{n-1}\ldots\delta_{i+1}\delta_i\ldots\delta_0)_r$. If a conflict occurs in the $SE_{ij}$ of stage $i$, then the remaining digits of the control sequence, $\delta_i\ldots\delta_0$ will be replaced by $\delta'_i\ldots\delta'0$ such that the two control sequences are equivalent. The first $(n-i)$ digits are the same, this implies that $\delta_i r^i \ldots \delta_0$ and $\delta' i r^i \ldots \delta'_0$ have to be equivalent. From theorem 1 one can conclude that the new control sequence is the r-complement of the first one. As the AMCRB-MIN consists of two complemented networks, the remaining links can be selected on the complement network according to the new control sequence.

The following example shows how this rerouting algorithm solves conflicts that can occur at intermediate stages of an AMCRB-MIN. Consider an AMCRB(27,3), and two messages to be routed. The first message is issued from node 0 to node 14 and the second is from node 9 to node 13. Using the standard routing, the two messages will be routed as follows: the distance between the nodes 0 and 14 is $d(0,14) = 112_3$ and that between nodes 9 and 13 is $d(9,13) = 011_3$. The corresponding paths are represented by solid lines in figure 4.8. Due to a conflict occurring at $SE_{9,1}$, one of the two messages needs to be rerouted. Supposing that the first message is chosen to be rerouted, the link represented by $d_1 = 1$ is replaced by $d_1 = 2$ and $d_0$ is replaced by its 3-complement which is $d_0$=1 as shown in figure 4.8.

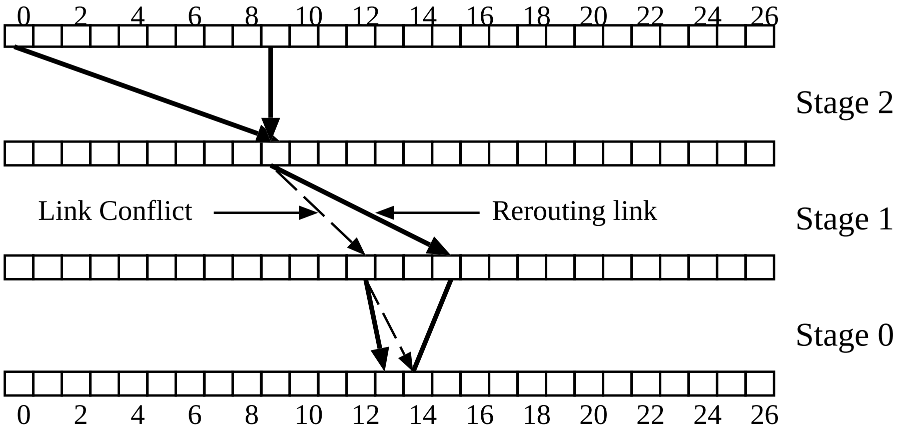

Figure 4.8: Rerouting in the AMCRB network

Finally, it is important to indicate a special case of the AMCRB network, recognized and studied in special contexts: the data manipulator.

**Proposition 7.** *The topology of the Data manipulator defined in [33] is a special case of the AMCRB-MIN.*

*Proof*: follows directly from the two definitions of the networks and $r = 2$. Feng's network is a centralized controlled network used to implement data manipulating functions. This explains its pretended low complexity. In the case of the AMCRB, crossbars are used in a distributed controlled MIN used as an alignment network.

## 4.6 Conclusion

In this chapter, an improvement technique for Delta-MINs, called over-sizing was introduced. Improvement techniques are used in order to apply simple modifications to Delta networks so that increased performance can be achieved without much loss in the architectural and algorithmic simplicity of Delta-MINs. Some previous techniques were presented and then the over sizing technique was introduced. The MCRB network, an over-sized Delta-MIN, as well as it characteristics were presented and a non-Banyan version of it, the AMCRB was defined before the conclusion.

In the last two sections, an evaluation and comparison methodology for MINs performance and a new improved Delta family were proposed. In the next section, the methodology will be applied to this new family which will be compared to Omega network, a well know Delta network.

# Chapter 5

# Performance evaluation results

## 5.1 Introduction

In this section we evaluate and compare the two MINs, i.e. the MCRB and Omega networks using the previously defined performance factors.

Three different evaluations are presented, one general, i.e. a simple evaluation of the networks with respect to the considered metrics, followed by a study of the effect of the degree of a MIN on its performance and finally a study of the scalability of MINs. In the first study, the two ways of using the UPF are considered, i.e. its use as a distance function for the comparison of networks and its use to define a group of Pareto optimal solutions. For the other two comparisons, only the comparison of UPFs as distance functions is used, because in these cases a real comparison is required in order to find the best network among different solutions, rather than a group of optimal solutions.

## 5.2 Mathematical validation of the simulator results

In this section the effectiveness of the proposed simulation approach is validated. Patel's formula and its approximation found by Kruskal and Snir for the acceptance probability are presented. These formulas are used in order to validate the effectiveness of the simulation approach, so as to be sure of the results obtained from the simulator.

In his definition and study of Delta networks [78, 79], Patel defines crossbars which are the SEs used in these networks. He then defines the *acceptance probability $P_A$* as the "*probability that an arbitrary request will be accepted*". In other words, the probability that a request arrives at its destination. To calculate this probability, Patel calculates the simultaneous bank conflict probability of a crossbar, and then approximates the behaviour of a stage of crossbars to the behaviour of a single crossbar.

His calculations give for an $M \times N$ crossbar, with a probability of request generation at its inputs equal to $m$:

$$P_A = \frac{N}{mM} - \frac{N}{mM}\left(1 - \frac{m}{M}\right)^M \tag{5.1}$$

Using equation 5.1, one can calculate the acceptance probability at each stage in the network, and then calculate the throughput of the one size buffer SE network. The acceptance probability of the final stage is then divided by what we call the message generation load or the workload, which is the number of processors generating messages per cycle.

Kruskal and Snir [53] presented an asymptotic analysis of the performance of unbuffered Banyan networks, and they found an approximation based on Patel's formula for calculating the acceptance probability of a network, this formula is:

$$P_A = \frac{2k}{(k-1)m + \frac{2k}{p}} \tag{5.2}$$

where $k$ is the degree of the crossbar, $m$ is the stage number for which we calculate acceptance probability, and $p$ is the message generation probability.

In fact, the random and uniform distribution of source and destination node message generating allows us to use Patel's formula [78] presented in equation 5.1 to calculate the throughput of the network. This formula gives the probability that a message existing on an input of a crossbar, or, by approximation, at a stage in the network, can pass the stage without a conflict. So, to calculate the probability of having a message on an output of a crossbar, the result given by Patel's formula must be multiplied by $\frac{M}{N}$, where the crossbar is an $M \times N$ crossbar. Then this result must be calculated considering the input probability of the stage. This leads to:

$$p_s^i = \left[\frac{N}{p_s^{i-1} \times M} - \frac{N}{p_s^{i-1} \times M}\left(1 - \frac{p_s^{i-1}}{N}\right)^M\right] \times \frac{p_s^{i-1} \times M}{N}$$

where $p_s^i$ is the probability of having a message on an output of one of the SEs in stage $i$.

Using this result and by dividing the probability of having a message on the output of the last stage by the message generation load we can find the throughput of the network.

Usually, performance measures for MINs are driven by the following types of requests [3] that an interconnection network may carry out.

- `one-to-one request type` where a source node sends a message to a specific destination node.

- `Permutation request type` where each active destination node is assigned to one and only one active source node.

- `Broadcast request type` where a source node sends the same message to more than one destination.

- `many-to-one request type` where more than one source node send requests to the same destination node.

Of these four request types, the above probability acceptance formulas can be applied on the many-to-one request type, and the permutation request type.

The only difference that we found when the destinations are groups of permutations is that, in an MCRB network, no conflict occurs in the last two stages of SEs. In addition:

$$p_s^l = p_s^{l-2} \times n \tag{5.3}$$

Where $l$ represents the last stage of the network. While in an Omega network, no conflict can occur only on the last stage of the SEs, and

$$p_s^l = p_s^{l-1} \tag{5.4}$$

Some examples of comparison of analytical and simulation results for MCRB and Omega networks for the many-to-one request and the permutation request types and for different input workloads (the far right columns of the tables), are shown on tables 5.1 and 5.2. Patel's formula was used in order to calculate the MCRB networks' throughputs, and Kruskal and Snir's formula to calculate the Omega networks' throughput. This is because Kruskal and Snir's formula can be easily applied to the Omega network case as it is a square network.

Tables 5.1 and 5.2 show that the proposed simulation approach gives good results. We observe that simulation results depend little upon workload and hence provide realistic assessment. The difference between calculated and simulated results is smaller in the many-to-one request case, but even in the permutation request case this difference does not exceed 10%. These good results allow us to trust our simulation approach and thus allow us to test other networks and communication patterns with confidence. In fact, simulation allows for a more flexible characterization of networks than an analytical model, as it permits better control over communication patterns and routing algorithms, so, real and popular communication patterns as defined by [60] can be analysed.

| | 16 | | | | 128 | | | |
|---|---|---|---|---|---|---|---|---|
| | MCRB | | Omega | | MCRB | | Omega | |
| | Sim | Calc | Sim | Calc | Sim | Calc | Sim | Calc |
| 30% | 84.91 | 83.53 | 77.22 | 76.90 | 76.10 | 76.17 | 64.04 | 65.57 |
| 50% | 75.81 | 74.69 | 65.30 | 66.66 | 65.34 | 65.19 | 51.04 | 53.33 |
| 100% | 57.98 | 57.96 | 44.96 | 50.00 | 47.41 | 47.18 | 32.65 | 36.36 |

Table 5.1: Comparison between simulation and calculation results for different MCRB and Omega networks in the many-to-one requests case. “Calc” corresponds to Calculation and “Sim” to Simulation

| | 16 | | | | 128 | | | |
|---|---|---|---|---|---|---|---|---|
| | MCRB | | Omega | | MCRB | | Omega | |
| | Sim | Calc | Sim | Calc | Sim | Calc | Sim | Calc |
| 30% | 95.60 | 92.77 | 85.88 | 79.99 | 86.39 | 82.29 | 70.98 | 65.57 |
| 50% | 92.43 | 88.25 | 76.93 | 72.72 | 78.82 | 75.13 | 58.69 | 57.14 |
| 100% | 84.25 | 77.92 | 58.13 | 57.14 | 63.79 | 59.00 | 39.31 | 40.00 |

Table 5.2: Comparison between simulation and calculation results for different MCRB and Omega networks in the permutations request case. “Calc” corresponds to Calculation and “Sim” to Simulation

## 5.3 Single criterion evaluations

To test the temporal aspects of the MINs that we are analysing, the MQM simulator explained previously is used to route BPC permutations. In order to calculate the number of cycles needed to route or reroute a certain number of permutations, a circuit switching strategy is used. Thus, when a message is detected at the input buffer of a SE at the first stage, a routing path is reserved if all the SEs describing it are free. A conflict situation occurs when a message tries to use a buffer already occupied by another message. In this case, the message stays in the input buffer at the first stage, but it will be erased from all other buffers previously allocated. When all routable messages arrive at their destinations, the simulator starts another attempt to reroute the non-routable messages from the previous cycle. This procedure will be repeated until all *input* messages arrivea at their destinations.

### 5.3.1 Integration Complexity

It is easy to verify that inter-stage complexity is lower than cross-points complexity for both MCRB and Omega networks. Therefore, the study of the integration complexity is limited to that of cross-points complexity. First of all, we calculate the complexity in terms of cross-points and compare them to the crossbar complexity. This step is important in order to eliminate networks of complexity greater than the crossbar of the same size.

The MCRB network is composed of $\log_r(N)$ linking stages and $\log_r(N)+1$ crossbar stages. The two end-point stages are composed of $N$ multiplexers / demultiplexers. Each multiplexer and demultiplexer has $r$ cross-points. Thus the complexity of the input stage and the output stage is equal to $2Nr$. Each of the remaining $\log_r(N)-1$ stages is composed of $N$ crossbars of degree $r^2$, which gives a total complexity of $Nr^2$ for each stage. Thus, the MCRB network complexity is

$$C_{MCRB} = 2Nr + [\log_r(N) - 1] \times Nr^2 \tag{5.5}$$

For this dissertation, the interesting parameter values of MCRB(N,r) are those that lead to a complexity lower than $N^2$, which is the complexity of a crossbar of size $N$. Figure 5.1 shows that MCRB complexity is, for some values of $N$ and $r$, greater than that of the crossbar. For $r \geq 3$, the interesting values of $r$ for which the complexity of the MCRB network is lower than $N^2$ are $r = \frac{log(N)}{2}$.

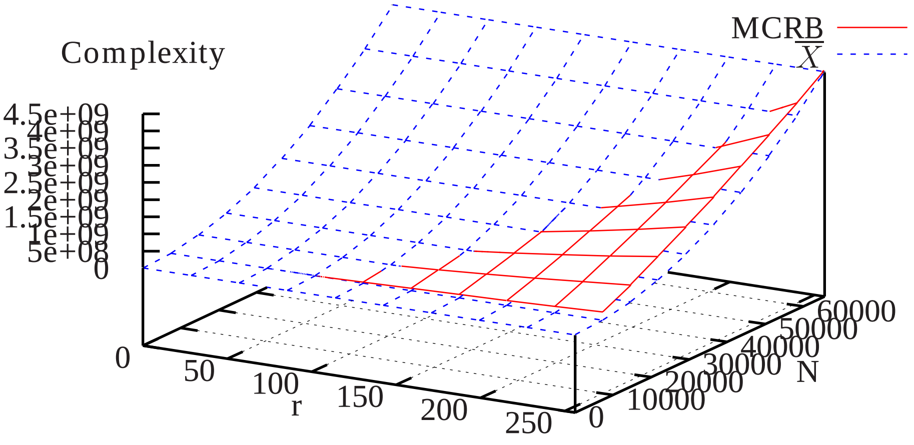

Figure 5.1: Crossbar and MCRB network complexity as a function of their size $N$ and the degree $r$ (only MCRB). $\overline{X}$ stands for crossbar

Figures 5.2 and 5.3 show different projections of Figure 5.1. We observe that for $r = 2$, MCRB(4,2) and MCRB(8,2) are more complex than the crossbar. When $N = 16$ complexity values for the MCRB network and the crossbar are equal. Therefore, before implementing an MCRB(N,r), one should make sure, using equation 5.5, that its complexity is lower than $N^2$.

