Applications of Particle Swarm Optimization to Digital Communication

By

Muhammad Zubair

A dissertation submitted to I.I.U. in partial fulfillment of the requirements for the degree of

DOCTOR OF **PHILOSOPHY**

Department of Electronic Engineering

Faculty of Engineering and Technology

International Islamic University, Islamabad.

2008

بسم الله الرحمن الرحيم

DEDICATED TO

HOLY PROPHET (P. B. U. H.)

THE GREATEST SOCIAL REFORMER

&

MY PARENTS

LOVING SPOUSE AND KIDS

Certificate of Approval

Ir is certified that we have read the project report submitted by **Muhammad Zubair [09-FET/PhDEE/F-07]**. It is our judgment that this report is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for degree of Ph.D. Electronic Engineering (PhDEE).

Supervisor
Dr. Ijaz Mansoor Qureshi
Dean, FET, IIU Islamabad.

External Examiner
Dr. Muhammad Arif
Principal Engineer (Associate Professor)
Pakistan Institute of Engineering and Applied Sciences
Islamabad.

Internal Examiner
Dr. Tanweer Ahmed Cheema
Assistant Professor, IIU, Islamabad.

Declaration

I hereby declare that this research and simulation, neither as a whole nor as a part thereof, has been copied out from any source. It is further declared that I have developed **this** research, simulation and the accompanied report entirely on the basis of my personal effort made under the guidance of my supervisor and teachers.

If any **part** of this report to be copied or found to be reported, I shall standby the consequences. No portion of this work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Muhammad Zubair
[09-FET/PhDEE/F-07]

Abstract

The whole class of evolutionary computing algorithms is inspired by the process of evolution in nature. Compared to the traditional optimization algorithms. a few striking features of these algorithms include their ability to address non-differentiable cost functions, robustness to the dynamically changing environment, and implementation on parallel machines. However, it was not until one and half decade ago, when these algorithms attracted researchers and got acknowledgement in terms of theu application to the real world problems. The main reason behind this increased interest of the researchers owes to the ever increasing computing power. As a result evolutionary computing algorithms have been widely investigated and successfully applied for a number of problems belonging to diverse areas. In this dissertation the standard binary particle swarm optimization (PSO) and its soft version, namely soft PSO (SPSO) have been applied to four different problems of digital communication.

Due to the exponentially growing computational complexity with the number of users in optimum maximum likelihood detector (OMLD), suboptimum techniques have received significant attention. We have proposed the SPSO for the multiuser detection (MUD) in synchronous as well as asynchronous multicarrier code division multiple access (MC-CDMA) systems. The performance of SPSO based MUD has been investigated to be near optimum, while its computational complexity is far less than OMLD.

Particle swarm optimization (PSO) aided with radial basis functions (RBF) has been suggested to carry out multiuser detection (MUD) for synchronous direct sequence code division multiple access (DS-CDMA) systems. The MUD problem has been taken as a pattern classification problem and radial basis functions have been used due to their excellent performance for pattern classification.

The two variants of PSO have also been used in a joint manner for the task of the channel and data estimation based on the maximum likelihood principle. The PSO algorithm works at two different levels. At the upper level the continuous PSO estimates the channel, while at the lower level, the soft PSO detects the data. The simulation results have proved to be better than that of joint Genetic algorithm and Viterbi algorithm (GAVA) approach.

List of Publications and Submissions

1. **M. Zubair, M. A. S. Choudhry A. N.** Malik and I. M. Qureshi, "Particle Swarm optimization assisted Multiuser detection along with Radial Basis Function", *IEICE TRANSACTIONS on Communications Vol.E90-B No.7 pp.1861-1863 July 2007.*

2. M. A. S. Choudhry, M **Zubair, A.** Naveed, and I. M. Qureshi, "Near Optimum Detector for DSCDMA System using Particle Swarm Optimization," IEICE TRANSACTIONS on Communications Vol.E90-B,No.11,pp.3278-3282 Nov. 2007.

3. **M. Zubair,** M. **A. S.** Choudhry A. N. ***Malik*** and I. M. Qureshi, "Multiuser detection for Asynchronous Multicarrier CDMA using Panicle Swarm Optimization", *IEICE TRANSACTIONS an Communications,* Vol.E91-B, No.5, pp.1636-1639, May. 2008

4. **M. Zubair, M. A.** S. Choudhry A. N. ***Malik*** and I. M. Qureshi, "Particle Swarm with Soft Decision for Multiuser detection of Synchronous Multicarrier CDMA", *IEICE TRANSACTIONS on Communications,* Vol.E91-B, No.5, pp.1640-1643, May. 2008

5. **M. Zubair, M. A. S.** Choudhry A. N. Malik and I. M. Qureshi, "**Joint Channel** and **Data Estimation using Particle Swarm** Optimization", ", *IEICE Transactions an Communications,* Vol.E91-B, No.9, pp.3033-3036, Sep. 2008.

Acknowledgement

I humbly praise and thank ALMIGHTY ALLAH, The compassionate and the merciful, who gave health, thoughts, affectionate parents, talented teachers, helping friends and opportunity to contribute to the vast body of knowledge. Peace and prayers for His Prophet HAZRAT MUHAMMAD (S.A.W.) whose incomparable life is the glorious model for humanity.

I offer my sincerest thanks to my supervisor, Dr. **Ijaz Mansoor Qureshi,** for his many years of patience, guidance, illustrious advice, useful suggestions, kind and dynamic supervision throughout the research work. His personal interest, valuable suggestions, most co-operative affectionate behavior, constructive and thoughtful criticism resulted in the completion of this dissertation.

I am also thankful to all my fellows and friends whose good wishes and encouragement enabled me to complete this dissertation successfully. I offer my heartfelt gratefulness to Dr. Muhammad Aamer Saleem, Dr. A. N. Malik, Dr. T. A. Cheema and Dr. Abdul Jalil for their help and valuable suggestions during the course of research which also cannot be forgotten.

I also wish to express my feeling of gratitude to my father, mother, brothers, sisters, cousins and all the relatives, who prayed for my health and brilliant future. Mr. Ghias Malik deserves my thanks for his assistance. Last but not least, I offer my profound gratitude to my wife for being patient and co-operative throughout my research period.

