Modelling And Design Of Concurrent Processing Algorithms

Thesis

Submitted In Partial Fulfilment For The
Requirements Of The M.Sc.Degree
(Computer Science)

By

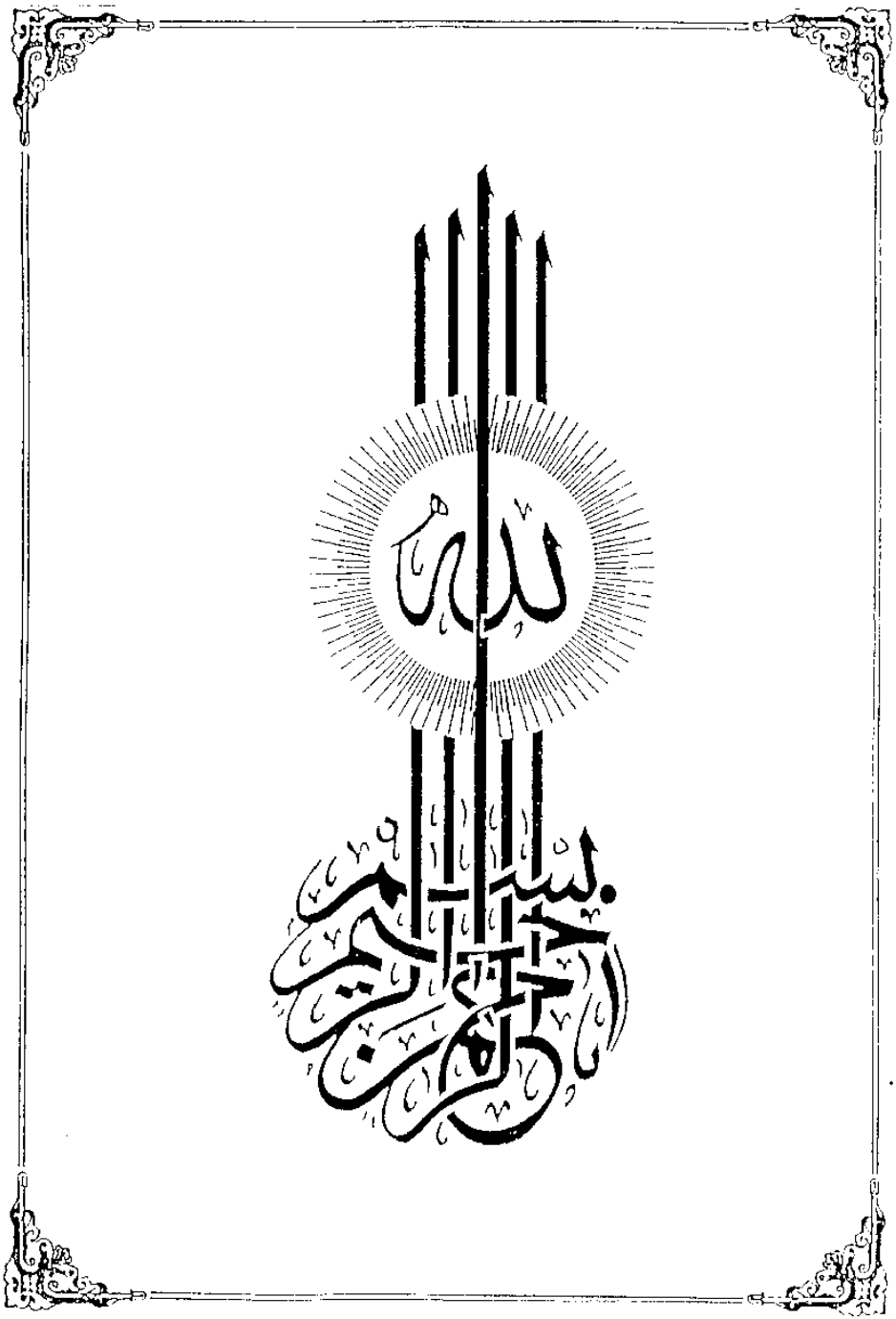
Tarek Helmi Abd El-Nabi
(B. Sc. Honor)