All crossbars in the square Omega networks which are considered in this dissertation are of size $r^2$ and are distributed on $\frac{log(N)}{log(r)}$ stages each containing $N/r$ SEs, this gives the Omega network its complexity:

$$C_\Omega = rN \log_r(N) \tag{5.6}$$

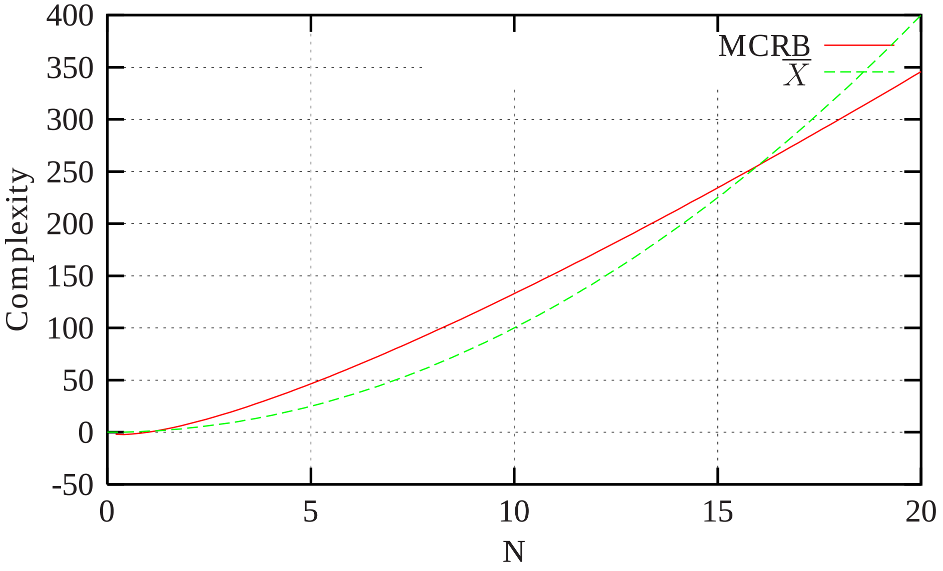

Figure 5.2: Crossbar's and MCRB network's complexity as a function of the network's size and for r=2. $\overline{X}$ stands for crossbar

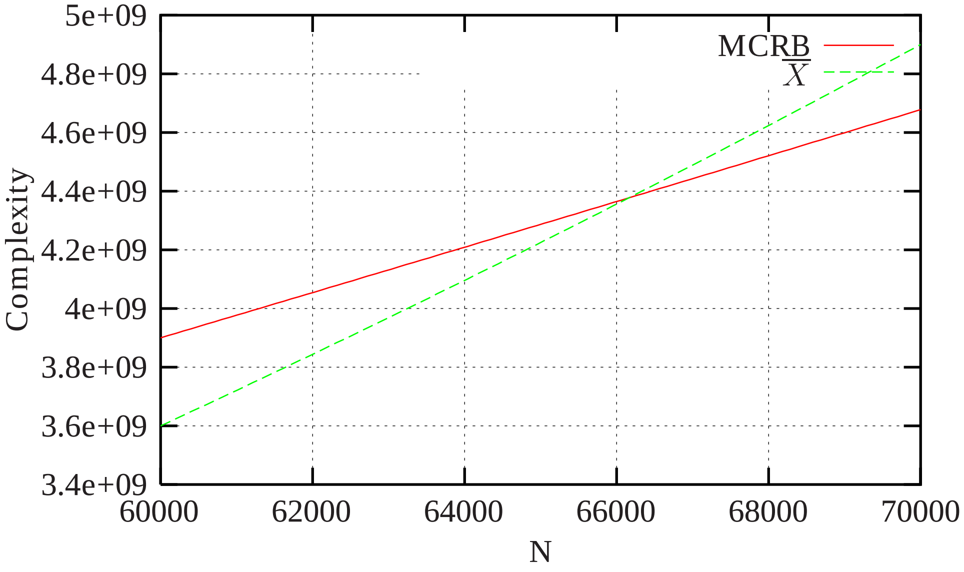

Figure 5.3: Crossbar's and MCRB network's complexity as a function of the network's size and for r=256. $\overline{X}$ stands for crossbar

From equation 5.6 one notices that the Omega network complexity is always less than $N^2$. The order of complexity between two MINs is defined as the ratio of their complexities, as both of them will be normalized to the complexity of the crossbar of the same size.

$$\Delta_C = \frac{C_{MCRB}}{C_{\Omega}} \tag{5.7}$$

By substituting the values of $C_{MCRB}$ and $C_{\Omega}$ from equations 5.5 and 5.6 respectively, one can write

$$\Delta_C = \frac{\log(r)}{\log(N)}(2-r) + r \tag{5.8}$$

For a certain value of $r > 2$, the term $\frac{\log(r)}{\log(N)}(2-r)$ is negative and smaller than $r$. Thus, when $N$ gets bigger, $\Delta_C$ gets bigger too and so MCRB network complexity is always greater than that of Omega networks. The question is then, how to justify the increase in complexity. In other words, will the network's performance be better than that of the less complex network?

### 5.3.2 Throughput of MCRB and Omega networks

Figures 5.4 and 5.5 show the throughput of MCRB and Omega networks of degrees 2 and 4 as a function of their sizes. As stated before, this throughput is studied using a workload that increases with the size of the network. This explains the downward trend of the throughput of all the studied networks.

As expected, the throughput of the MCRB network is higher than that of the Omega network. This is a direct result of the fact that the initial workload of an MCRB is distributed on a greater number of SEs than for a similar size Omega network, and thus less conflicts take place.

To observe the difference between the cases, the throughput variation of different networks is plotted as a function of size in figure 5.6. The variation is calculated as the difference (subtraction) of the throughputs of MCRB and Omega networks.

The figure shows that, even if the throughput varies inversely with the increasing of the MIN's size, higher degree MCRBs have a greater throughput with respect to Omega networks.

### 5.3.3 Permutation Capability of MCRB and Omega Networks

Figures 5.7, 5.8, and 5.9 show some examples of the permutation capabilities of Omega networks and MCRB networks with different values of $r$. These figures show the percentage of permutations that can be routed *within* a certain number

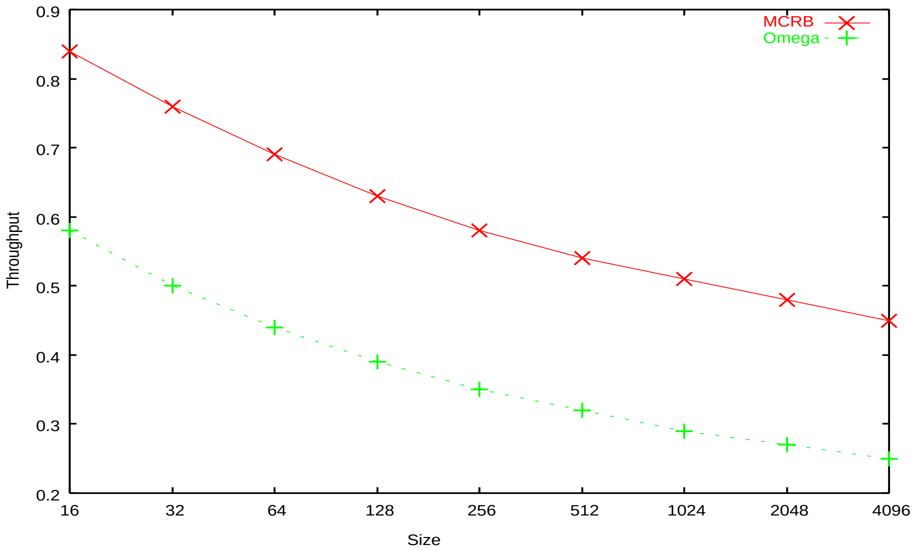

Figure 5.4: The throughput of MCRB and Omega networks of degree 2 and different sizes

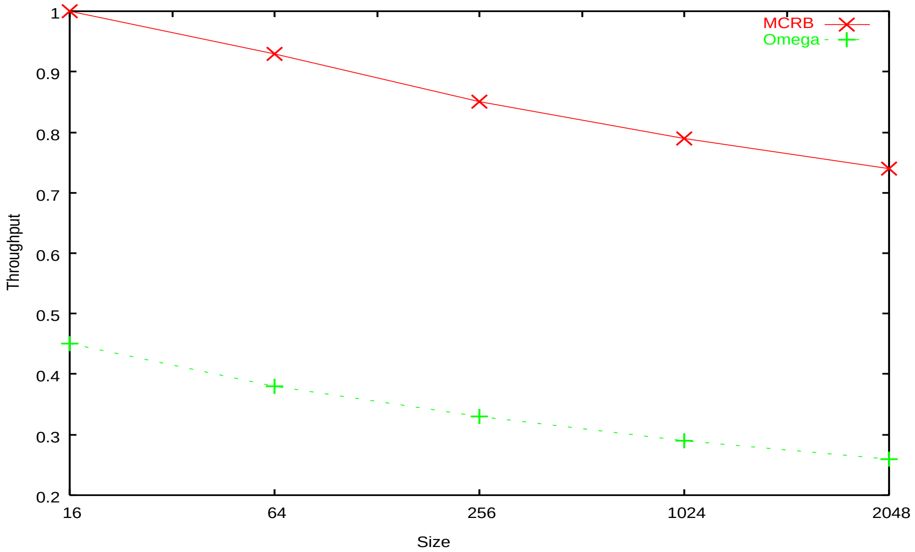

Figure 5.5: The throughput of MCRB and Omega networks of degree 4 and different sizes

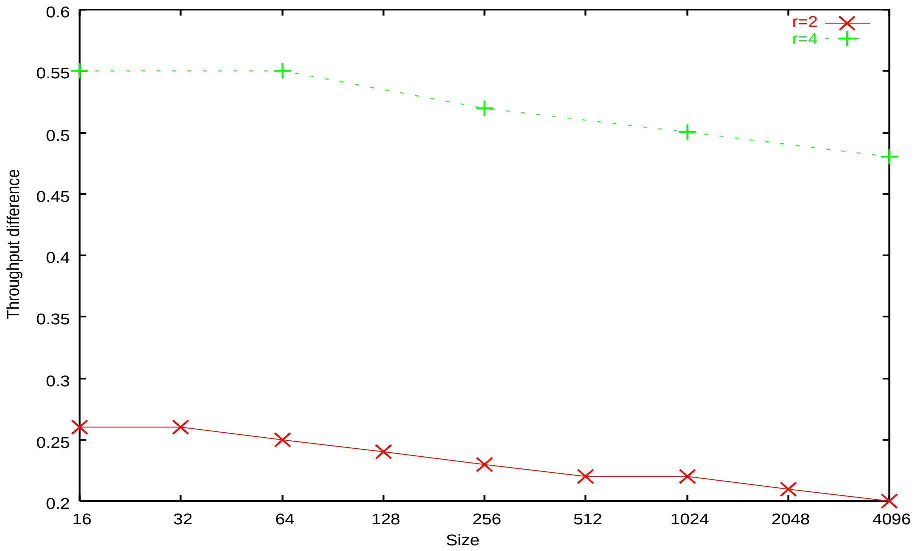

Figure 5.6: The throughput variation of MCRB and Omega networks of degrees 2 and 4

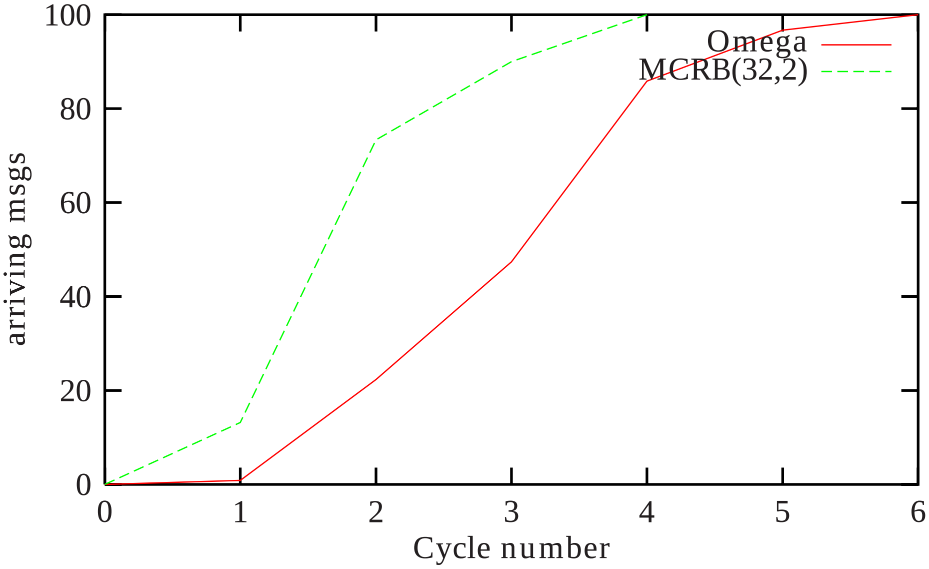

Figure 5.7: Permutation capabilities of MCRB and Omega networks of size 32

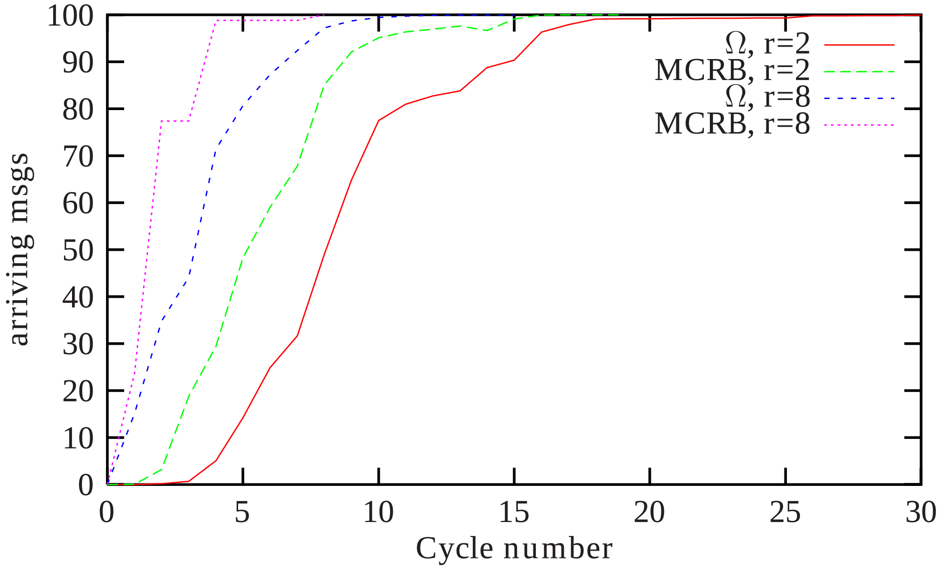

Figure 5.8: Permutation capabilities of MCRB and Omega networks of size 512

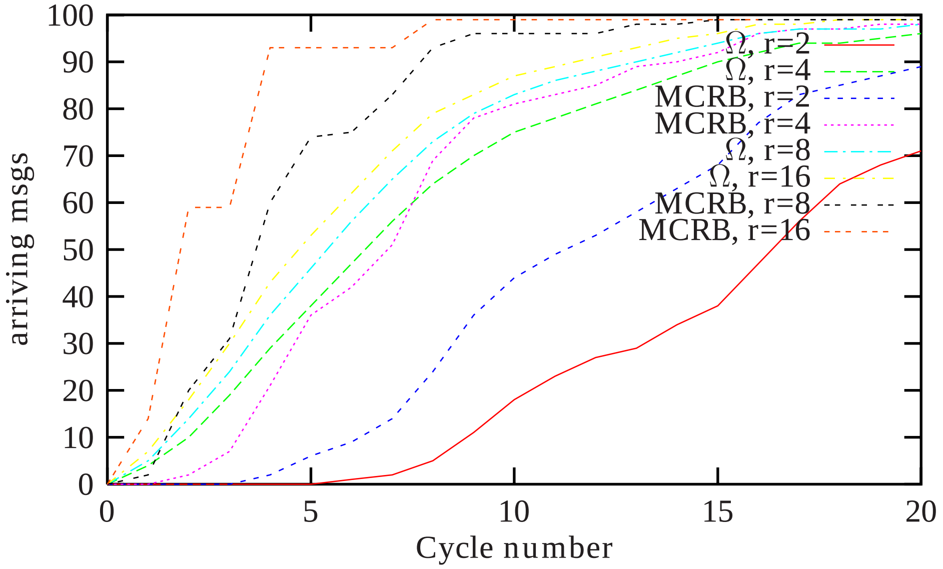

Figure 5.9: Permutation capabilities of MCRB and Omega networks of size 4096

of cycles. This means that for a cycle $i$, the presented value is the percentage of permutations that could be routed in all cycles $i, i-1, i-2, \ldots, 1$.