(Muhammad Zubair)

List of Figures

2.1 Flowchart of basic PSO algorithm. 14

2.2 Various Neighborhood Topologies. 16

3.1 Transmitter for k^{th} user for MC-CDMA system. 42

3.2 Transmitter for k^{th} user for MC-DS-CDMA system. 45

3.3 Transmitter of MC-CDMA System. 46

3.4 Proposed Synchronous MC-CDMA Receiver. 47

3.5 BER Performance of HPSO for synchronous MC-CDMA system with 10 users 54

3.6 BER Performance of SPSO for synchronous MC-CDMA system with 10 users. 54

3.7 Performance Comparison of GA, SPSO, and HPSO for synchronous MC-CDMA system with 10 users.. 55

3.8 BER Performance of HPSO for synchronous MC-CDMA system with 15 users. 56

3.9 BER Performance of SPSO for synchronous MC-CDMA system with 15 uses. 56

3.10 BER Performance of HPSO for synchronous MC-CDMA system with 20 users 57

3.11 BER Performance of SPSO for synchronous MC-CDMA system with 20 users 57

3.12 Performance Comparison of GA, SPSO, and HPSO for synchronous MC-CDMA system with 20 users.. 58

3.13 BER Performance of HPSO for synchronous MC-CDMA system with 32 users. 59

3.14 BER Performance of SPSO for synchronous MC-CDMA system with 32 users. 59

3.15 Proposed Asynchronous MC-CDMA Receiver. 61

3.16 BER Performance of HPSO for asynchronous MC-CDMA system with 10 users. 69

3.17 BER Performance of SPSO for asynchronous MC-CDMA system with 10 users. 69

3.18 Performance Comparison of GA and HPSO for asynchronous MC-CDMA system with 10 users. 70

3.19 BER Performance of HPSO for asynchronous MC-CDMA system with 10 users 71

3.20 BER Performance of SPSO for asynchronous MC-CDMA system with 10 users 71

3.21 BER Performance of HPSO for asynchronous MC-CDMA system with 20 users 72

3.22 BER Performance of SPSO for asynchronous MC-CDMA system with 20 users 72

3.23 BER Performance of HPSO for asynchronous MC-CDMA system with 32 users. 74

3.24 BER Performance of SPSO for asynchronous MC-CDMA system with 32 users. 74

4.1 Radial basis function network.. 77

4.2 System Model of Transmitter. 79

4.3 RBF-PSO assisted receiver. 80

4.4 BER Performance of PSO-RBF assisted MUD for system with 10 users. . 83

4.5 Performance Comparison of PSO-RBF assisted MUD with GA assisted MUD system with 15 users. 84

4.6 BER Performance of SPSO-RBF assisted MUD for system with 15 users. 84

4.7 Performance Comparison of RBF assisted PSO with different suboptimal techniques for system with 20 users. 85

4.8 BER Performance of SPSO-RBF assisted MUD for system with 20 users. 86

4.9 Performance of HPSO-RBF assisted MUD for system with 32 users 86

4.10 Performance of SPSO-RBF assisted MUD for system with 32 users 87

5.1 Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 1 for 2 PAM.. 95

5.2 Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 2 for 2 PAM. 96

5.3 Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 3 for 2 PAM. 96

5.4 Performance of PSO in terms of MSE against the number of iterations for channel 1. 97

5.5 Performance of PSO in terms of MSE against the number of iterations for 98 channel 2. . . . .

5.6 Performance of PSO in terms of MSE against the number of iterations for channel 3. 98

5.7 Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 1 for 8 PAM. 99

5.8 Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 2 for 8 PAM. 100

5.9 Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 3 for 8 PAM. 100

List of Abbreviations

1G	First Generation
2G	Second Generation
ACO	Ant Colony Optimization
AMPS	Advanced Mobile Phone Systems
AWGN	Additive white Gaussian Noise
BER	Bit Error Rate
CDMA	Code Division Multiple Access
DS	Direct Sequence
EA	Evolutionary Algorithm
EP	Evolutionary Programming
ES	Evolutionary Strategies
FDMA	Frequency Division Multiple Access
FH	Frequency Hopping
GA	Genetic Algorithm
GD	Gradient Descent
GP	Genetic Programming
HPSO	Hard Particle Swarm Optimization
ISI	Inter-Symbol Interference
LMMSE	Linear Minimum Mean Square Error
MAI	Multiple Access Interference

MC-CDMA	Multi-Carrier Code Division Multiple Access
MLD	Maximum Likelihood Detector
MMSE	Minimum Mean Square Error
MOPSO	Multiple Objective Particle Swarm Optimization
MSE	Mean Square Error
MUD	Multiuser Detection
OFDM	Orthogonal Frequency Division Multiplexing
PIC	Parallel Interference Cancellation
PN	Pseudo-Noise
PPIC	Partial Parallel Interference Cancellation
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
SIC	Successive Interference Cancellation
SNR	Signal to Noise Ratio
SPSO	Soft Particle Swarm Optimization
SPSO1	Sofl Particle Swarm Optimization Version I
SPSO2	Sofl Particle Swarm Optimization Version 2
TDMA	Time Division Multiple Access
WCDMA	Wide-Band Code Division Multiple Access

Contents

Abstract vi

List of Publications and submissions vii

Acknowledgement viii

List of Figures ix

List of Abbreviations xiii

1. Introduction 1

1.1 Multiple Access Techniques 1

1.2 Evolutionary Algorithms 2

1.3 Contributions of Dissertation 3

1.4 Organization of Dissertation 4

2. Particle Swarm Optimization 6

2.1 Introduction 6

2.2 Stochastic Algorithms 7

2.3 Evolutionary Algorithms 8

2.3.1 Evolutionary Programming 9

2.3.2 Evolutionary Strategies 10

2.3.3 Genetic Programming 10

2.3.4 Genetic Algorithms 11

2.4 Particle Swarm Optimization 12

2.4.1 The Continuous PSO Algorithm 12

2.4.2 Topologies for PSO 15

2.4.2.1 Gbest Model 15

2.4.2.2 Lbest Model 15

2.4.2.3 Wheel Topology 15

2.4.2.4 Von Neumann Topology 17

2.4.3 Modifications to the PSO for Improved Convergence 17

2.4.3.1 Selection of Velocity 18

2.4.3.2 Inertia Weight 18

2.4.3.3 Constriction Factor 19

2.4.3.4 Acceleration Constants 20

2.4.3.5 Maintenance of Diversity 20

2.4.4 PSO Variants 21

2.4.4.1 Amalgam of Differential Evolution and PSO 21

2.4.4.2 Amalgam of Evolutionary Programming and PSO. 22

2.4.4.3 Amalgam of Evolutionary Strategies and PSO 22

2.4.4.4 Amalgam of GA and PSO 23

2.4.4.5 Amalgam of Genetic Programming and PSO 24

2.4.4.6 PSO for Multiobjective Optimization 25

2.4.4.7 PSO for Dynamic Environment 27

2.4.4.8 PSO for Constrained Optimization 29

2.4.4.9 PSO for Noisy functions 30

2.4.4.10 Discrete PSO 30

2.4.4.11 Gaussian PSO 31

2.4.4.12 Dissipative PSO 32

2.4.4.13 PSO with Passive Congregation........................ 32
2.4.4.14 Cooperative PSO.................................... 33
2.4.4.15 Concurrent PSO..................................... 33
2.4.4.16 Hierarchical PSO................................... 33
2.4.4.17 Angle Modulated PSO................................ 34
2.4.4.18 Fully Informed PSO................................. 34
2.4.4.19 Principle Component PSO............................ 34
2.4.4.20 Attractive Repulsive PSO........................... 34
2.4.4.21 Gregarious PSO..................................... 35
2.4.4.22 Soft PSO-I... 35
2.4.4.23 Soft PSO-II.. 36
2.4.5 Applications of PSO.................................... 36
2.4.5.1 Artificial Neural Networks........................... 36
2.4.5.2 Image Processing..................................... 37
2.4.5.3 Signal Processing.................................... 37
2.4.5.4 Antennas... 37
2.4.5.5 Power.. 37
2.4.5.6 Robotics... 38
3. PSO Assisted Multiuser Detection for MC-CDMA.................. 39
3.1 Introduction.. 39
3.2 Background.. 39
3.3 Problem formulation of MUD of Synchronous MC-CDMA system. 46
3.3.1 HPSO Assisted MUD for Synchronous MC-CDMA.............. 49