Supervisors

Prof. Dr. Nashat Faried M. Fathi
Prof. Of Mathematics
Department Of Mathematics
Faculty Of Science
Ain Shams University

Dr. Mostafa Sami M. Mostafa
Ass. Prof. Of Computer Science
Department Of Mathematics
Faculty Of Science
Suez Canal University

Ain Shams University
(1996)



Modelling And Design Of Concurrent Processing Algorithms

Thesis Supervision : **Approved**

1. Prof. Dr. Nashat Faried M. Fathi

2. Ass. Prof. Mostafa Sami M. Mostafa

Student's name: **Signature**

Tarek Helmi Abd El-Naby

Acknowledgment

ACKNOWLEDGMENT

Praise be to ALLAH for giving me the ability to achieve this work.

I am indebted to Dr. *Mostafa Sami M. Mostafa*, Ass. Prof. of computer science, Faculty Of Science, Suez Canal University for suggesting the problem and continuous guidance throughout this thesis. Also I am very thankful to Prof. Dr. *Nashat Faried M. Fathi*, Prof. of Mathematics, Faculty Of Science, Ain Shams University for the preparation and writing of this thesis.

Thanks are also due to all members of the Mathematics Department Of Suez Canal University for their kind help and encouragement.

My thanks also go to Dr. Ahmed Ahmed El-Shamy, Dr. Tawfik Atit-Allah and Dr. Medhat Rakha.

Finally, I would like to express my deep gratitude to my friends . Hesham Kamal Arafat, Taha El-Kzaz, Sayed Ibrahim, Ahmed Hassn and Ashraf Saber for helping me throughout this thesis.

Contents & List of Figures

CONTENTS

ABSTRACT I

INTRODUCTION 1

CHAPTER 1: AN OVERVIEW ON CONCURRENT PROCESSING 6

1.1- PROCESS, PROCESSOR AND RESOURCE 6

1.1.1- RELATION BETWEEN PROCESSES 7

1.1.2- TYPES OF SYSTEM RESOURCES 8

1.2- CONCURRENT PROCESSING 8

1.3- PARALLEL PROCESSING LEVELS 10

1.4- DIFFERENT TYPES OF PARALLELISM 11

1.4.1- Concurrency 11

1.4.2- Multiprocessing 11

1.4.3- Multiprogramming 12

1.4.4- Pipelining 12

1.4.5- Multipass algorithms, Interleaving, Co-routines, and Pseudo-parallelism 12