These figures show that there is a considerable gain in the permutation capacity of the MCRB network relative to the Omega network. Although the MCRB network is very powerful in terms of permutation capability, which is expected, it still has higher integration complexity. A detailed discussion of the relation between the improvement of MCRB network's performance and its complexity is presented later.

### 5.3.4 The Universality of MCRB and Omega networks

Larger size MINs have greater universality, this is due, as in the case of the throughput, to an increasing workload which leads to a higher probability of conflicts.

Figure 5.10 shows that for $r = 2$ the maximum number of cycles needed to route permutations in an MCRB is considerably less than for an Omega network. The gap increases with the network size. This is good for the MCRB, especially given that for $r = 2$ it is only two times more complex than Omega (see equation 5.8).

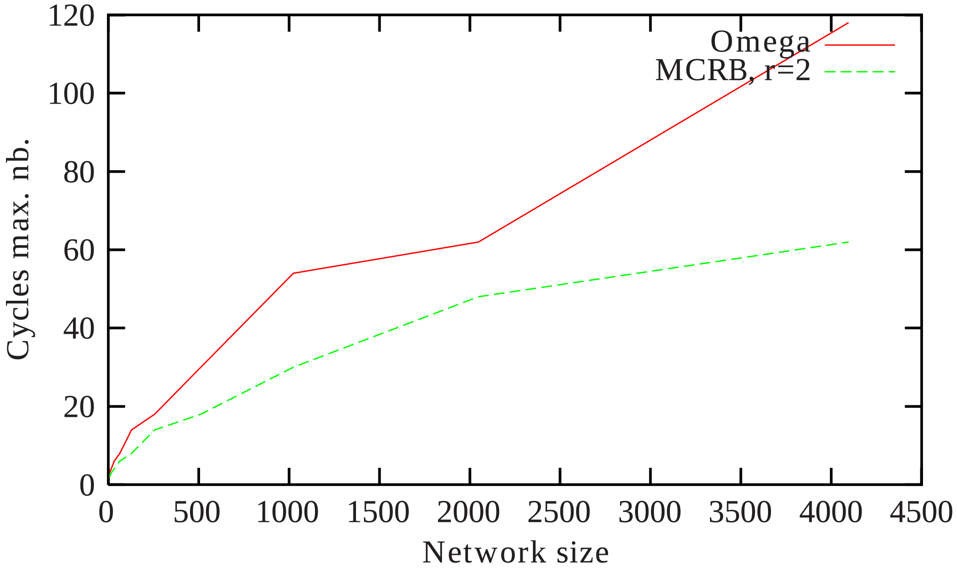

Figure 5.10: The maximum number of cycles needed to route a certain number of BPC permutation on MCRB and Omega networks. The y axis is the maximum number of cycles needed to route tested permutations

The last sections showed that, except for the complexity, which is higher for the MCRB than for the Omega network, the former performs better for the considered metrics. In the following sections, metrics will be associated together in order to study the effect of contradictory performance factors when evaluated simultaneously.

## 5.4 Two and three criteria evaluations

In this section, Two and three dimensional evaluations will be presented. Networks of different sizes and degrees are to be tested. Note that not every composition of factors is available for use in comparison of MINs. For example, the use of complexity with universality does not represent an acceptable test as MINs of small size are not complex and do not need a lot of cycles to route the simple permutations that they are capable of making.

### 5.4.1 Complexity and Throughput UPF

Comparing the UPFs of several MINs regarding only taking into account the complexity and the throughput means that the universality of the networks is not an important factor to be evaluated. On the other hand, the “betterness” of the network is judged using a low complexity and a high throughput. Table 5.3 shows the normalized values of throughput and complexity of considered MINs and figure 5.11 is a plot of it.

| MIN | Min. Norm. Th. | Norm. Comp | Pareto |
|---|---|---|---|
| MCRB(256,2) | 0.39 | 0.11 | * |
| Omega(256,2) | 0.63 | 0.06 | * |
| MCRB(256,4) | 0.11 | 0.19 | * |
| Omega(256,4) | 0.65 | 0.06 | |
| MCRB(512,2) | 0.43 | 0.25 | |
| Omega(512,2) | 0.66 | 0.13 | |
| MCRB(512,8) | 0 | 1 | * |
| Omega(512,8) | 0.78 | 0.17 | |
| MCRB(1024,2) | 0.46 | 0.56 | |
| Omega(1024,2) | 0.69 | 0.28 | |
| MCRB(1024,4) | 0.17 | 1 | |
| Omega(1024,4) | 0.69 | 0.28 | |

Table 5.3: The Pareto optimal MINs for throughput-complexity UPF

The figure can be analysed in two ways, either as a multi-objective optimization problem or as a performance comparison problem.

For the first approach, MCRB(512,8), MCRB(256,4), MCRB(256,2) and Omega(256,2) are Pareto optimal solutions. Of special interest are MCRB(512,8) and MCRB(256,4). The later MIN has the smallest UPF value. MCRB(512,8) is an optimal solution for the throughput while its complexity is very high. Note that

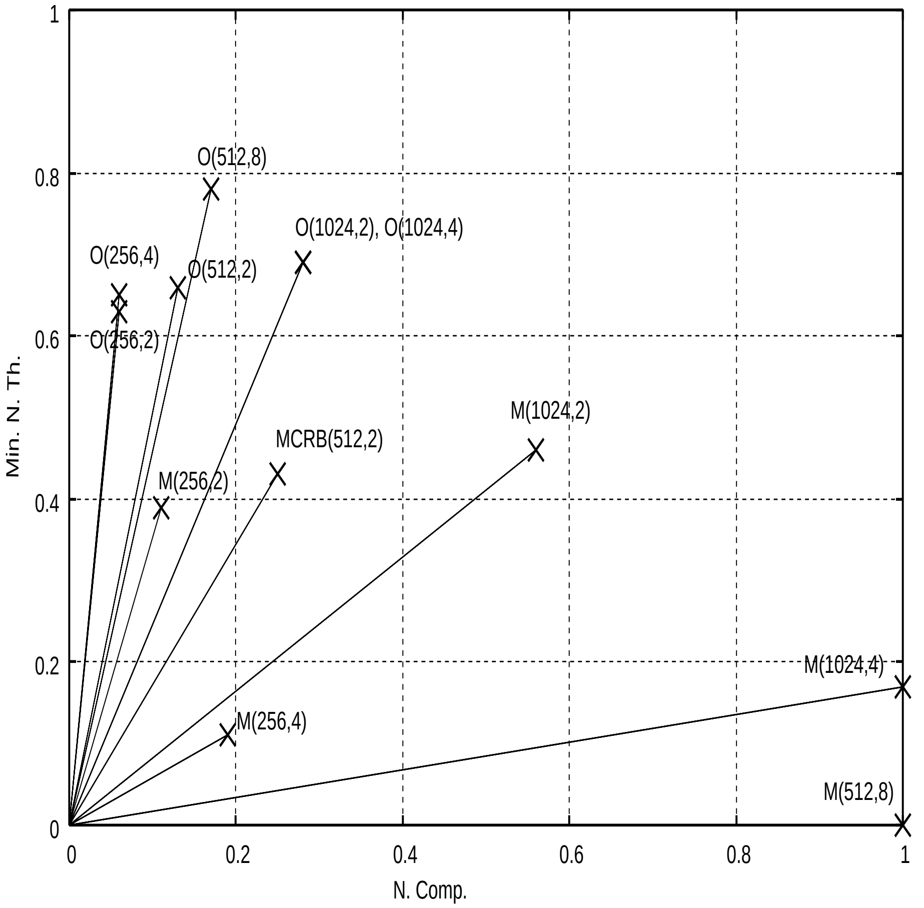

Figure 5.11: Comparing some MCRB and Omega MINs regarding their throughputs and complexities

the $y$ axes represents the minimized normalized value of the throughput and not the throughput itself, in other worlds, MCRB(512,8) is not non-blocking. However, among the studied networks, it has the largest throughput. On the other hand, the Omega(256,2) is an optimal solution, even if its throughput is very low. Its optimality is due to its very low complexity as shown in figure 5.11.

Considering the comparison as a distance function, one can note that MCRB(256,4) is the best among the tested MINs and MCRB(1024,4) and MCRB(512,8) have high UPFs and thus they are considered to be unacceptable networks. The figure also shows the reason for penalizing these two networks, that is, their complexities. In fact, these two MINs are among the most complex networks in the sample of studied networks.

One attractive point when considering the application of the UPF as a distance function for the comparison of MIN performance is the possibility of comparing different MINs of different architectural characteristics. One can note on figure 5.11 that using this logic MCRB(256,4), MCRB(256,2), MCRB(512,2) are better than Omega(256,2) and Omega(256,4).

The last point to be considered is the possibility of using the UPF distance function in order to make a choice among the Pareto optimal values, which in the studied case results in the MCRB(256,4) as a *best* MIN among the optimal possible solutions. This utilization may lead to the elimination of both MCRB(512,8) because of its high complexity and Omega(256,2) because of its low throughput.

### 5.4.2 Universality and Throughput UPF

Here, the cost of the MIN is not an important factor to be evaluated. So, we are looking for the MIN that gives the best universality AND throughput at the same time regardless of the complexity. Table 5.4 shows the normalized values of throughput and universality of considered MINs and figure 5.12 is a plot of it.

The best way to study this figure and to understand the use of the UPF to evaluate and compare MINs performance would be the comparison of figures 5.11 and 5.12.

It is expected that MCRB networks show a very good performance when complexity is not considered. This claim is clearly represented in figure 5.12. All Omega networks are less powerful than all MCRB-MINs even with size difference between the two families.

MCRB(512,8) (which has one of the largest complexities among studied MINs) becomes one of the most powerful networks for the universality-throughput UPF case and becomes one of the two Pareto optimal solutions. From a Pareto optimization point of view MCRB(512,8) and MCRB(256,4) are optimal solutions. On the other hand, considering distance, both MCRB(512,8) and MCRB(256,4) are equal with respect to the universality-throughput UPF.

| MIN | Min. Norm. Th. | Norm. Univ. | Pareto |
|---|---|---|---|
| MCRB(256,2) | 0.39 | 0.26 | |
| Omega(256,2) | 0.63 | 0.33 | |
| MCRB(256,4) | 0.11 | 0.11 | * |
| Omega(256,4) | 0.65 | 0.3 | |
| MCRB(512,2) | 0.43 | 0.35 | |
| Omega(512,2) | 0.66 | 0.56 | |
| MCRB(512,8) | 0 | 0.15 | * |
| Omega(512,8) | 0.78 | 0.28 | |
| MCRB(1024,2) | 0.46 | 0.56 | |
| Omega(1024,2) | 0.69 | 1 | |
| MCRB(1024,4) | 0.17 | 0.31 | |
| Omega(1024,4) | 0.69 | 0.56 | |

Table 5.4: The Pareto optimal MINs for throughput-universality UPF

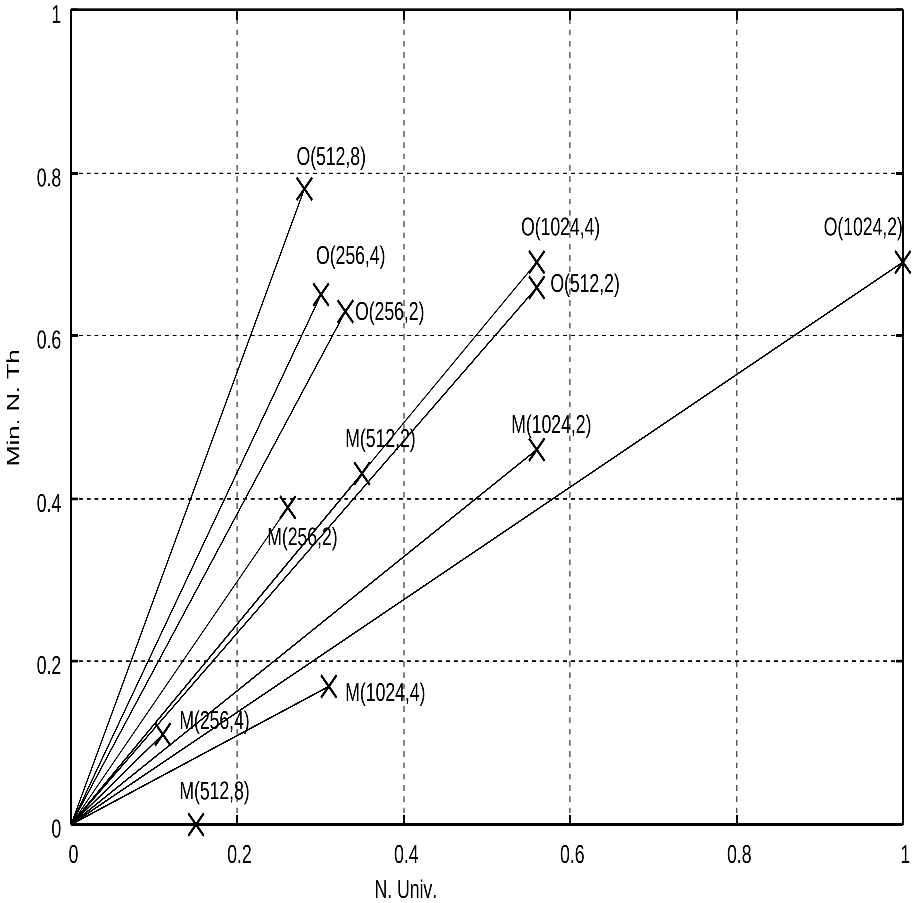

Figure 5.12: Comparing some MCRB and Omega MINs regarding their throughputs and complexities

### 5.4.3 3 criteria evaluations

Here all the three metrics previously studied either separately or on two dimensions are to be analyzed together. Figure 5.13 plots the values of the considered MIN UPFs of table 5.5.

| MIN | Min. Norm. Th. | Norm. Comp | Norm. Univ | Pareto | UPF |
|---|---|---|---|---|---|
| MCRB(256,2) | 0.39 | 0.11 | 0.26 | * | 0.23 |
| Omega(256,2) | 0.63 | 0.06 | 0.33 | * | 0.51 |
| MCRB(256,4) | 0.11 | 0.19 | 0.11 | * | 0.06 |
| Omega(256,4) | 0.65 | 0.06 | 0.3 | * | 0.52 |
| MCRB(512,2) | 0.43 | 0.25 | 0.35 | | 0.37 |
| Omega(512,2) | 0.66 | 0.13 | 0.56 | | 0.77 |
| MCRB(512,8) | 0 | 1 | 0.15 | * | 1.02 |
| Omega(512,8) | 0.78 | 0.17 | 0.28 | | 0.72 |
| MCRB(1024,2) | 0.46 | 0.56 | 0.56 | | 0.84 |
| Omega(1024,2) | 0.69 | 0.28 | 1 | | 1.55 |
| MCRB(1024,4) | 0.17 | 1 | 0.31 | | 1.13 |
| Omega(1024,4) | 0.69 | 0.28 | 0.56 | | 0.87 |

Table 5.5: The Pareto optimal MINs for throughput-universality-complexity and the UPF of a MIN sample

The analysis of this table will be given here as well as a comparison of some of the networks with their performance given on figures 5.11 and 5.12. The analysis of table 5.5 shows that MCRB(256,4) has the smallest UPF, and thus can be considered as the best among the studied MINs. This result has been already noted with the previous evaluations.