3.3.2 SPSO Assisted MUD for Synchronous MC-CDMA 52

3.3.3 Performance of PSO Assisted MUD for Synchronous MC-CDMA System. 53

3.4 Problem Formulation of MUD of Asynchronous MC-CDMA System 60

3.4.1 HPSO Assisted MUD for Asynchronous MC-CDMA 64

3.4.2 SPSO Assisted MUD for Asynchronous MC-CDMA 67

3.4.3 Performance of PSO Assisted MUD for Asynchronous MC-CDMA System. 68

4. Radial Basis Functions and PSO Assisted Multiuser Detection for DS-CDMA. 75

4.1 Introduction. 75

4.2 Background 75

4.3 Problem Formulation of RBF assisted MUD for Synchronous DS-CDMA System. 78

4.3.1 HPSO for RBF Assisted MUD 81

4.3.2 SPSO for RBF Assisted MUD 82

4.3.3 Simulation Results and Discussion 82

5. Joint Channel and Data Estimation using Particle Swarm Optimization 88

5.1 Introduction. 88

5.2 Background 88

5.3 Problem Formulation.. 90

5.4 PSO for Joint Channel and Data Estimation 92

5.5 Simulation Results and Discussion 94

6. Conclusion and Future Work 101

6.1 Conclusion 101

6.2 Future Work 102

References 104

CHAPTER 1

INTRODUCTION

1.1 Multiple Access Techniques

As we move from 3rd to 4th generation, the wireless communication faces one of the major challenges in terms of capacity, which is the rate of flow of information. More capacity is required to serve high data rates for multimedia applications and growing number of users. The problem can be dealt with by having more efficient multiple access schemes and better multiuser detection schemes.

Traditionally, frequency division multiple access (FDMA) and time division multiple access (TDMA) have been used for multiple access in wireless communication. TDMA is superior to FDMA providing better data rates than FDMA. Also the battery consumption at user end is minimized due to non-continuous operation in TDMA. Still more useful scheme is the CDMA. One of the major advantages of CDMA is the soft user capacity, that is, there is no sharp limit of maximum number of users. Another advantage is the assignment of unique code to each user, known as signature, due to which it utilizes the full bandwidth. Due to these advantages, the CDMA has the ability to cope with the upcoming challenges faced by modern wireless communication.

The major problem faced by CDMA is the multiple access interference (MAI). It arises due to non-orthognality of the signatures assigned to the users. The remedy is the multiuser detection. Multiuser detection has been an active area of research since its inception. Starting from the optimum detector proposed by Verdu [1], which is computationally prohibitive, a bulk of literature based on suboptimal techniques is available.

As the data rates are going up, the role of equalizers is also becoming more important. Knowledge of channel parameters makes the job of receiver easy. The systems using FDMA generally do not require equalizers, however, these are required for TDMA systems.

1.2 Evolutionary Algorithms

Initially the evolutionary techniques were not enthusiastically investigated and applied to various research fields owing to their computational cost. However, the situation changed in the last one and a half decade. Due to decreasing cost of computation, evolutionary techniques have undergone a revival. In the recent decade, not only the previously existing evolutionary algorithms have been applied to various research problems, but also some new techniques, like particle swarm optimization (PSO), ant colony optimization (ACO) etc have been proposed. This has opened a new era for the researchers. Now the emphasis is on two dimensions. One is to propose new techniques and the other is to make different evolutionary techniques co-work, as we can see in case of joint simulated annealing and genetic algorithm (SAGA) and joint GA and PSO etc.

All this activity has resulted in new evolutionary algorithms which are fast in convergence, robust to environment and applicable to a vast variety of problems. The evolutionary algorithms have been successfully applied to various research areas like image processing, neural networks, multiuser detection etc. Now it is becoming possible to use evolutionary algorithms to real time problems. One of the major advantages of evolutionary algorithms is that they work for non-differentiable functions. Also they are robust against local minima.

PSO is relatively new evolutionary computing technique proposed in 1995. Since then, it has been an area of great interest for the researchers. One can see the number of variants proposed so far.

1.3 Contributions of Dissertation

Following are the major contributions of this dissertation:

Multiuser detection for multicarrier CDMA system has been performed using PSO. Initially the solution to the multiuser problem was provided by Verdu [1], who proposed the optimal detector. However its computational requirement grows exponentially with the number of users in the system and is too large to be used practically. Later on, different suboptimal solutions were provided to reduce the computational cost. Among various suboptimal solutions, PSO belongs to the family of evolutionary computing. In this dissertation, binary PSO and its proposed variant have been used for multiuser detection (MUD). The problem has been addressed for both synchronous and asynchronous environment over Rayleigh flat fading channel. The proposed variant, named as soft PSO (SPSO), has given results even better than that of the binary PSO.

Radial basis functions (RBF) are supposed to be excellent for pattern classification. Another view of MUD could be a pattern classification problem. Hence the other contribution of dissertation is the PSO assisted Radial Basis functions which have been used for MUD of synchronous direct sequence CDMA (DS-CDMA) system over additive white Gaussian noise (AWGN) channel. Again binary and soft PSO have been employed and their performance comparison is provided in the simulations.

The channel estimation and hence equalization for data communication is another active area of research. Lots of work on this topic can be found in literature. Yet another contribution is the use of PSO for joint estimation of the channel parameters and data over AWGN channel. The novelty lies in the fact that both the soft and continuous PSO algorithms have been used together in an iterative manner. The continuous PSO determines the channel parameters while the soft PSO estimates the data.

1.4 Organization of Dissertation

The dissertation is organized as follows:

Chapter 2 is about particle swarm optimization (PSO). A brief overview of evolutionary algorithms is presented. A detailed coverage of various aspects of PSO is presented, which has to be used in the subsequent chapters. It includes the basic PSO algorithm, major variants of PSO proposed in literature, and application of PSO to various areas in research.

Chapter 3 discusses multiuser detection for both synchronous and asynchronous multicarrier DS-CDMA systems using PSO. Two variants, namely soft PSO and hard PSO have been applied in both cases. The channel for both the systems has been assumed to be Rayleigh flat fading. It is shown that the PSO algorithm due to its fast convergence outperforms the other evolutionary computing algorithm.

Chapter 4 presents multiuser detection of synchronous DS-CDMA system using PSO along with radial basis function, which belongs to the family of neural networks. RBF is an excellent tool for solving non-linear as well as classification problems. The motive behind using RBF is that it addresses the MUD problem as pattern classification.