1.5- CONCURRENCY PROBLEMS 13

1.5.1- Mutual Exclusion 14

1.5.2- Synchronization 15

1.5.3- Deadlock 16

1.5.4- Switching Problem 17

1.6- DESCRIPTION OF PARALLEL PROCESSES 18

1.6.1- Parallel initiation and termination mechanisms 18

1.6.2- Synchronization mechanism 21

1.6.3- Protection mechanism 21

1.7- CORRECTNESS OF CONCURRENT (PARALLEL) PROGRAMS 22

CHAPTER 2: PARALLEL ARCHITECTURES 23

2.1- EFFICIENT ARCHITECTURES FOR PARALLELISM 23

2.1.1- Special - or General - Purpose System 23

2.1.2- Grain Size (Granularity) 23
2.1.3- Interconnection (Topology) 24
2.1.4- Tightness of Coupling 25
2.1.5- Control and Data Mechanisms 26
2.1.6- Task Allocation and Routing 26
2.1.7- Reconfiguration 27
2.1.8- Programming Languages 27
2.1.9- Nature of Technology 27
2.1.10- Performance Evaluation 27
2.2- CLASSIFICATION OF PARALLEL ARCHITECTURES 27
2.2.1- The Relation of The Memory to The Processors 27
2.2.2- Multiplicity Of Instruction-Data Streams 28
2.3- EVALUATION CRITERIA 31
2.4- PARALLELISM IN UNIPROCESSOR SYSTEMS 32
2.5- BASIC PARALLEL ARCHITECTURES 33
2.5.1- Multiprocessors 33
2.5.2- Vector Processors 33
2.5.3- Pipeline Processors 33
2.5.4- Array Processors 34
2.5.5- Systolic Processors 35
2.5.6- Wavefront Array Processors 37
2.5.7- Cube Architectures 37
2.5.8- Pyramid Architectures 38
2.5.9- Prism Architectures 39
2.5.10- The INMOS Transputer 39
2.6- PERFORMANCE OF PARALLEL COMPUTERS 41
CHAPTER 3: PARALLEL PROCESSING IN ARTIFICIAL NEURAL NETWORK 43
3.1- ARTIFICIAL NEURAL NETWORK 43
3.2- NEURAL NETWORK CONCEPTS 45
3.3- MODELLING THE SINGLE NEURON 49
3.4- LEARNING ALGORITHMS 51

3.4.1- The Perceptron Learning Algorithm 51
3.4.2- Widrow-Hoff (Delta) Rule 52
3.4.3- Backpropagation Algorithm 53
3.5- PARALLELISM IN NEURAL NETWORK 56
3.6- ALGEBRAIC PARTITION OF SUPERVISED LEARNING ALGORITHMS 58
3.6.1- Wavefront Algorithm 58
3.6.2- Fox-Like Algorithm 60
3.7- PARALLEL BACKPROPAGATION MACHINE 63
3.8- PARALLELISM IN BACKPROPAGATION 66
3.9- PARALLEL IMPLEMENTATIONS 69
CHAPTER 4: MODEL DESCRIPTION AND ITS IMPLEMENTATION 70
4.1- OVERVIEW ON HIGH LEVEL PETRI NET 70
4.2- MODELLING OF LEARNING PHASE OF TWO LAYER ANN BY USING HLPN 71
4.2.1- Problem Definition 71
4.2.2- Model Description 73
4.3- SOFTWARE IMPLEMENTATION 75
4.3.1- Learning And Testing 76
4.3.2- Delta Rule Model 92
4.3.3- Demo 93
4.3.4- Help 93
CONCLUSION 95
APPENDIX 96
REFERENCES 136

List of Figures

Figure (1.1) Interleaving and Overlapping techniques 9
Figure (1.2) Deadlock system 16
Figure (1.3) The precedence graph of Co-begin and Co-end 19
Figure (1.4) The precedence graph of Fork instruction 20
Figure (1.5) The precedence graph of Join instruction 20

Figure (2.1) Different topologies of processors 24
Figure (2.2) The difference between the tightly and loosely coupled systems 25
Figure (2.3) SISD computer 28
Figure (2.4) SIMD computer 29
Figure (2.5) MISD computer 30
Figure (2.6) MIMD computer 30
Figure (2.7) Block diagram of general pipelining 34
Figure (2.8) Torus of 3x3 processors and ring of five processors 35
Figure (2.9) Basic principle of a systolic system 36
Figure (2.10) Three dimensional binary cube 37
Figure (2.11) Pyramid network 39
Figure (2.12) Basic structure of transputer 40
Figure (2.13) The structure of some multi-transputer 40
Figure (2.14) Various estimates of the speedup of an processor system over a single processor 42

Figure (3.1a) Simple neuron 44
Figure (3.1b) Details of the basic model 44
Figure (3.2) The difference between the two-layer and three-layer neural networks 46
Figure (3.3) Usual activation functions 48
Figure (3.4a) Supervised learning 49
Figure (3.4b) Unsupervised learning 49
Figure (3.5) A multilayer network 54
Figure (3.6) Matrix vector product Y=W.X on a torus of wavefront algorithm 59
Figure (3.7) State of matrices used for the product Y=W.X before and after the first step of the Fox-Like algorithm on a 4x4 torus 61
Figure (3.8) General architecture of parallel network 63
Figure (3.9) Detailed architecture of each stage 64
Figure (3.10) Speedup in training parallelism 68
Figure (3.11) Speedup in spatial parallelism 68