Because of their high complexities, the MCRB(512,8) and MCRB(1024,4) which seemed to be very powerful in the case of the universality-throughput UPF are among the less powerful MINs in this inclusive UPF case. Note that the MCRB(1024,4) was not as bad as it seems in this case for the case of the complexity-throughput UPF case.

As for the analysis from a Pareto optimality point of view, we notice that the Pareto optimal MINs for this inclusive case are at the intersection of the Pareto optimal MINs found for the previous two cases. This is a direct result of the Pareto optimal solution definition.

### 5.4.4 Conclusion

In this section, the UPF was used for the comparison of different Omega and MCRB MINs. It was used as a distance function for the comparison of MINs, as a utility to find optimal MINs, and also as a criterion to make a choice from the group of optimal solutions.

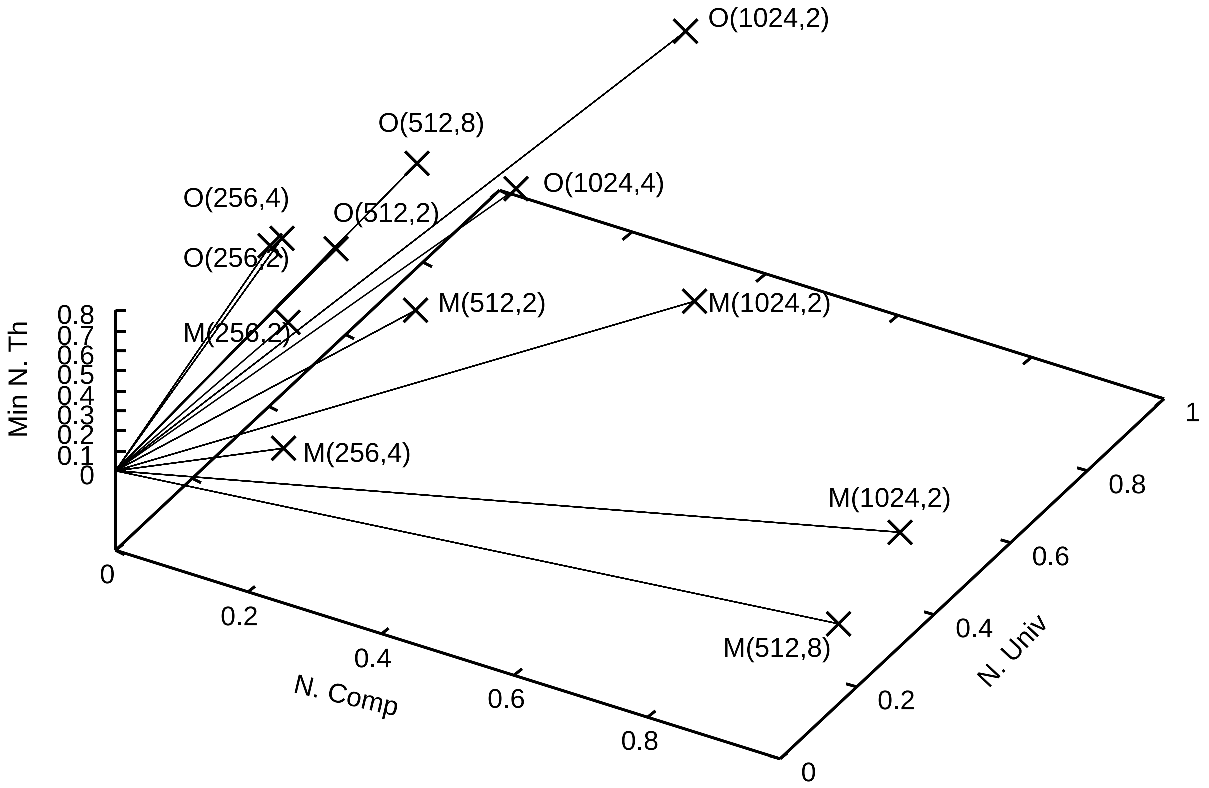

Figure 5.13: 3-dimensional representation of the UPF

When Pareto optimality was considered, almost all optimal MINs were MCRB MINs of different sizes and degrees. The Pareto optimal Omega network was among the optimal solutions thanks to its relatively low complexity with respect to the considered networks. When using the UPF as a utility in order to chose a MIN from the optimal solutions, this Omega network can be eliminated. In fact, the very high UPF value means that one of the considered metrics is relatively bad, and thus, the solution can be eliminated.

In following sections, the UPF is to be used in order to evaluate and compare some specific metrics for MINs. These are the effect of the degree on the performance of a MIN and the MINs scalability. The UPF in these cases will not studied as a Pareto optimal utility because the goal of this study is to find *a best* value among different suggested solutions. The study, thus, is restricted only to the distance evaluation of the UPF.

## 5.5 The effect of the degree of MINs on their performance

As stated previously, a multistage interconnection network is usually defined by its topology, routing algorithm, switching strategy, and flow control mechanism. One of the factors defining the topology of a multistage interconnection network is its degree. Today's technological progress makes it possible to build and use crossbars of size up to 128. Such crossbars can be used as SEs in parallel architectures such as multistage interconnection networks. In this section, the effect of the SE size on the performance of the MIN is studied on the Omega and MCRB MINs.

Networks of degrees 2, 4, and 8 were studied by Cheemalavagu and Malek in [20]. In their paper, they were limited to 8 degree MINs because of the space and time needed for the simulation. Furthermore, they used networks with and without buffers. In this dissertation we investigate MINs of degrees of up to 64, all of them unbuffered. In order to carry out this study, Delta and over-sized Delta MINs of different degrees are tested.

The MINs tested are the Omega and the MCRB networks. Two stages MCRB networks will not be tested as they are non-blocking and more complex than the crossbar. BPC (Bit Permute Complement) permutations are used as work loads [60, 69].

### 5.5.1 Universality

Concerning universality, the results that we got correspond to those found by Cheemalavagu and Malek [20]. Figure 5.14 shows that the universality of networks of degree 4 is better than that for those of degree 2 and 8 for 64 size SW-Banyans. Figure 5.15 shows that, always, for over-sized Delta networks less universality is obtained with crossbars of bigger degrees. The same figure shows that there is an optimal value of degree for the Omega network that gives the best universality. This can be explained using figure 5.16 as follows: The figure shows that according to the acceptance probabilities formulas presented above, the acceptance probability of the MIN increases for higher degree, while the acceptance probability of a crossbar decreases. Thus, a certain balance between the two probabilities is maintained for some degree values. When the conflict probability reaches to a high level larger universality is obtained.

This explains the existence of the optimal value for Delta networks. On the other hand, as the number of crossbars in an over-sized Delta network is $r$ times the number of crossbars in a Delta network, this limit is not reached and the universality of the network is always better when using crossbars of bigger sizes.

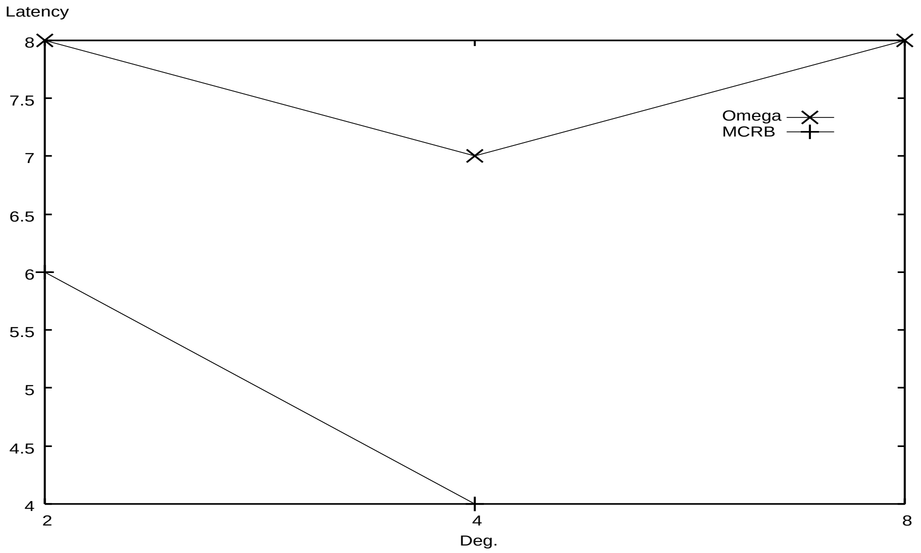

Figure 5.14: Universality of 64 size networks as a function of the degree

## 5.5.2 Throughput

Figure 5.17 shows the throughput of different size networks as a function of their degree. The figure shows that for over-sized Delta networks, throughput is bigger when larger size crossbars are used. However, Delta networks have, once again, an optimal degree value, which is 2 for size 64 networks, 2 and 4 for size 1024 and 4 for 4096 size networks.

Note that when a very big number of messages arrives at the first stage of a network, using our approach to calculate the throughput, a considerable number of messages are discarded. No conflicts can occur at the last stage as the destinations are groups of permutations. This means that conflicts can occur only on a limited number of stages in large degree MINs which might explain the slight augmentation of the throughput of Omega(4096,64) as related to Omega(4096,16).

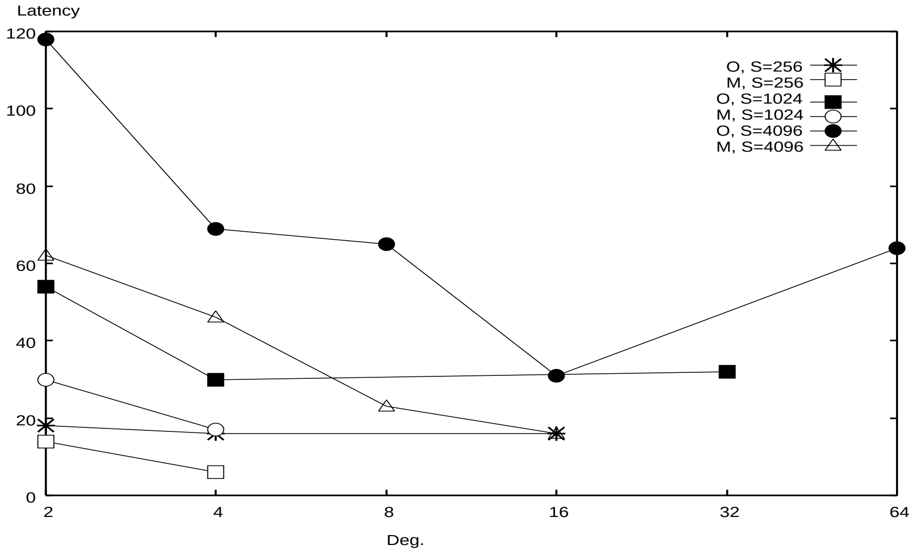

Figure 5.15: Universality as a function of the degree

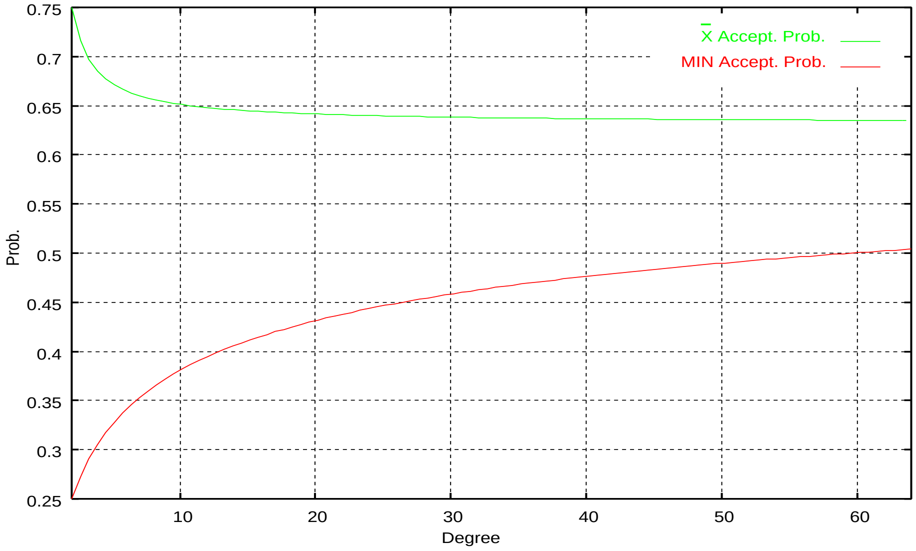

Figure 5.16: The crossbar and network acceptance probability for 4096 size Delta network

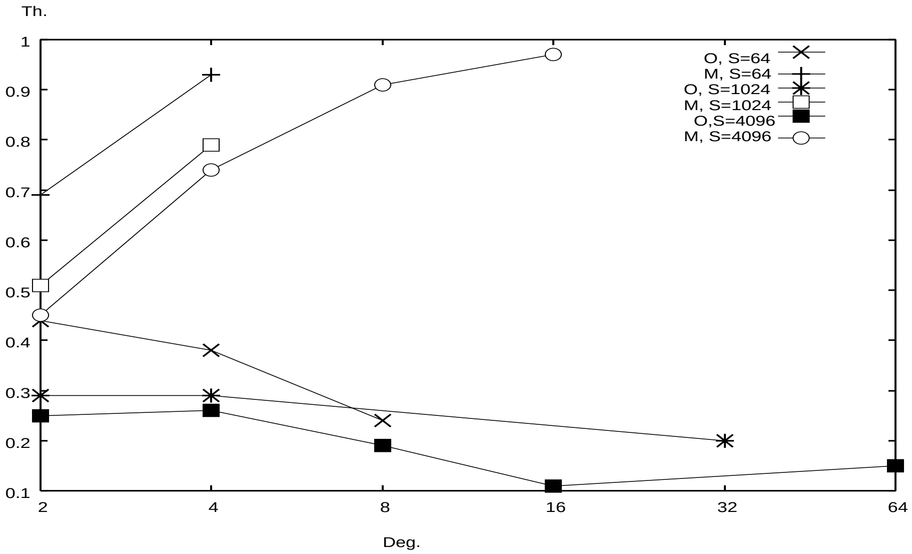

Figure 5.17: Throughput as a function of degree

### 5.5.3 Complexity and throughput UPF

Figure 5.18 shows a complexity-universality UPF plot. Note that on this figure as in the previously mentioned approach, the performance values of the Delta network have an optimal degree value. For small size networks, small degrees give the best results, while for networks of size 1024, crossbars of size 2 and 4 give the same best performance. However, with 4096 size network the optimal degree value is 4. On the other hand, an optimal degree value can be noted for the MCRB networks of size 4096, which is $r = 4$.