Chapter 5 addresses joint channel and data estimation using PSO. The novelty of this chapter is that both the continuous and soft PSO have been used in an iterative manner. The continuous PSO has been used for estimating channel parameters while the soft PSO has been used for determining the data bits. Simulation results are provided at the end.

Chapter 6 is dedicated for conclusions and future directions in which the PSO work may be extended.

CHAPTER 2

PARTICLE SWARM OPTIMIZATION

2.1 Introduction

Optimization may be defined as finding a set of parameters that satisfies some cost function under some constraints. Almost every real world problem can be cast into an optimization problem. When modeled mathematically, some problems, by nature, turn into a linear optimization problem while others may be non-linear.

Traditionally there have been three different ways to solve an optimization problem. namely, deterministic methods, analytic methods and stochastic methods [2]. The deterministic algorithms are characterized by the fact that, given some initial condition, they always follow the same path to reach the same final solution. Some of the deterministic algorithms are linear programming, divide and conquer, dynamic programming, branch and bound, Newton-Raphson algorithm, steepest descent algorithm including all its variants, Fletcher-Reeves. Davidson-Fletcher-Powell algorithms etc. Most of the algorithms tend to get stuck in suboptimal solution. Moreover, an algorithm efficient in solving one optimization problem may not be able to perform well for another problem. Also, these algorithms are not suitable for discrete variables problems.

Classical analytical methods. like Lagrangian method. often fail as the dimension of the problem becomes large. Moreover, combinatorial problems, like quadratic assignment, timetabling or scheduling problems use discrete states and such problems have non-continuous objective functions that are not in the domain of analytical methods. Stochastic methods is the thud category and given below in an elaborate manner.

2.2 Stochastic Algorithms

Deterministic methods tend to fail when used on untractable NP hard problems. One of the best ways out is to go for stochastic algorithms [3]. These algorithms do not guarantee to reach the exact global optimal solution, but often they tend to reach closer. Stochastic algorithms are superior to other algorithms in a number of ways. They do not require the cost function to be continuous and differentiable. They are easy to implement and well suited for combinatorial problems.

Some well known stochastic optimization algorithms are Simulated Annealing, Tabu Search, random search, Hill Climbing algorithm and evolutionary algorithms. All these algorithms may be viewed as test generate and test algorithms. The main steps involved in all these stochastic algorithms are given as follows. The only difference lies in the way new perturbations are made and the way solution is found.

1) An initial solution is generated and is named as current solution.
2) The current solution is perturbed to give a new solution.
3) Fitness of the new solution is evaluated. If it has a higher fitness, it is taken as current solution otherwise the previous solution is maintained as current solution.
4) Stop the algorithm ifconvergence is achieved otherwise go to step 2.

These algorithms face the problem of slow convergence.

2.3 Evolutionary Algorithms

The term evolutionary computation encompasses a whole field of techniques. As the name implies, all the techniques are based on one common point i.e. evolution. . A population of candidate solutions is generated (usually at random values) and evolved to reach the best possible solution. Evolutionary computation may also be taken as generalized stochastic algorithm. The evolution process itself in a certain evolutionary technique may differ from the process ofanother evolutionary technique.

The first step in the implementation of any evolutionary algorithm is the initialization of the population of candidate solutions. There exist a number of ways to initialize the population. This is followed by evaluation of the whole population through some fitness function, which is specific to the problem under consideration. On the basis of fitness, some individuals are selected as parents for generating new candidate solutions, also known as offspring or children. The process of generating offspring is usually called crossover. in which parent solutions may recombine in a number of ways to possibly generate a number ofchildren. Mutation is another operator in addition to crossover. It is used to avoid stagnation. The nest step is to decide which individuals (includiny parents and children) will be moved to the nest generation. This nest generation is then subjected to the same procedure and the process ofevolution continues till convergence.

There are a number ofadvantages of evolutionary algorithms. For example, it is possible and relatively easy to integrate problem specific prior knowledge which results in a much more efficient exploration of the search space. Also the traditional optimization techniques may be easily combined with evolutionary algorithms to give a robust system. Evolutionary algorithms can easily handle dynamic changes in the problem while the traditional techniques need to be restarted. Unlike many other methods, evolutionary algorithms can be implemented on parallel hardware [4][5]. Last but not the least. evolutionary algorithms do not require the objective function to be differentiable.

The evolution ofevolutionary algorithms can be traced back to 1950's but the process of their evolution was too slow. One major reason behind this was the non-availability of high speed parallel computing facilities. With the advent of constantly growing and cheaper computing power, evolutionary algorithms have attracted the researchers. There are four major classical evolutionary techniques.

2.3.1 Evolutionary programming (EP)

Evolutionary programming was introduced by Fogel [6] in 1966. The main feature of EP is that it is normally used to optimize real valued continuous functions. Instead of having crossover operator for information sharing, EP relies on mutation. That is why, originally it did not have crossover operator. It used only the selection and mutation operators. A candidate solution in the population is composed of a pair of real valued vectors. One of these vectors represents the objective variables while the other contains the strategy parameters for mutation. EP typically uses stochastic tournament selection. As a result the best individual is always preserved. Mutation is adaptive and is applied using a uniform probability distribution. The mutation rate goes down as the algorithm approaches the optimum. In every iteration one parent generates one offspring. The nest generation is produced by taking some N best members of the population such that all had fitness above the median of the population. EP has been successfully used for optimization in power systems [7][8], and imaging [9][10].

2.3.2 Evolutionary Strategies (ES)

Evolutionary Strategies was developed by Rechenberg [11][12] and Schwefel [13]. ES was basically used to solve real valued problems. ES maintains two sets of real valued vectors. one of which contains the parameters to be optimized and the other strategy parameters, which controls the mutation of the objective parameters. ES uses both the mutation and recombination operators. For next generation, one of the two strategies, namely *comma* and ***plus,*** is followed. In comma strategy, only a subset of the offsprings is carried forward to become parents for the next generation, thus wasting some possibly good candidate solutions. This is, however supposed to provide diversity to the search space. In plus strategy both the parents and the offspring compete to go to the nest generation. Some other strategies were also introduced later on by various researchers. The mutation in ES is performed by addition of normal distributed random numbers to both the objective parameters and the strategy parameters. ES has been applied for parameter estimation [14], image processing, computer vision [15] and task scheduling [16].

2.3.3 Genetic programming

Genetic programming [17]-[19] is used to evolve computer programs by applying genetic operators on a population of programs. An individual in the population of solutions is a program. The programs are represented as parse trees. The variables and constants, known as terminals. act as leaves of the tree and the inner nodes act as functions to be performed on the leaves. An individual gene is formed by concatenating the leaves and inner nodes together. There are three different ways to generate a random initial population. In the first method known as *"Grow"*, an individual created, may be of any depth up to a specified maximum depth. As a result one may see an individual with only one node in the tree. In the second technique known as *"Full"*, an individual is guaranteed to be of a certain depth. The thud technique known as *"Ramped-half and half"* is a combination of the first two techniques. It is supposed to provide diversity more than the other two techniques. The fitness test of the population is problem specific. There are two ways to generate the offspring. In the first method known as Reproduction, an individual is simply copied to the next generation without performing any operation on it. Koza [20][21] has proposed a number of useful crossover techniques. The algorithm stops once the optimal solution has been achieved. The major disadvantage of Genetic Programming is that it demands for too heavy processing power.

2.3.4 Genetic Algorithms

Genetic Algorithms (GAs) was a brainchild of Bremermann [22] but popularized by John Holland [23][24]. GAs enjoy a lot of variety in terms of representation. generation of initial population, selection criterion, mutation and crossover methods. The parameters to be optimized, known as genes are concatenated to make a string which is called a chromosome. Depending upon the problem scenario the initial population is generated either at random or through some biased procedure. The rate of crossover is at least ***95%.*** There are a number of ways to apply crossover, one point crossover, multipoint crossover and uniform crossover. Rate of mutation is less than ***5%.*** The purpose of mutation is to avoid stagnation and premature convergence. In order to preserve best individuals and promote them to nest generations, the elitism operator is used. A few successful applications of GA's include image processing [25]-[27], electromagnetics [28] and signal processing [29].

2.4 Particle Swarm Optimization

The particle swarm optimization is a population based optimization technique. It was proposed by Kennedy and Eberhart in 1995 [30][31]. Some striking features of PSO are its simplicity in implementation and fast convergence. It has been successfully applied to various problems in the field of Artificial Intelligence for training neural networks, multiuser detection, etc.

2.4.1 The Continuous PSO Algorithm

As stated earlier, PSO is a population based optimization technique. The population is termed as swarm and an individual candidate solution in the swarm is referred to as a particle. There are two types of possible representations of a particle. It can be real valued or binary valued depending on the nature of the problem beingoptimized. It must however be remembered that the original PSO algorithm was defined for real valued problems. The swarm can be initialized in two different ways. The first approach is to initialize it at random values. The other approach is to go for biased initialization which depends on the problem at hand. Any i* particle in the swarm has the following attributes associated with it.

1) Its current position denoted by x_i
2) Its current velocity denoted by v_i
3) The best position it has visited so far denoted by p_i

The velocities of the particles in the swarm are also initialized at small random real values. As the algorithm starts all the fitness of all the particles is evaluated. As stated earlier the fitness function is dependent on the problem being solved. The particle having the highest fitness is selected as the global best particle. Particle velocity and the position of a particle are updated as follows,

$$v_{im}(n) = v_{im}(n-1) + rand1.\varphi_1.(p_{im} - x_{im}(n-1)) + rand2.\varphi_2.(p_{gm} - x_{im}(n-1)) \quad (2.4.1)$$

$$x_{im}(n) = x_{im}(n-1) + v_{im}(n) \quad (2.4.2)$$

where φ_1 and φ_2 are positive constants, usually both having a numerical value of 2.0, and $rand1$ and $rand2$ are (uniformly distributed between 0 and 1) random numbers.

The eq. (2.4.1) requires some explanation. The first term on the right hand side is the previous velocity. It is termed as "momentum", or "inertia". The second term represents the cognitive component. It represents the private thinking of a particle. It is also referred to as "remembrance", or "memory". The third term is the social component. It represents the collective thinking of the population. It is also called "cooperation' or "group knowledge". The movement of the particles through the search space needs to be controlled in order to have a balance between exploitation and exploration of the search space. Hence the original PSO algorithm includes a parameter , that limits the velocity. The choice of value for V_{max} is left onto the user.

Usually it is taken between 4 and 6. Too small value of V_{max}, results in more exploitation than exploration of the search space. Similarly too big value of V_{max}, results in more exploration than exploitation. The complete PSO algorithm is described in the form of flowchart in Fig. 2.1.

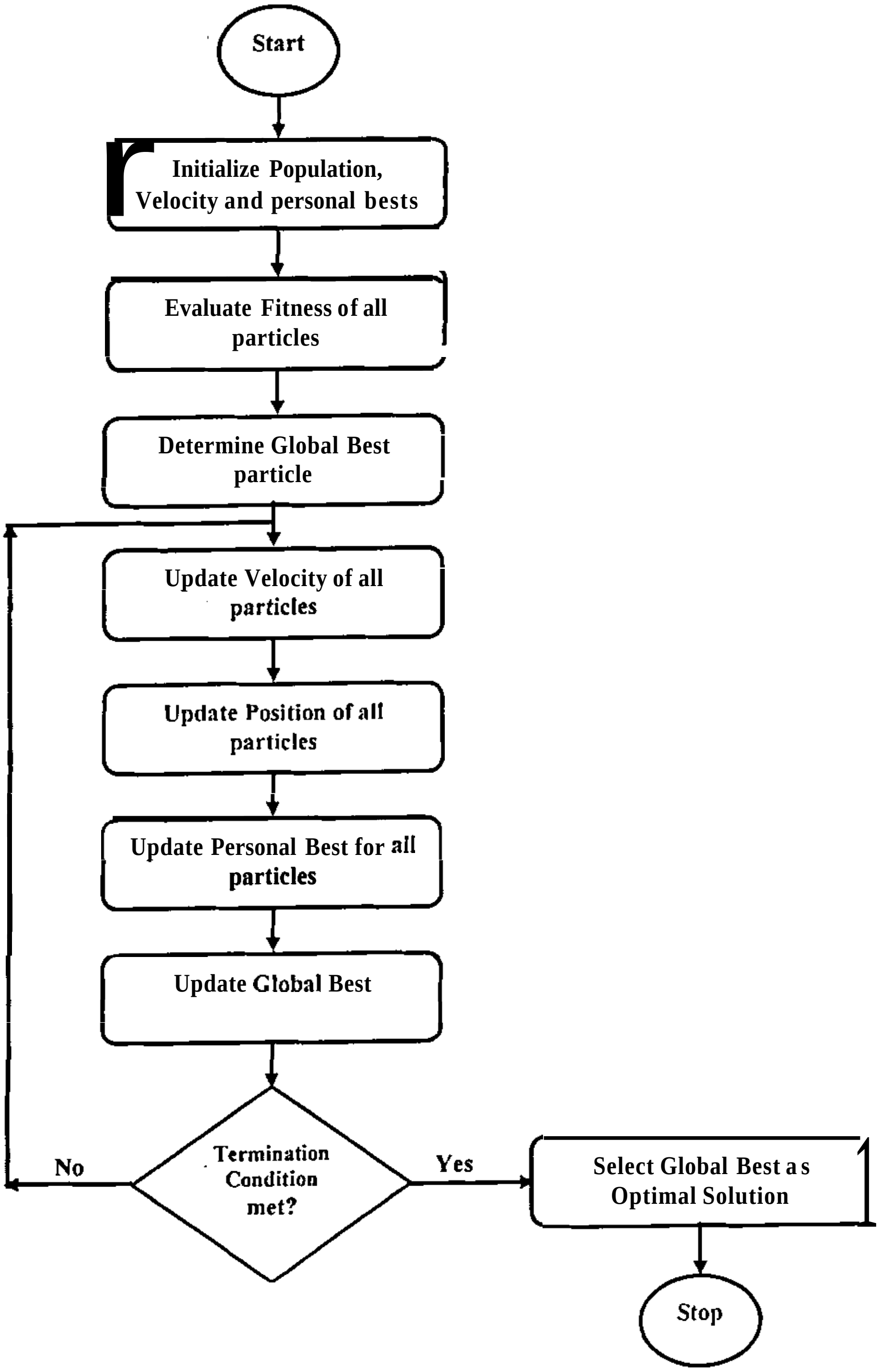


Fig. 2.1: Flowchart of basic PSO algorithm

2.4.2 Topologies for PSO

2.4.2.1 Gbest Model

This topology is also known as star topology as shown in Fig. 2.2a. In this topology every particle is connected to every other particle in the swarm. The particle having the highest fitness in the swarm is called the global best particle. This particle acts as a role model for all the particles in the swarm. All the particles get accelerated towards it. The main problem with this topology is that it is not very good at multimodal optimization. The *gbest* model has been shown to exhibit faster convergence at the cost of robustness [32].