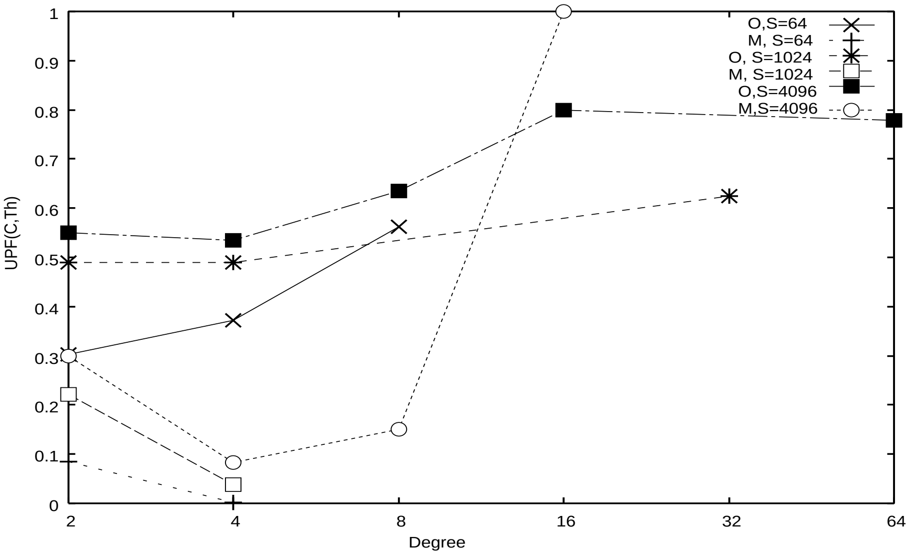

Figure 5.18: Complexity-throughput UPF as a function of degree

### 5.5.4 Universality and throughput UPF

Remember that complexity plays, when considered, an important role in degrading the network's performance in relation to the complexity-throughput UPF case. This is because over-sized Delta networks' complexities increase very rapidly with an in increase in degree.

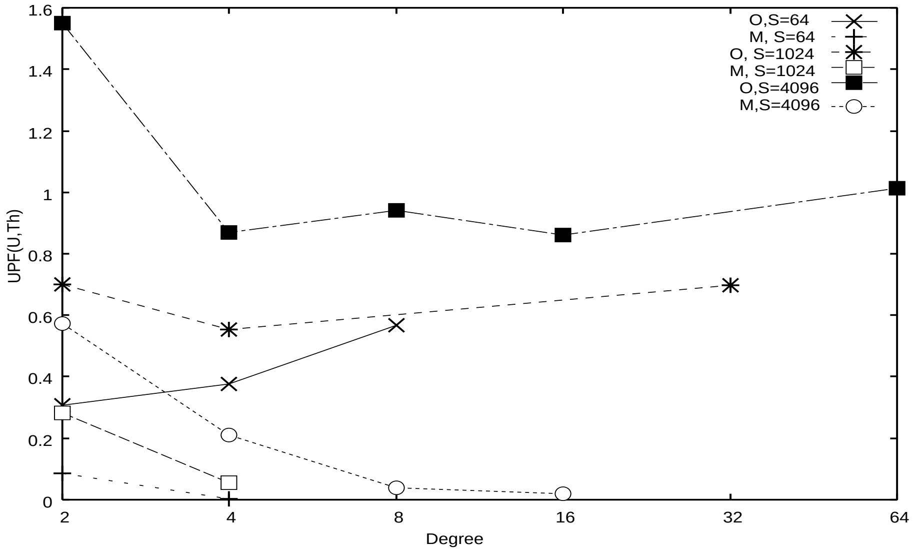

Figure 5.19: Universality-throughput UPF as a function of degree

On figure 5.19 the 4096 size Omega networks show more than one optimal degree value. In fact, it gives the same UPF value for two different degrees. In multi-dimensional evaluations, this is expected, as for different metrics different degrees are optimal for a certain size MIN. For this special case, i.e. 4096 size Omega networks the best throughput value is obtained for $r = 16$ and the best universality value is obtained for $r = 4$. Two optimal values can thus be obtained, for the case when both metrics are evaluated.

### 5.5.5 3 criteria UPF

Figure 5.20 presents the comparison and evaluation of a number of networks as related to the three factors we are considering as examples, i.e. complexity, universality and throughput.

Once more, we can observe the important effect of complexity on the overall system performance. This can easily be seen by comparing figures 5.20 and 5.18, in which the curves have almost identical shapes (the shapes are more different when complexity is not considered, see figure 5.19).

However, the figure shows that some optimal degree values of this case can be different from those obtained in the case of complexity-throughput UPF. For

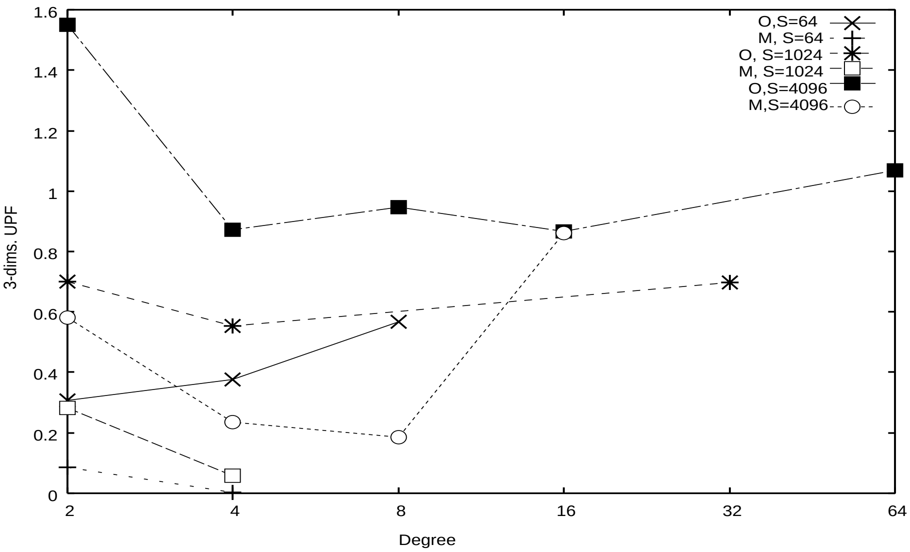

Figure 5.20: Inclusive UPF as a function of the degree

example, in the case of MCRB-MINs of size 4096 the optimal degree value is 8 while it was 4 when only complexity and throughput were considered.

### 5.5.6 Conclusion

In this section the effect of the degree of a MIN on its performance was studied. The results that we got for the studied MINs confirmed results obtained in [20]. These results stated that in general an optimal degree value exists for which the collective evaluation of certain popular metrics gives the best results.

## 5.6 Scalability

The speedup of an algorithm on a certain parallel architecture is defined as the ratio of the algorithm's best serial execution time to its parallel execution time on the architecture under consideration. Efficiency is the ratio of speedup to the parallel machine size.

While many scalability evaluation studies refer to the relationship between the size of the resources of a parallel system and its speedup, efficiency, and/or utilization, our proposition links it to the UPF.

Here we are studying the scalability of MINs in parallel machines. Any desired number of measurable MINs performance metrics can be used in the scalability evaluation. Also, the increase of workload in proportion to size in order to maintain the same efficiency is taken into consideration. This is an important condition to respect when studying scalability [41].

As we are dealing with MIN scalability, we are interested in hardware related scalability. Note that the scalability dependency on software is an important issue as a system can be scalable for a certain number of algorithms and non-scalable for other ones. Only one communication pattern is tested in this section and the presented scalability results correspond only to this communication pattern.

**Proposition 8.** *The UPF is a suitable metric for the evaluation of scalability of MINs.*

*Proof.* Nussbaum and Agarawal [71] listed some requirements for a "*useful definition of scalability*". We will discus the compatibility of the UPF with these requirements as well as with the general definition of scalability. First, the UPF is a performance metric which can link performance to cost (complexity). Not only the size of the MIN is considered, but a lot of architectural characteristics can be considered, which gives a better idea about the scalability of a MIN.

Second, by definition, the UPF is an algorithm dependent performance factor as it is a measures based metric. It also takes into consideration the necessary increase in workload with the increase in size of the MIN. Still it does not depend on efficiency, cost-effectiveness, or even speedup. It only compares the performance of several different architectures.

On the other hand, physical constraints as defined in [71] are not studied, as our study does not deal with wiring issues.

In the following, we will study the scalability of two families of MINs, the Omega MINs, a member of the Delta MINs, and the MCRB network.

### 5.6.1 Universality Scalability

It is obvious that for an increasing workload, the universality of a bigger size network is greater. Thus, measuring the universality scalability of a MIN can be defined as a scalability less tendency to increase. In other words, better scalability corresponds to less unscalability. Figure 5.21 shows the scalability of Omega and MCRB networks of different degrees. It is clear that the MCRB network is more scalable than the Omega network, i.e. the tendency of the universality to increase is less than that of Omega networks. We observe that networks of greater degrees are more scalable.

### 5.6.2 Throughput scalability

Greater sizes networks can transfer a bigger number of messages. However, because of a greater conflict probability, MINs of higher sizes have smaller throughputs. Once again, we are facing a case of a performance metric that is supposed to be maximized for a scalable network, while in fact bigger size networks have smaller throughputs. Figure 5.22 presents the throughput of a number of MINs as a function of their sizes. In this figure, all networks have almost linear scalability. However, the decreasing tendencies of MCRBs and Omegas of degree 4 seem to be better than for the other two networks.

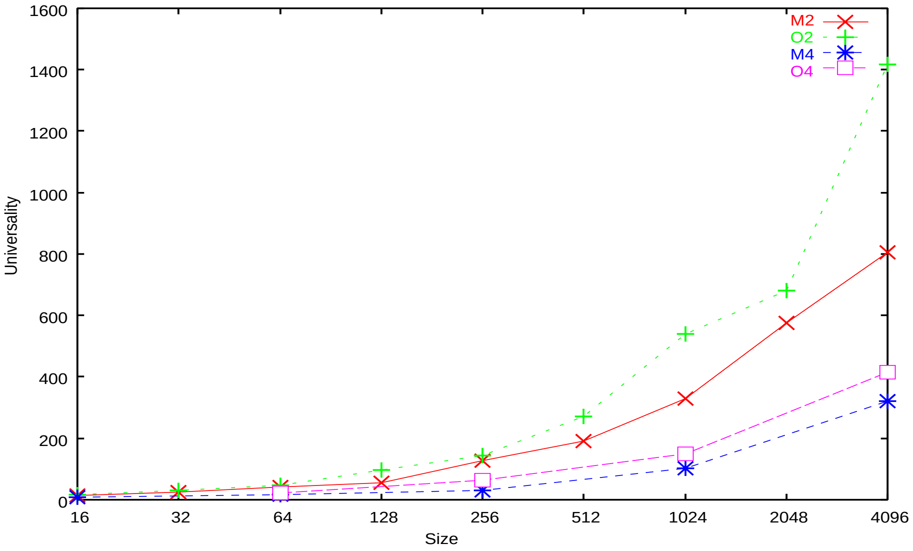

Figure 5.21: The universality scalability of some MINs

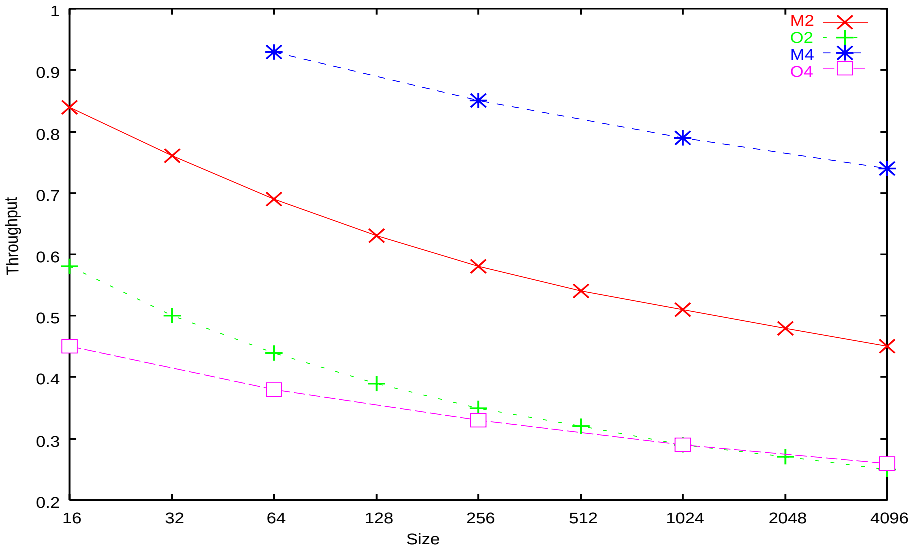

Figure 5.22: The throughput scalability of some MINs

### 5.6.3 Complexity scalability

As shown in figure 5.23 and considering the same rule as in the previous two cases, Omega networks seem to be more scalable than MCRB networks. This is normal as the MCRB complexity increases much more rapidly than the Omega complexity as a function of the size.

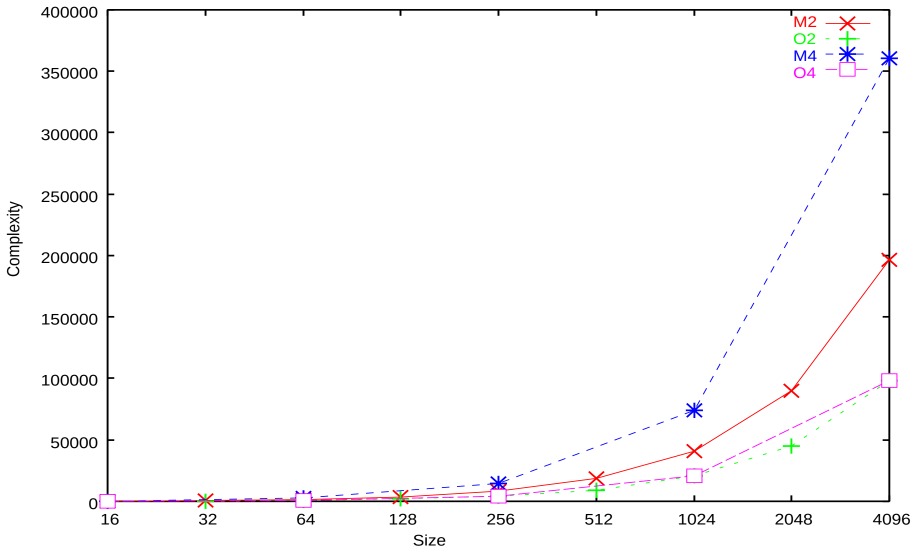

Figure 5.23: The complexity scalability of some MINs

The use of the UPF for the scalability evaluation of MINs is aimed to resolve such a paradox where different performance metrics give different results or in other words where factors scale differently when a sample of networks is considered.

### 5.6.4 Multiple-criteria scalability

In this section, more than one performance metric will be considered in order to evaluate the performance and thus the scalability of the networks.

Figures 5.24, 5.25, and 5.26 show the performance evolution of the two 2-dimension and one 3-dimension UPFs for a number of MINs. Note that the UPF by its formula given in equation 3.5 is considered as a factor to be minimized.

The figures show that for the different tests, the evolution has the same shape, especially for small size MINs. In fact, this is due to the domination of the normalized values of the throughput of the networks for these small size networks. It is certain that this effect could be compensated by giving the throughput a lower weight (less importance) than other factors.

On the other hand, the Omega MIN of degree 4 seems to have the most linear performance degradation. In other words, among the tested networks, Omega MINs of degree 4 constitute the most predictable architecture for the construction of larger size networks. The fact that $\Omega(4096, 4)$ has the least inclusive UPF value supports this conclusion.

Note that for both 2-dimension cases, the MCRB network of smaller degree gives a more acceptable scalability than larger degree networks.