2.4.2.2 Lbest Model

This topology is also referred to as circle/ring topology and is shown in Fig. 2. 2b. In this model the swarm is divided into a number of subswarms. Different neighborhoods or subswarms are independent but closely connected inwardly. Each subswarm has its own best particle known as local best particle or the neighborhood best particle. It may happen that one subswarm reaches it optimum while the other is still searching for it. Neighborhoods may be realized on indices as well as topological basis. It is important to understand that lbest is a generalization of the gbest model. If the neighborhood in lbest model is extended to the whole swarm it becomes the gbest model

2.4.2.3 Wheel Topology

Yet another topology *is* known as wheel topology. It is shown in Fig. 2. 2c. The main feature of this topology is that it effectively isolates individual particles from one another and all the information is passed to one central or focal individual. The focal individual

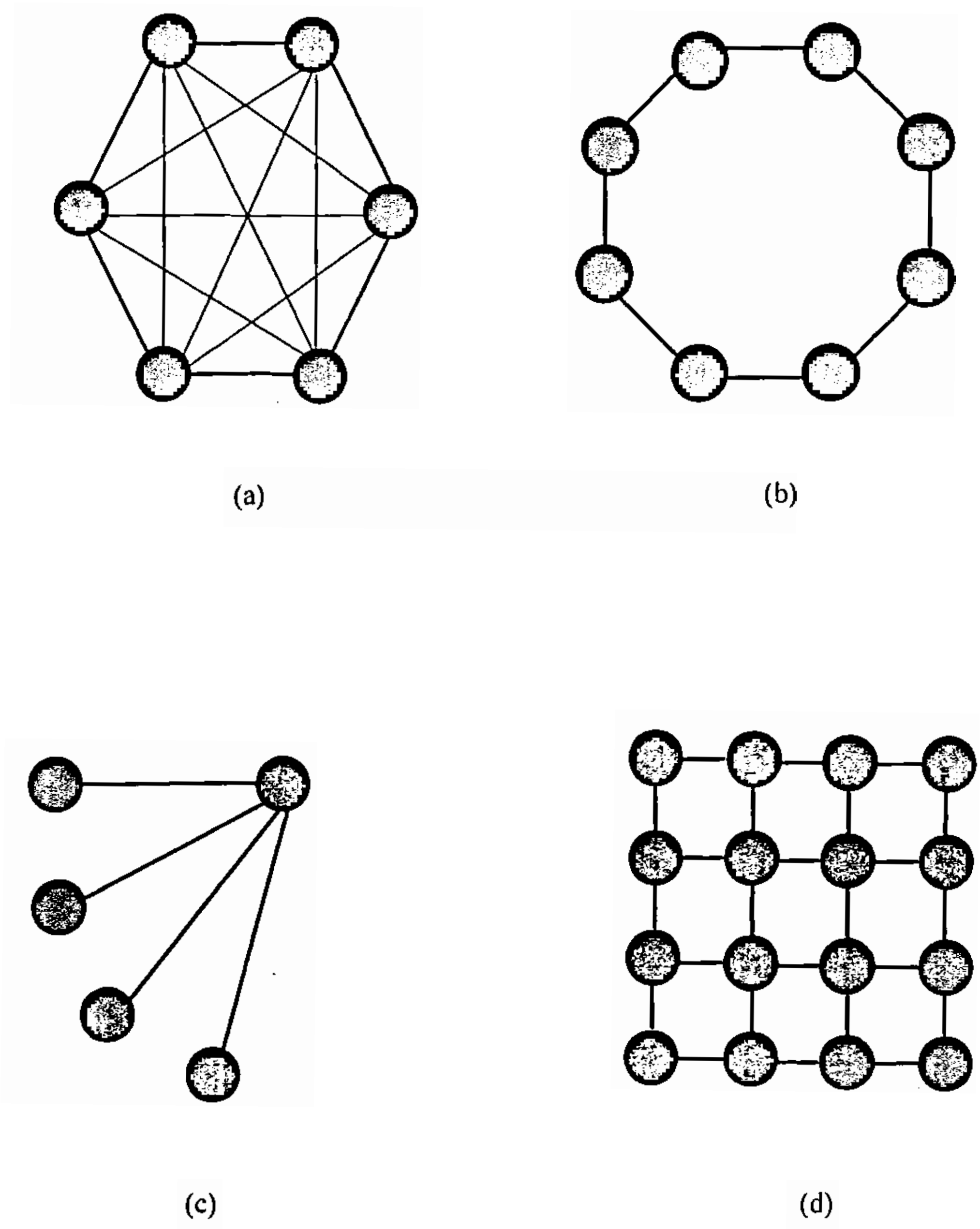


Fig.2.2: Various Neighborhood Topologies (a) Star, (b) Ring, (c) Wheel, (d) Von Neumann

modifies its trajectory in light of the information passed to it and if it results in bener performance. the information is communicated to the rest of the population.

2.4.2.4 Von Neumann topology

Another topology which is von Neumann topology, is shown in Fig. 2.2d. It is characterized by a grid of connected particles. Each particle is connected to its neighboring four particles (left, right, above and below).

The choice of a particular topology has thoughtful effect on the speed of convergence of the algorithm. Kennedy [33] suggested that gbest topology is faster but may be trapped in local minima while lbest is more likely to achieve global optima at the expense of slower convergence. The von Neumann topology is also slower in convergence because of its structure which takes more time to propagate the information to all the particles in the swarm. Kennedy and Mendes [34] have conducted experiments and found the von Neumann topology better than other topologies mentioned above. However, the selection of neighborhood topology depends on the type of problem at hand. It may happen that one particular topology working nicely may not work that good on some other type of problem.

2.4.3 Modifications to the PSO for improved convergence

Since its introduction, researchers from various areas have not only used PSO for problem solving but also contributed in terms of improving the performance of the PSO algorithm itself. The various steps of the algorithm have been modified with no change in the major structure of the algorithm. In this section we describe some of the highlighting modifications. In addition to the choice of proper neighborhood topology, the nest important issue is the choice of parameters. Once it is established that a certain algorithm achieves global optima, the next step to improve the algorithm is to find the ways to speed up the convergence process. This includes choosing and adjusting the velocity. setting the maximum velocity. and choosing appropriate values of acceleration constants. PSO has also undergone a number of investigations. In the following. we describe the major contributions by various researchers.