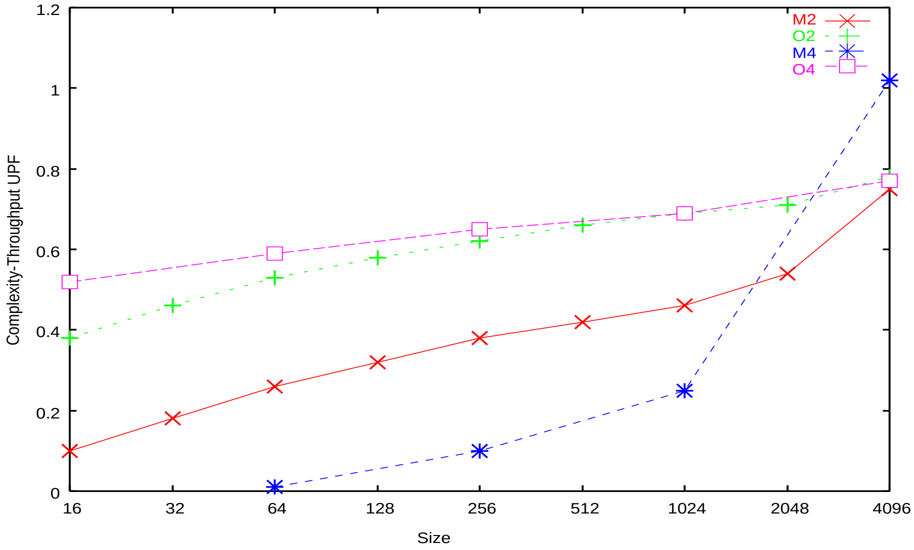

Figure 5.24: The complexity-throughput UPF scalability of the studied MINs

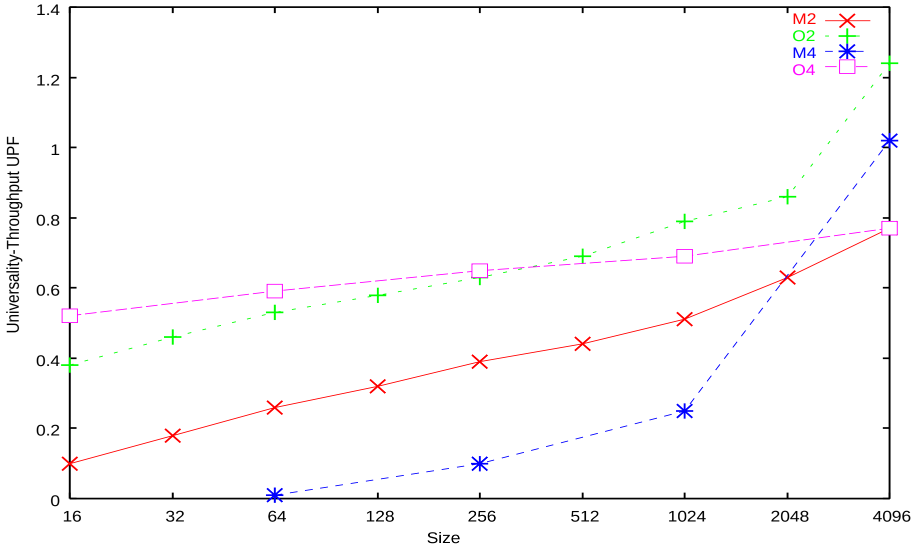

Figure 5.25: The universality-throughput UPF scalability of some MINs

Note that while the MCRB showed better scalability for one-dimension cases of throughput and universality evaluations, in the case of higher dimension evaluations, Omega network performance seems to be better.

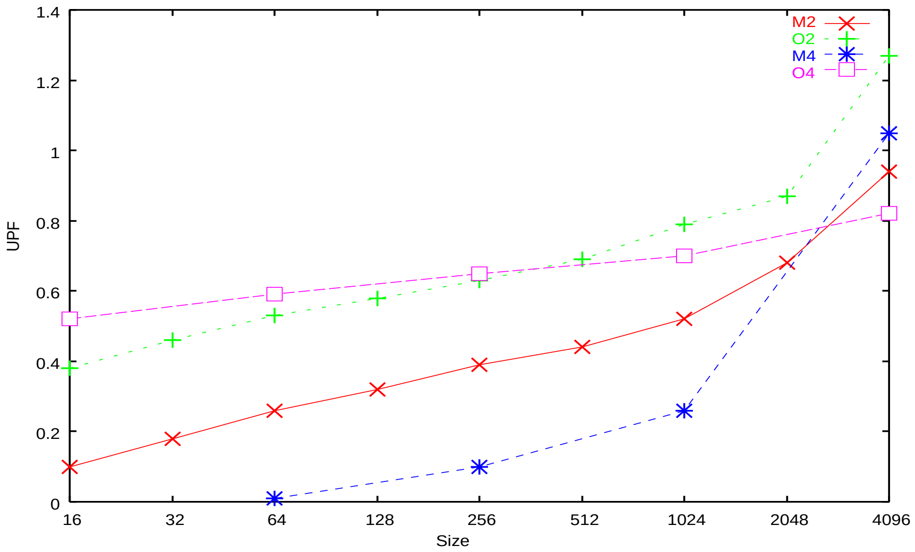

Figure 5.26: A 3-dimensions UPF scalability of some MINs

### 5.6.5 Conclusion

In this section the UPF was used for the evaluation of the scalability of MINs. Even if it is known that MINs are not scalable for scalable workload, their performance evolution for such cases as a function of the size, cost and other scalability parameters is an important factor for performance prediction and network characterization. Thus, it was proved that the UPF is a suitable metric for scalability evaluation and it was used in order to evaluate the scalability of two MIN families. When complexity, throughput and universality were considered together, Omega networks were shown to be more scalable than MCRB networks, when both scalability linearity and tendency were considered.

## 5.7 Conclusion

In this chapter, the methodology proposed in chapter 3 is applied to on Omega network, member of the Delta-MINs and to MCRB network, member of the oversized Delta networks proposed in chapter 4. The proposed methodology, based on a distance function called the UPF was applied to example networks in order to establish a general purpose comparison, and then to study the effect of the degree of considered MINs on their performance, and finally to evaluate their scalability.

# Chapter 6

# Conclusion

## 6.1 Summary of the dissertation

This thesis discusses two main aspects: the multi-criteria performance evaluation of multistage interconnection networks and the proposition of a new class of this family of interconnection networks.

In the literature, a lot of work has been devoted to the performance evaluation of communication systems in parallel computers. One family of these communication systems is the interconnection networks, and more precisely multistage interconnection networks. Multistage interconnection networks constitute an important communication system in parallel machines due to their interesting characteristics such as low complexity, acceptable throughput, constant general communication time in case of conflict absence, etc.

However, almost all the work done in this domain restricted the performance to one performance metric, or to different performance metrics but evaluated separately. A few studies considered the problem as a multi-criteria problem, i.e. tried to evaluate the network considering several metrics at a time. Two examples were presented in this dissertation with a discussion of each of them. One example, which is that published by Cheemalavagu and Malek[20] considered the throughput latency ratio and was restricted to this ratio, i.e. while a lot of of performance factors might interfere in the performance evaluation of interconnection networks, this study was restricted to only two metrics. On the other hand, the multi-level filters methodology, proposed in [57] is a synthesis procedure that seemed to need more formalization.

The constant need for more powerful communication systems for parallel computers,the attractive characteristics of multistage interconnection networks and the application possibilities in recent architectures such as SMP machines and NoCs are the bases of the work presented in this dissertation. A new family of multistage interconnection network has been proposed. A multi-criteria performance evaluation methodology has been introduced and multi-criteria performance evaluations of two multistage interconnection network families have been established.

In chapter 2 we tried to place our work in the context of research work on parallelism and parallel architectures. In fact, while SMPs in general are still using crossbars and / or buses as communication systems, and while these architectures are not always enchanting solutions for communication systems in parallel computers, some NoCs are based on multistage interconnection implementations. A survey of multistage interconnection networks was given, and the main aspects of multistage interconnection networks as well as some important examples were discussed in some detail. This was followed by the main context of the problem this dissertation is dealing with that is, the performance evaluation of multistage interconnection networks with respect to different performance metrics, in other words, the multi-criteria performance evaluation of these networks. The above mentioned examples found in the literature were discussed at the end of the chapter.

In this dissertation the proposed solution to this problem was the consideration of the multistage interconnection network performance evaluation problem as a multiple-criteria decision making problem and was introduced in chapter 3. Because of the conflicting nature of the solutions to this kind of problem (found in almost every field of research), solving multiple-criteria decision making problems is more difficult than studying single criterion problems. In general the solution to these problems consists at transforming the problem into a single-objective one. This is done by finding a function that represents the multidimensional values of each solution. While this function is not always easy to find, we propose, and prove the possibility of the use of a distance function in the case of the performance evaluation of multistage interconnection networks. We call this function the universal performance factor. This function is based on measured values of different performance metrics and obtained from a simulation and can be used either as a distance function for the comparison of different architectures or as a a solution of a multi-criteria Pareto optimization problem. The simulation tool was discussed in detail. Its architecture, the simulated workload and the considered simplifying assumptions. For simplicity, most of the analysis of this thesis assumed circuit switching routing. The chapter was ended by listing the performance factors tested as examples for the evaluation procedure.

The second main aspect discussed in the dissertation is the proposition of a new multistage interconnection network family. This is presented as an improvement technique of Delta networks. Many improvement techniques were proposed in the literature, some of them are listed at the beginning of chapter 4. Then we introduced the technique that we propose, which is the over-sizing of the network. The general architectural presentation of the over-sized Delta networks is presented along with its characteristics, and the MCRB network is given as an example of this new class. Two special cases of the MCRB network are presented at the end of the chapter, one is an interesting special case of the network, which is when its size is a square of the degree. While this case is interesting in general, it is not the subject of this dissertation as all considered networks are less complex than the crossbar, and these networks are more complex than a crossbar of the same size. The other case is the AMCRB, which results from the application of a further improving technique on the MCRB network. This network is not, *a priori*, the subject of this dissertation as it is not a Banyan network.

The last chapter presents the results of the application of performance methodology on the case study networks that is, on the Omega networks and the MCRB networks of different architectural characteristics, i.e. of different sizes and degrees. First, the simulation tool was validated. This was done by comparing the different results obtained with the simulator with mathematical probabilistic well known results. Then two cases of evaluation and comparison of interconnection networks were presented: a general case, studied from two points of view: that of the multi-criteria optimization problem and that of the network comparisonproblem, i.e. the use of the UPF only as a distance function. The second case is the study of the effect of the degree of interconnection networks on their performance, and the final case considers the evaluation of the scalability of the networks using the UPF function.

## 6.2 Future directions

The perspectives opened by our work are numerous and various. They can be grouped into two main categories:

- Work that can be done on the proposed methodology so that it can be used for further investigations and the evaluation of more complicated systems.
- Work to be established on the proposed architecture: the over-sized Delta network as well as the MCRB network.

### 6.2.1 Open questions with respect to the evaluation methodology

**Communication patterns**

In this dissertation, only one communication pattern was tested that is, the BPC permutation pattern. This communication pattern was chosen because of its wide coverage of frequently used communication patterns for scientific application. Not only do other types of permutations have to be tested, but also special communication cases such as broadcasting and hot-spots seem to be an important issue that needs to be addressed.

**Performance metrics**

In this dissertation, a limited number of performance metrics is used for the performance evaluation and comparison of the networks. These metrics were chosen because of their wide use in the literature, representativity and wide coverage. Howeve, more metrics could be used in order to have more accurate results.

### Switching strategy and bufferization

While circuit switching strategy was used for almost all tests in this dissertation, it is known that packet switching improves the performance of interconnection networks. When packet switching is used, switching elements can be implemented with buffers used in a case of a conflict by bufferizing one or more of the messages that will be routed in the following cycles to their destinations. The effect of the buffers is very significant and when comparing two different architectures, it has to be taken into consideration.

### Architectural issues

While the proposed methodology is intended to compare only multistage interconnection networks without any exterior effect, a real implementation of a communication systems requires considering the overall computing system. Thus, issues like the effect of the cache in parallel architectures must be considered. In this case the read and write communication aspects, and thus the cache coherency must be simulated. This is important when studying a whole system because on one hand the use of a cache decreases the communication tasks on the network because of the data existing in the caches, and on the other hand, when only one network is used, the communications needed for assuring cache coherency increases this communication workload.

### Real time issues

Multistage interconnection networks are mainly characterized by a constant time of communication through the network when no conflicts occur. This is an important issue for real time systems. Thus, conflicts constitute the main problem for the use of multistage interconnection networks in real time contexts. In order to study and compare networks in this context, a certain priority can be attributed to a percentage of messages. This priority corresponds to the time limit during which the messages are supposed to reach their destinations [48]. In our case, this time will be measured as a number of interconnection cycles.

### SystemC

SystemC is a language used in order to establish low level simulations. Such simulations might be interesting for the study of networks on chips.

## 6.2.2 Open question related to the proposed network class architecture

AMCRB and MCRB($r^2$,r) networks were not studied in detail in this dissertation because of their architectural characteristics, i.e. one is more complex than the crossbar and the other is non-Banyan. However these networks have very interesting features that deserve a special study. In addition, hierarchic MCRBs and IMCRBs (Incomplete MCRB) are the subject of current work because of their interesting properties.

# Appendix: Some results concerning the AMCRB-MIN

In this appendix, the results obtained on the AMCRB networks are presented. The goal of this study is to evaluate the effectiveness of the proposed rerouting algorithm used in case of a conflict.

## 6.3 Complexity of the AMCRB-MIN

The complexity difference between AMCRB and MCRB MINs can be calculated by observing that the multiplexers of the first stage are of size $1 \times 2r$, the demultiplexers of the last stage are of size $2r \times 1$ and finally, the SEs of the other stages are of size $2r \times 2r = 4r^2$. The complexity of the AMCRB is calculated using the following equation.

$$C_{AMCRB} = 4Nr + 4\left(\frac{\log(N)}{\log(r)} - 1\right)Nr^2 \tag{6.1}$$

As in the case of the MCRB network, the crossbar forms a complexity upper limit, and networks having complexity higher than this limit are not particularly considered in this dissertation. Figure 6.1 presents the interesting complexity values of the AMCRB-MIN.

Interesting AMCRB-MINs can be found by taking cross sections on figure 6.1. This will give figures 6.2, 6.3 and 6.4.

The figures show that AMCRB-MINs having complexity smaller than a corresponding size crossbars are of size $N \geq 128$ for degree 2 and $N \geq 256$ for $r = 4$. On the other hand, the complexity of the AMCRB-MIN is higher than crossbar of the same size for all values $r \geq 8$.