2.4.3.1 Selection of Velocity

The original PSO algorithm defines fixed limits on the maximum velocity of a particle referred to as V_{max}. However, later on Fan and Shi [35] proposed that a dynamically changing V_{max} can help to improve the performance. Another proposal made by Abido [36][37] suggested to adjust the velocity dynamically as follows,

$$V_{max} = (x_{max} - x_{min}) / N \qquad (2.4.3)$$

where x_{min} and x_{max} are the maximum and the minimum values attained by the particle so far and N specifies the number of intervals in k^{th} dimension.

2.4.3.2 Inertia Weight

One of the earliest and perhaps the most widely used work on the improvement of rate of convergence was contributed by Shi and Eberhart [38]. A weighting factor w as introduced in the velocity update equation is given as follows.

$$v_{im}(n) = wv_{im}(n-1) + rand1.\varphi_1.(p_{im} - x_{im}(n-1)) + rand2.\varphi_2.(p_{gm} - x_{im}(n-1))$$
(2.4.4)

which can be instantiated to the original PSO algorithm by letting $w = 1$. Shi and Eberhart have performed experiments by varying w in static as well as dynamic manner [39][40]. They have found the convergence to be faster when $w \in [0.8, 1.2]$.

It is evident from the above equation that the basic purpose of introducing this factor was to control the contribution of the previous velocity. This control over velocity helps to exploit the search space more than to explore. From this point of view the factor w is just like the temperature control in simulated annealing.

2.4.3.3 Constriction Factor

Clerc [41]-[42] proposed the idea of constriction factor and argued that there is no need of having V_{max} anymore. Clerc proposed several constriction models. Consider the following model,

$$v_{im}(n) = \chi(wv_{im}(n-1)) + \varphi_1(p_{im} - x_{im}(n-1)) + \varphi_2(p_{gm} - x_{gm}(n-1)) \quad (2.4.5)$$

where

$$\lambda = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|} \quad (2.4.6)$$

with the condition that $\varphi = \varphi_1 + \varphi_2, \quad \varphi > 4$

Russ [43] has compared the performance of the PSO algorithm with and without constriction factor. It was found that in general the inclusion of constriction factor resulted in an improved rate of convergence. For some test functions, constriction factor did not work well. However simultaneous application of constriction factor and velocity limit resulted in improved performance.

2.4.3.4 Acceleration Constants

Setting the acceleration constants controls the way the particles move towards the individual and global best particle. Ozcan *et al* [44] conducted investigation regarding the effect of these constants on convergence. They reported that as the coefficients are increased, the frequency of oscillation of the particle around the optimum increases while smaller values result in sinusoidal pattern. In most of the applications, both the constants are fixed at a numerical value of 2.0.

2.4.3.5 Maintenance of diversity

For any population based optimization technique, the importance of diversity goes without saying. The more the diversity. the more the chances to reach the optimal solution. However, there is always a chance of stagnation which results in premature convergence. In order to overcome such kind of situation, diversity schemes were introduced.

One of the earliest schemes was proposed by Sungathan [45]. He suggested using metric based neighborhood instead of indexed based neighborhood. The distance between any two particles is calculated at every iteration of the algorithm. For each particle the ratio $\|x_a - x_b\| / d_{max}$ is calculated in every iteration. Here d_{max} is the maximum distance between any two particles in the swarm and $\|x_a - x_b\|$ is the distance between particle a and b. Sungathan suggested using the selection threshold given below, for defining the neighborhoods.

$$selection\_threshold = \frac{3 \times k + 0.6 \times k_{max}}{k_{max}} \qquad (2.4.7)$$

Here k_{max} is the maximum number of iterations in the algorithm and k is the current iteration. At the start of algorithm small value of selection threshold defines small neighborhoods. As the algorithm progresses. the neighborhoods grow. and approach the size of the whole swarm by the end of the algorithm thus making it Gbest PSO. Sungathan also proposed to vary the stochastic acceleration parameters during the algorithm. The changes proposed by him proved to give better results.

2.4.4 PSO Variants

A class of PSO variants has been evolved by amalgamating it with other evolutionary techniques, described earlier like GA, ES, etc. This has, in fact, resulted in an increased diversity. These variants are described in the following.

2.4.4.1 Amalgam of Differential Evolution and PSO

The differential evolution (DE) is a population based stochastic search algorithm introduced by Storn and Prince [46]. DE is quite similar to GA. However in DE. mutation is performed first and then recombination followed by selection. The details of mutation are different from mutation in GA. Here in DE, weighted difference of two vectors is added to the vector being mutated. Another peculiar feature of DE is that every candidate solution undergoes mutation, and takes part in recombination and selection for the next generation. DE is defined only for real valued representation. The differential evolution PSO (DEPSO) was introduced by Zang and Xie [47] and further investigated by Talbi [48] and Moore [49].

2.4.4.2 Amalgam of Evolutionary Programming and PSO

The concept ofselection from the evolutionary programming (EP) was introduced in PSO by Angeline [50]. She suggested the following steps to be performed before going to update the velocity of the particles.

Step 1: A particle is picked up and compared to all other particles in the swarm. If the selected particle has better fitness than the other, it is given a point. This process is done for all the particles in the swarm.

Step 2: All the particles in the swarm are sorted in the order ofdecreasing points.

Step **3:** The upper half of the particles is duplicated and the lower half is removed from the swarm.

Experimental results by Angeline have shown that the inclusion of selection in the PSO has resulted in a better local search at the cost of degraded global search.

2.4.4.3 Amalgam of Evolution Strategies and PSO

One of the earliest attempts in this regard was to make PSO cowork with evolutionary strategies (ES) for application to power systems [51][52]. A particle is defined as a set of object parameters (the variables to be optimized) and strategic parameters (the weight, w_{ik}). The movement of particles was quite similar to classical PSO, except that the constants and random variables in (2.4.1) are replaced by weights, as given below.

$$w_{ik}^{*} = w_{ik} + \tau N(0,1) \tag{2.4.8}$$

where $N(0,1)$ is a random variable with Gaussian distribution having 0 mean and unit variance and τ is learning parameter which may be fixed or changed after mutation. It can be fised or treated as strategic parameter. The major steps of EPSO included replication, mutation, reproduction, evaluation and selection. All these steps form the major structure of evolutionary strategies algorithm. Replication is related to particle presence in the swarm. A particle is replicated r times (r is set at 1 in [52]). Each particle generates offsprings by following the particle movement which is quite similar to updating the particle position in classical PSO. Through stochastic tournament selection. the best particles are moved to the nest generation.