## 6.4 The efficiency of the proposed rerouting algorithm

A non-routable message is one that does not correspond to the conditions of the rerouting algorithm presented in theorem 2 page 65. While this is not the main

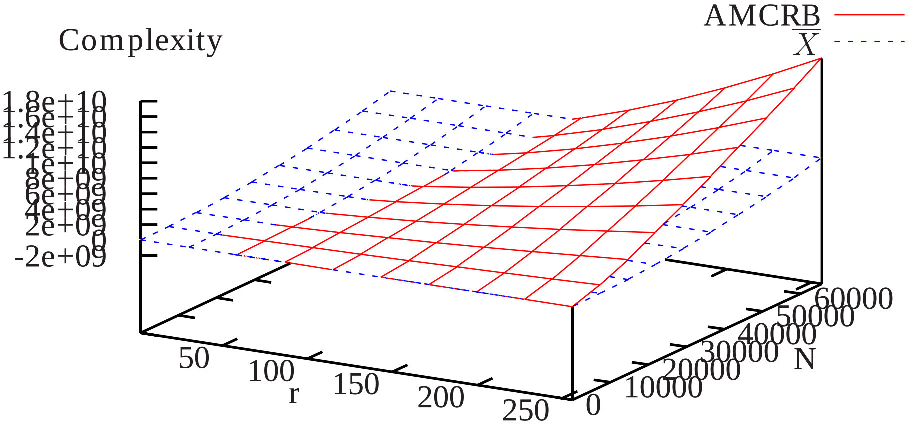

Figure 6.1: The AMCRB complexity as a function of r and $N$

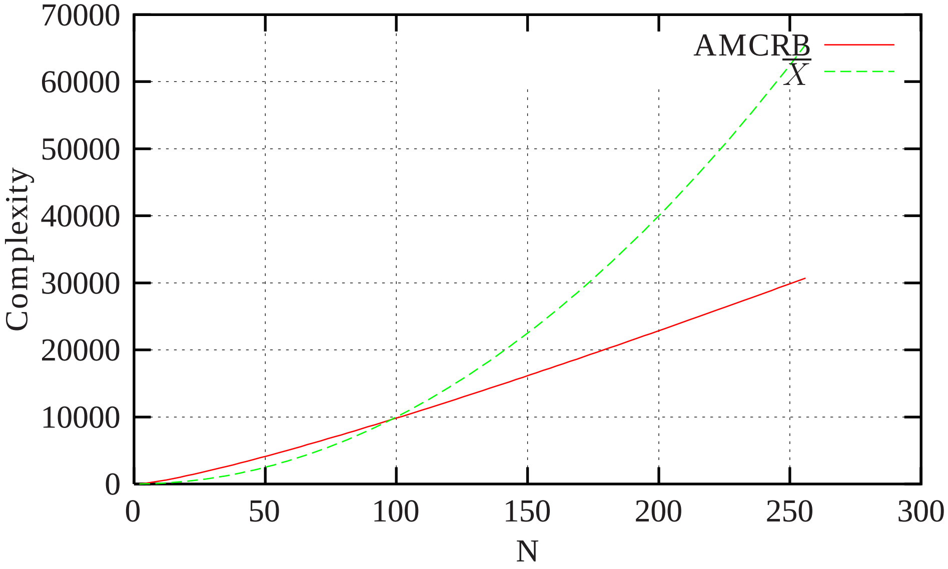

Figure 6.2: The AMCRB complexity compared to the crossbar as a function of the size and for $r = 2$

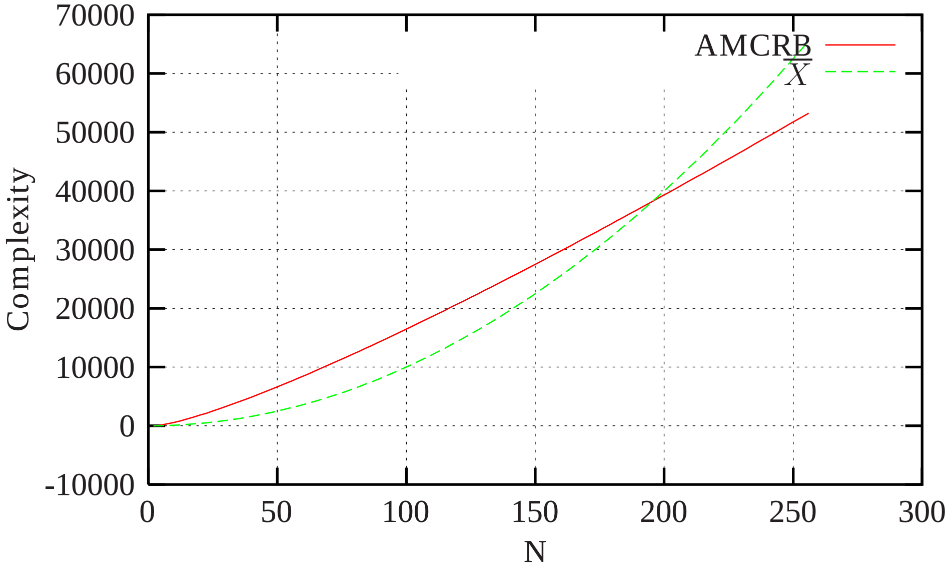

Figure 6.3: The AMCRB complexity compared to the crossbar as a function of the size and for $r = 4$

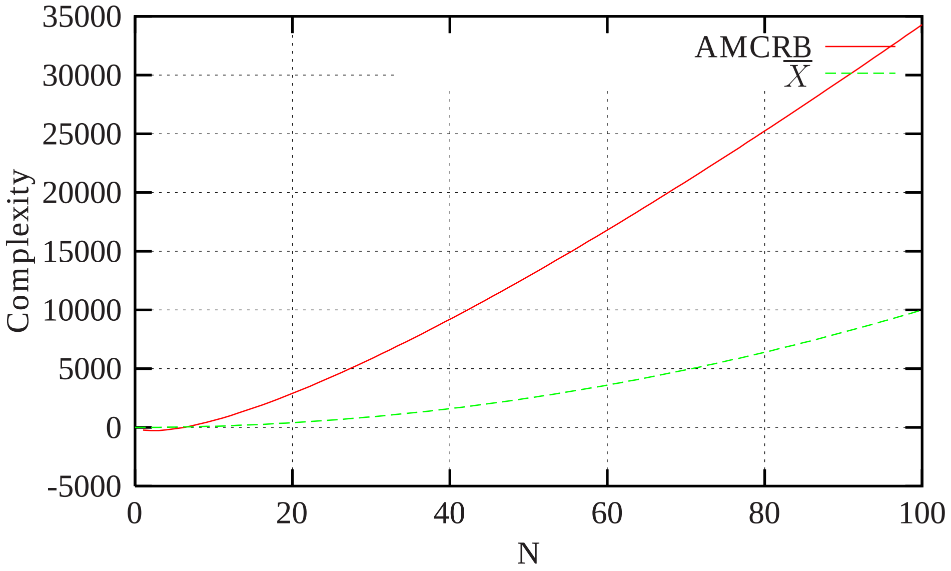

Figure 6.4: The AMCRB complexity compared to the crossbar as a function of the size and for $r = 8$

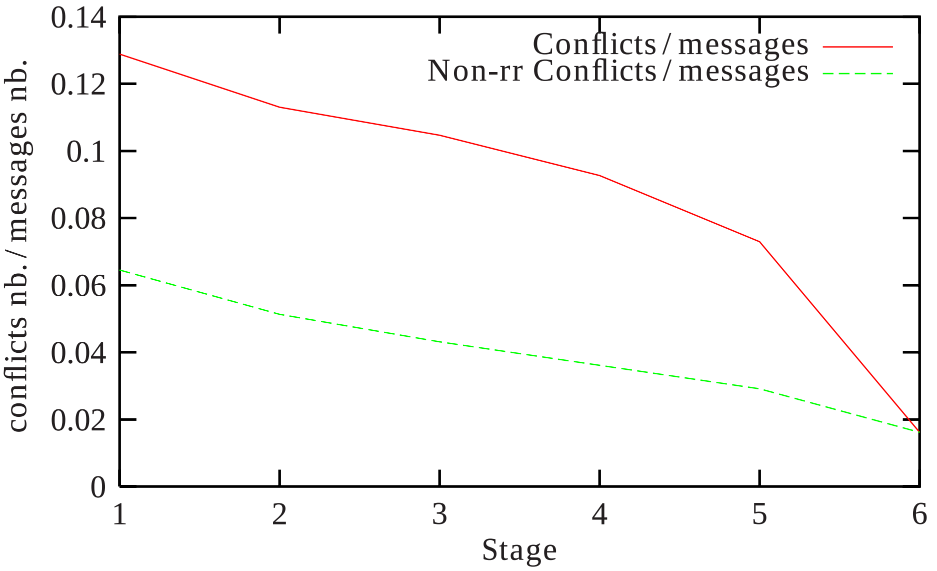

Figure 6.5: The difference between the percentages of reroutable and non-reroutable conflicts on the AMCRB(128,2) as a function of the stage

interest of this dissertation, in order to have an idea about the effectiveness of the rerouting algorithm, i.e. its capacity to reroute conflict messages, the case of an AMCRB(128,2) will be studied as an example.

Figure 6.5 shows that, in general, the percentage of conflicts decreases through the network. In fact, the decreasing role of the negative network makes the possibility of a conflict less than when messages are routed on only one network. In other words the probability of a conflict is less as messages are distributed on two separated networks.

The efficiency of the rerouting algorithm is evaluated by the fact that in the worst case, i.e. starting from the first stage, it can solve around 50% of the conflicts, so the percentage of conflicts becomes only slightly more than 6% in the considered case.

The algorithm’s capacity to solve conflicts decreases for the next stages as satisfying the algorithm conditions becomes more difficult, so that no conflicts can be solved at this stage, and because some conflicts, if any occur, can do so on the negative network.

# List of publications

- Ahmad Chadi ALJUNDI, “Performance Evaluation of the Multistage Multidimensional Ring Interconnection Network”, DEA (Diplome d’Etudes Approfondies) Dissertation, Laboratoire d’Informatique Fondamentale de Lille, Universite des Science et Technologies de Lille, 2000.

- Ahmad Chadi ALJUNDI, Jean-Luc DEKEYSER, M-Tahar KECHADI and Isaac D. SCHERSON, "Comparative Simulations and Performance Evaluation of MCRB Networks Using Multidimensional Queue Management", In Proceedings of the 2002 international Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS02), pages 288-296, July 2002, San Diego, USA.

- Ahmad Chadi ALJUNDI, Jean-Luc DEKEYSER, M-Tahar KECHADI and Isaac D. SCHERSON, "A study of an evaluation methodology for unbuffered multistage interconnection networks", In Proceedings of the 17th International Parallel and Distributed Processing Symposium (IPDPS), the Workshop on Performance Modeling, Evaluation, and Optimization of Parallel and Distributed Systems (PMEO-PDS’03), 22 -26 April, 2003, Nice Acropolis Convention Center, Nice, France.

- Ahmad Chadi ALJUNDI, Jean-Luc DEKEYSER and Isaac D. SHERSON, "An Interconnection Networks Comparative Performance Evaluation Methodology: Delta and Over-Sized Delta Networks", In Proceedings of the ISCA 16th International Conference on Parallel and Distributed Computing Systems (PDCS’03), pp. 1-8, 13-15 Aug. 2003, Reno, Nevada, USA.

- Ahmad Chadi ALJUNDI and Jean-Luc DEKEYSER, "The Effecte of the Degree of Multistage Interconnection Networks on their Performance: the Case of Delta and Over-sized Delta Networks". In Proceedings of the 2004 Euromicro on Parallel and Distributed Processing (PDP’04), A Coruna, Spain, 11-13 Feb. 2004.

- Ahmad Chadi ALJUNDI, Jean-Luc DEKEYSER and M-Tahar KECHADI, “On the Scalability of Multistage Interconnection Networks”, In Proceedings of the First International Conference on Information and Communication Technologies: from Theory to Applications (ICTTA’04), Damascus, Syria, 19-23 Apr. 2004.

- Ahmad Chadi ALJUNDI, Jean-Luc DEKEYSER, M-Tahar KECHADI and Isaac D. SCHERSON, “A Universal Performance Factor for Multi-Criteria Evaluation of Multistage Interconnection Networks”, Second reading, submitted to the Special Issue of Future Generation Computer Systems on Systems Performance Analysis and Evaluation.

# Bibliography

[1] *AIX Versions 3.2 and 4 Perfomance Tuning Guide*, chapter 1. Apr. 1996. http://nscp.upenn.edu/aix4.3html/aixbman/prftungd/toc.htm.

[2] Compaq 8-way multiprocessing architecture. Technical Report ECG058/0399, Compaq Computer Corporation, Mar. 1999.

[3] G.B. Adams III, D.P. Agrawal, and H.J. Siegel. A survey and comparaison of fault-tolerant multistage interconnection networks. *Computer*, 20(6):14–27, Jun. 1987.

[4] D. P. Agrawal. Graph theoretical analysis and design of multistage interconnection networks. *IEEE. Trans. Comp.*, C-32(7):637–648, Jul. 1983.

[5] W. Aiello, S. N. Bhatt, F. R. K. Chung, A. L. Rosenberg, and R. K. Sitaraman. Augmented ring networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(6):598–609, 2001.

[6] B. D. Alleyne. *Methodologies for Analysis and Design of Data Routers in Large SIMD Computers*. PhD thesis, Princton Univ., June 1994.

[7] B.D. Alleyne and I.D. Scherson. On evil twin newtorks and the value of limited randomized routing. *IEEE Trans. on Parallel and Distributed Systems*, 11(9), Sep. 2000.

[8] B.W. Arden and H. Lee. Analysis of chordal ring networks. *IEEE Transactions on Computers*, 30(4):291–295, April 1981.

[9] B.W. Arden and K.-T. Tang. Routing for generalized chordal rings. In *Proc. ACM 18th Computer Science Conf.,*, pages 271–275, February 1990.

[10] Amer Baghdadi. *Exploration et conception systématique d'architectures multiprocesseurs monopuces dédiées à des applications spécifiques*. PhD thesis, Institut National Polytechnique de Grenoble, 2002.

[11] L. Barriere, J. Fabrega, E. Simo, and M. Zaragoza. Fault-tolerant routings in chordal ring networks. *Networks*, 36(3):180–190, 1999.

[12] L. Barriere and M. Mitjana. Gossiping in chordal rings under the line model. In *Workshop on communications of the 23ed symp. on mathematical foundations of computer science*, pages 37–47, 1998.

[13] V.E. Benes. *Mathematical theory of connecting networks and telephone traffic*. Academic press, 1965.

[14] L.N. Bhuyan, Q. Yang, and D.P. Agrawal. Performance of multiprocessor interconnection networks. *IEEE Computer*, 22(2):25–37, Feb. 1989.

[15] A. Borodin and J.E. Hopcroft. Routing, merging, and sorting on parallel models of computation. In *14th Annual ACM Symp. on Theory of Computing*, pages 338–344, 1982.

[16] P. Budnick and D.J. Kuck. The organization and use of parallel memories. *IEEE Trans. Comput.*, C-20:1566–1569, Dec. 1971.

[17] J.R. Burke, C. Chen, T. Lee, and D.P. Agrawal. Performance analysis of single stage interconnection networks. *IEEE Trans. on Comp.*, 40(3):357–365, Mar. 1991.

[18] D. G. Cantor. On non-blocking switching networks. *Networks*, 1(4):367–377, 1971.

[19] R. Carl and J.E. Smith. Modeling superscalar processors via statistical simulation. In *Workshop on Performance Analysis and its Impact on Desgin*, Jun. 1998.

[20] S. Cheemalavagu and M. Malek. Analysis and simulation of banyan interconnection networks with 2x2, 4x4 and 8x8 switching elements. In *Proc. Real-Time Syst. Symp.*, pages 83–89, 1982.

[21] C. Clos. A study of non-blocking switching networks. *Bell system tech. journal*, 32(2):406–424, Mar. 1953.

[22] G.R. Couranz, M.S. Gerhardt, and C.J. Young. Programmable radar signal processing using the rap. In *Proc. 1974 Sagamore Comput. Conf. Parallel Processing*, pages 37–52, 1974.

[23] H.S.M. Coxeter. Self-dual configuration and regular graphs. *Bulletin of the American Mathematical Society*, 56:413–455, 1950.

[24] D. E. Culler, J. P. Singh, and A. Gupta. *Parallel Computer Architecture (A Hardware/Software Approch)*, chapter Interconnection Network Design. Morgan Kaufmann Publishers, 1999.

[25] N.J. Davies. *The performance and scalabaility of parallel systems*. PhD thesis, University of Bristol, 1994.

[26] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, 2001.

[27] F. Della Croce, A. Tsoukiàs, and P. Moraïtis. Why is difficult to make decisions under multiple criteria. In *Proc. Sixth International Conference on AI Planning and Scheduling (AIPS'02) Workshop on Planning and Scheduling with Multiple Criteria*, pages 41–45, Toulouse, France, 2002.

[28] D. M. Dias and J. R. Jump. Analysis and simulation of buffered delta networks. *IEEE Trans. Comput.*, C-30(4):273–282, Apr. 1981.

[29] R. Duncan. A survey of parallel computer architectures. *IEEE Computer*, 23(2):5–16, Feb. 1990.

[30] L. Eckhout, K. De Bosschere, and Henk Neefs. Performance anaylsis through synthetic trace generation. In *IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS-2000)*, pages 1–6, Apr. 2000.

[31] P.-J. Erard and P. Degenon. *Simulation par événements discrets (Concepts et réalisation en Simula, Ada et Smalltalk)*. Presses polytechniques et universitaires Romandes, 1996.

[32] T. Feng. A survy of interconnection networks. *Computer*, 14(12):12–27, 1981.

[33] T.Y. Feng. Data manipulatin functions in parallel processors and their omplementations. *IEEE Transactions on computers*, C-23(3):309–318, Mar. 1974.

[34] M.J. Flynn. Some computer organizations and their effectivenes. *IEEE Trans. Comput.*, C-21(9):948–960, Sep. 1972.

[35] F. Giacomini et al. Evaluation report on the performance of commercial components and the low-level sci software. Technical report, Software Infrastructure fro SCI (SISCI), Oct. 1999. http://sci.web.cem.ch/SCI/WP2/D.2.4.2_Components.pdf.

[36] J. Gittoes, R. Barker, D. Lacan, T. Sinou, and C.C. Kion. *RS/6000 SMP Enterprise Servers Architecture and Implementation*. IBM Corporation, 2 edition, Dec. 1996.

[37] G.R. Goke and G.J. Lipovski. Banyan networks for partitioning multiprocessor systems. In *Proc. 1st Annu. Symp. Comput. Arch.*, pages 21–28, 1973.

[38] A. Gottlieb. An overview of the nyu ultracomputer project. Technical report, The NYU Ultracomputer project, Oct. 1987.

[39] A. Gottlieb, R. Grishman, C.P. Kruskal, K.P. McAuliffe, L. Rudolph, and M. Snir. The nyu ultracomputer–designing a mimd, shared-memory parallel machine (extended abstarct). *IEEE Trans. on Computers*, 32(2):175–189, Feb. 1982.

[40] P. Guerrier and A. Greiner. A generic architecture for on-chip packet-switched interconnections. In *Proc. of the conference on the design, automation and test in Europe*, pages 250–256, 2000.

[41] J.L. Gustafson. Reevaluating amdahl's law. *Commun. ACM*, 31(5):532–533, May 1988.

[42] D.T. Harper and J.R. Jump. Performance evaluation of reduced bandwidth multistage interconnection networks. In *Proc. 14th Annual Symp. on Computer Architecture*, pages 171–175, 1987.

[43] L.C. Higbie. The omen computer: associative array processor. In *Compcon 72, IEEE Comput. Soc. Conf. Proc.*, pages 287–290, Sep. 1972.

[44] L. Hu and I. Gorton. Performance evaluation for parallel systems: a survey. Technical Report UNSW-CSE-TR-9707, School of computer science and engineering, University of NSW, Sydney, Australia, 1997.

[45] K. Hwang and F.A. Briggs. *Computer Architecture and Parallel Processing, 5th printing*. McGraw-Hill series in computer organization and architecture. McGraw-Hill International Editions, 1989.

[46] K. Hwang and Z. Xu. *Scalable Parallel Computing, Technology, Architecture, Programming*. WCB McGraw-Hill, 1998.

[47] A. Jantsch and H. Tenhunen. *Networks on Chip*. Kluwer Academic Publishers, 2003.

[48] J. Jonsson and J. Vasell. A comparative study of methods for time-deterministic message delivery in a multiprocessor architecture. In *Processdings of the IEEE International Parallel Processing Symposium (IPPS)*, pages 392–398, April 1996.

[49] M. T. Kechadi. *Un Modele de Fonctinnement Desordonne Pour les Systemes Multiprocesseurs Pipelines Vectoriels a Mémoire partagees (Definition, Modelisation et Proposition d'Architecture)*. PhD thesis, Universite des Sciences et Technologie de Lille, Laboratoire d'Informatique Fondamental de Lille, Mar. 1993.

[50] M. T. Kechadi. Mcrb: A new interconnection network for multiprocessor systems. In *Misc. Papers, CD-ROM of the 2002 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'02), ISBN: 1-892512-39-4, Las Vegas, USA*, June 2002.

[51] K. Keeton, D.A. Patterson, Y.Q. He, R.C. Raphael, and W.E. Baker. Performance characterization of a quad pentium pro smp using oltp workloads. In *Proc. of the Int'l Symp. on Computer Architecture, Barcelona, Spain*, pages 15–26, Jun. 1998.

[52] D. Krizanc and F.L. Luccio. Boolean routing on chordal rings. In *Proc. of the 2nd colloquium on structural information and communication complexity*, pages 89–100, 1995.

[53] C. P. Kruskal and M. Snir. The performance of multistage interconnection networks for mutliprocessors. *IEEE Trans. Comput.*, C-32(12):1091–1098, Dec. 1983.

[54] C.P. Kruskal and M. Snir. A unified theory of interconnection network structure. *Theoretical Computer Science*, 48:75–94, 1986.

[55] C.P. Kruskal, M. Snir, and A. Weiss. The distribution of waiting times in clocked multistage interconnection networks. *IEEE Trans. Comput.*, C-37(11):1337–1352, November 1988.

[56] D.J. Kuck. Iiiliac iv software and application programming. *IEEE Trans. on Comput.*, C-17(8):758–770, Aug. 1968.

[57] V. Lakamraju, I. Koren, and C.M. Krishna. A synthesis of interconnection networks: a novel approach. In *Proc. of the Int'l. Conf. on dependable systems and networks (DSN2000)*, pages 501–509, NY. USA, June 2000.

[58] D. A. Lawrie. Access and alignment of data in an array processor. *IEEE Trans. Comput.*, C-24(12):1145–1155, Dec. 1975.

[59] J.A. et al. Leijten. Prophid: A heterogeneous multi-processor architecture for multimedia. In *Proc. Int'l Conference on Computer Design*, 1997.

[60] J Lenfant. Parallel permutations of data: A benes network control algorithm for frequently used permutations. *IEEE Trans. Comp.*, C-27:637–647, July 1978.

[61] T. Lin and L. Kleinrock. Performance analysis of finite-buffered multistage interconnection networks with a general trafic pattern. In *Proceedings of the 1991 ACM SIGMETRICS Conferance on measurement and modeling of computer systems*, volume 19, pages 68–78, 1991.

[62] K.J. Liszka, J.K. Antonio, and H.J. Siegle. Problems with comparing interconnection networks, is an alligator better than an armadillo? *IEEE Concurrency*, 5(4):18–28, October-December 1997.

[63] Y.-S. Liu. *Architecture and performance of processor-memory interconnection networks for MIMD shared memory parallel processing systems*. PhD thesis, New York University, 1990.

[64] E. Lu and S. Q. Zheng. Parallel algorithms for controlling group switches. In *Proceedings of the 15th ISCA International Conference on Parallel and Distributed Computing Systems*, pages 84–89, Sep. 2002.

[65] R. T. Marler and J.S. Arora. Survey of multi-objective optimization methods for engineering. www.ccad.uiowa.edu/ tmarler/rtm/img/survey.pdf.

[66] A. Merchant. *Analytical Models for the Performance Analysis of Banyan Networks*. PhD thesis, Stanford University, 1991.

[67] D. Nassimi and S. Sahni. An optimal routing algorithm for mesh-connected parallel computers. *Journal of the ACM*, 27(1):6–29, Jan. 1980.

[68] D. Nassimi and S. Sahni. A self routing benes network. In *Proceedings of the 7th annual symposium on Computer Architecture*, pages 190–195, La Baule, United States, 1980.

[69] D. Nassimi and S. Sahni. A self-routing benes network and parallel permutation algorithms. *IEEE Trans. Comput.*, C-30:332–340, May 1981.

[70] D.B. Noonburg and J.P. Shen. A framework for statistical modeling of superscalar processor perfromance. In *3rd IEEE Symposium on High-Performance Computer Architecture (HPCA '97), San Antonio*, pages 298–309, Feb. 1997.

[71] D. Nussbaum and A. Agarwal. Scalability of parallel machines. *Communications of the ACM*, 34(3):56–61, Mar. 1991.

[72] S. Nussbaum and Smith J.E. Modeling superscalar processors via statistical simulation. In *International Conference on Parallel Architectures and Compilation Techniques (PACT'01), Barcelona, Spain*, May 2001.

[73] S. Nussbaum and J.E. Smith. Statistical simulation of symmetric multiprocessor systems. In *35th Annual Simulation Symposium*, Apr. 2002.

[74] K. Padmanabhan and A.N. Netravali. Dilated networks for photonic switching. *IEEE trans. Coomunications*, 35(12):1357–1365, Dec. 1987.

[75] Y. Pan, C. Qiao, and Y. Yang. Optical multistage interconnection networks: New challenges and approaches. *IEEE Communications Magazine*, pages 50–56, Feb. 1999.

[76] Avneesh Pant. Ongoing analysis of nt smp clustering products and technologies. Technical report, National Center ofSupercomputing Applications (NCSA), Aug. 1997. http://archive.ncsa.uiuc.edu/People/apant/NTCluster/smp.html.

[77] B. Parhami and D.-M. Kwai. Periodically regular chordal rings. *IEEE Transactions on parallel and Distributed Systems*, 10(6):658–767, Jun. 1999.

[78] J. H. Patel. Processor-memory interconnections for mutliprocessors. In *Proc. 6th Annu. Symp. on Comput. Arch. Newyork*, pages 168–177, 1979.

[79] J. H. Patel. Performance of processor-memory interconnections for multiprocessors. *IEEE. Trans. Comput.*, C-30(10):771–780, Oct. 1981.

[80] K. Pibulyarojana, S. Kimura, and Y. Ebihara. An improvement of banyan networks and 2-dilated banyan networks based on bypasses positining. *IEICE Trans. Commun.*, E83(7):1474–1487, July 2000.

[81] D.K. Pradhan and K.L. Kodandapani. A uniform representation of single- and multistage interconnection networks used in simd machines. *IEEE Trans. on computers*, c-29(9):777–790, Sep. 1980.

[82] Computational Science Education Project, editor. *Computer Architecture*. 1996. e-book, sponsored by U.S. Department of Energy,csep1.phy.ornl.gov / CSEP / PS_FILES / CA.PS.

[83] P. Rad and D. Rodi. Server consolidation: Examening the application consolidation approach. *Dell Power Solutions*, pages 88–91, Nov. 2002. http: / / www.dell.com / us / en / esg / topics / power_ps4q02-rad.htm.

[84] Ramachandani. Analysis of asynchronous concurrent systems by petri nets. Technical Report MAC-TR-120, Mass. Inst. Technology, 1974.

[85] D. Royo, M. Valero-Gatca, A. Gonzalez, and C. Mar. A methodology for useroriented scalability analysis. In *Proceedings of the IEEE Int'l Conf. on Application Specific Systems, Architectures, and Processors*, pages 304–315, 1997.

[86] I. D. Scherson. Orthogonal graphs for the construction of a class of interconnection networks. *IEEE Trans. Parallel and Distributed Systems*, 2(1):3–19, Jan. 1991.

[87] I. D. Scherson and A. S. Youssef. *Interconnection networks for high-performance parallel computers*. IEEE computer society press, 1994.

[88] J.T. Schwartz. The burroughs fmp machine. Technical Report Ultracomputer note 5, Courant Institute, New York, 1980.

[89] H.D. Shapiro. Theoretical limitations on the efficient use of parallel memories. *IEEE Trans. Comput.*, C-27(5):421–428, May 1978.

[90] H.J. Siegel. Analysis techniques for simd machine interconnection networks and the effects of processor address masks. *IEEE Trans. Comput.*, C-26(2):153–161, Feb. 1977.

[91] A. Sivasubramaniam, U. Ramachanran, and H. Venkateswaran. A comparative evaluation of techniques for studying parallel system performance. Technical report, College of Computing, Ceorgia Institute of Technology, Sep 1994.

[92] P. G. Sobalvarro. Probabelistic analysis of multistage interconnection network performance. Master's thesis, Electrical Engineering and Computer Science, MIT, Apr. 1992.

[93] Aad J. van der Steen. Overview of recent supercomputers. Technical report, Dept. of Computational physics, Utrecht University, The Netherlands, 1998.

[94] H. S. Stone. Parallel processing with perfect shuffle. *IEEE Trans. Comput.*, C-20(2):153–161, Feb. 1971.

[95] H.S. Stone. *High-performance computer architecture*. Addison-Wesley publishing company, 1987.

[96] T.H. Szymanski and V.C. Hamacher. On the permutation capability of multistage interconnection networks. *IEEE Trans. Comp.*, C-36(7):810–822, Jul. 1987.

[97] T.H. Szymanski and V.C. Hamacher. On the universality of multistage interconnection networks. In *Interconnection Networks for High-Performance Parallel Computers*, pages 73–101. IEEE Computer Society Press, 1994.

[98] C. Tocci and H.J. Caulfield. *Optical interconnection-foundations and applications*. Artech, 1994.

[99] A.H. Webster. Special features in simda. In *Proc. 1972 Sagamore Comput. Conf.*, pages 29–40, 1972.

[100] H.A.G. Wijshoff. *Data organization in parallel computers*. Kluwer Acadimic Publishers, 1989.

[101] D.L. Willick and D.L. Eager. An analytical model of multistage interconnection networks. In *Proc. 1990 ACM SIGMETRICS Conf. on Measurement and Modeling of Computer Systems*, pages 192–199, Boulder, Colorado, May 1990.

[102] W.A. Wulf and C.G. Bell. C.mmp- a multi-mini-processor. In *AFIPS Proc. Fall Joint Computer Conf.*, pages 765–777, 1972. NL.

[103] Y. Yang and J. Wang. Wide-sense nonblocking clos networks under packing strategy. *IEEE Trans. on Computers*, 48(3):265–284, Mar. 1999.

[104] Y. Yang, J. Wang, and Y. Pan. Permutation capability of optical multistage interconnection networks. *Journal of parallel and distributed computing*, pages 72–91, 2000.

[105] G.W. Zimmerman and A.H. Esfahanian. Chordal rings as fault-tolerant loops. *Discrete Applied Mathematics*, 37 / 38:563–573, July 1992.