2.4.4.4 Amalgam of GA and PSO

The process of coupling between the evolutionary computation and PSO continued with the work by Lovbjerg *et al* [53], who applied recombination operator to the PSO algorithm and called it breeding. Remember that this modification was made to the original PSO algorithm, defined for real valued problems. The following modifications were proposed:

1) Calculate the particle velocities and positions according to the original PSO algorithm
2) Assign a parent probability to each particle.
3) Choose two particles and generate two offsprings through crossover. Replace the parents with the new offsprings.
4) The personal best positions of the particles used as parents, should be assigned as their current position.

Also he divided the swarm into sub-swarms for the sake of increased diversity. The changes suggested resulted in improved performance in case of multimodal functions.

However, for unimodal functions the convergence was slower than the original PSO algorithm.

Another hybrid of GA and PSO was proposed by El-Dib [54]. They applied reproduction to both the position and velocity vectors. Two randomly selected position particles generate two children *as* follows,

$$\begin{aligned} child_1(x) &= p.parent_1(x) + (1-p).parent_2(x) \\ child_2(x) &= p.parent_2(x) + (1-p).parent_2(x) \end{aligned} \qquad (2.4.9)$$

where p is a uniformly distributed random number between 0 and 1.

Two randomly selected velocity vectors generate two child velocity vectors as follows,

$$\begin{aligned} child_1(v) &= (parent_1(v) + parent_2(v)).\frac{|parent_1(v)|}{|parent_1(v) + parent_2(v)|} \\ child_2(v) &= (parent_1(v) + parent_2(v)).\frac{|parent_2(v)|}{|parent_1(v) + parent_2(v)|} \end{aligned} \qquad (2.4.10)$$

Robinson *et al* [55] investigated the application of alternating GA and PSO algorithms for optimizing a profiled corrugated horn antenna. They performed experiments and found that the best results were given by using PSO to GA algorithm i.e. the optimization starts with PSO and aflerwards GA takes over to reach the optimum solution. They announced that the PSO-GA approach performed better than the two schemes, GA-PSO and GA, both.

2.4.4.5 Amalgam of Genetic Programming and PSO

More recently Poli et *al* [56][57] proposed hybridization of genetic programming with **PSO**. They have focused on the possibility of evolving optimal force generating equations to control the particles in a PSO using genetic programming. Their focus was on the development of PSO through the use of GP so that instead of one problem, a whole class of problems could be addressed. In other words, the main thrust of GP is on finding the appropriate function, instead of the standard function. that drives the particles in the swarm to their local and global best positions. They used two benchmark functions. city-block sphere problem and Rastrigin's problem and came up with seven new PSO algorithms.

2.4.4.6 PSO for Multiobjective function optimization

There is a great deal of real world problems, which are of multiobjective nature. For example in the design of automobile. potential objectives include cost, fuel consumption. speed, capacity and comfort etc. In these problems it is difficult to find the best solution since the objectives to be optimized are conflicting. In addition to other techniques from operation research used for multiobjective optimization, evolutionary techniques have also gained considerable attention by the researchers. Schaffner [58][59] is considered to be among the pioneers to investigate the use of evolutionary computing for optimizing multiobjective functions. An excellent overview of the topic can be seen in [60] and [61]. A number of multiobjective PSO (MOPSO) algorithms have been proposed recently. There are two major issues in developing MOPSO algorithm [62][63]. One is to decide how to evolve the social and cognitive parameters that can lead other particles towards the optimum solution. The other is to maintain diversity. It is worth mentioning that the process selection of cognitive parameters in MOPSO is the same as in traditional uni-objective PSO algorithm. There are two steps in the selection of social factor. First is to setup a pool of potential particles. Second step is to actually choose the social leader. For this purpose there exist two classes of algorithms. Algorithms belonging to one class rely on random approach. The general strategy followed by the algorithms of this class is to assign weights to the particles. Ray and Liew [64] have suggested to first make a group of leader particles having higher fatnesses. All the particles then move towards a randomly selected leader. keeping in view the fact that the leader having the least number of followers has the highest chance of selection. Coello *et al* [65] have used two step procedure. First they store non-dominated solutions in a secondary population. The primary population then uses a randomly selected neighborhood best particle from the secondary population for updating their velocities. Later on they incorporated Pareto dominance and special mutation operator for enhancing the exploratory capabilities of the PSO algorithm [66].

The algorithms in the second class use some quantitative method instead of random approach to select the social leader. Mostaghim *et al* [67] proposed sigma method. He suggested using sigma values for each particle in the primary as well as in the candidate pool for the selection of social leader. The leader having the minimum sigma distance from the particle will be chosen as the leader. The use of sigma values results in an increased pressure on selection. Hu *et al* [68] introduced dynamic neighborhood strategy for selecting the social leader. The objectives are first divided into two groups say G1 and G2. G1 is termed as neighborhood objective and G2 is referred to as optimization objective. Each particle chooses its neighborhood by finding the distance between itself and all other particles. Only one objective is optimized at a time like lexicographic ordering. After the completion of neighborhood finding process local and then the global best particle is chosen.

Vector Evaluated PSO (VEPSO) is based on the concept of vector evaluated genetic algorithm (VEGA). Parsopoulos *et al* [69] proposed the vector evaluated PSO. The idea behind was to have two or more swarms. Each swarm goes for one objective function and the swarms share information.

More recently, Liang *et al* [70] have proposed another variant, named comprehensive learning PSO (CLPSO). The novelty of this new variant lies in the fact that any particle in the swarm can use the history of any other particle to improve itself. This ensures more diversity and hence helps to avoid premature convergence. Although CLPSO has shown good results for multimodal problems. it cannot perform equally well on unimodal functions.

2.4.4.7 PSO for Dynamic environment

A dynamic system is characterized by a fitness function that changes almost continuously over time, thus rendering the previously found optimal solution almost obsolete. A number of real world systems have dynamic nature. Scheduling systems are the best example of dynamic systems. where most of the time is spent in rescheduling. Parsopoulos *et al* [71] proved that the PSO can track slow moving optima without any changes at all. Eberhart [72] suggested slight change to the original PSO algorithm by randomizing the inertia weight between **0.5** and 1.0. The reason was to make the contribution of the social and cognitive factors flexible so that the exploration and exploitation could be adjusted according to changing environment. Afterwards, they suggested [73] specialized adaptive PSO for dynamic environment. A two step procedure was suggested which included detection and then responding to the changing environment. The change is detected when a different function value is observed upon re-evaluation. The response suggested, was to randomize the entire swarm. Earlier. Carlisle *et al* [74][75]suggested two responses upon the detection of change. First was to set the current position of the particle to the previous best and the second was to go for the winner of the current and previous best.

Blackwell and Bently [76][77] introduced the notion of charged particles to update the particle position. The special feature of this scheme was that it does not require any detection of changing environment. The particles repel each other and make a kind of orbit around a converging nucleus of neutral particles. It helps to stop the swarm from converging to a single point. The particles remain spread. thus making it easier to respond to change and readjust the optima. Das *et al* [78][79] have proposed small population PSO (SPPSO). The main feature of this variant is that the number of particles in the swarm is quite small (five or even less). The population is regenerated after every N iterations. All the particles except g_{best} and the p_{best} are replaced. SSPSO has performed well in dynamic environment.

Another challenging class of problems is the dynamic multimodal problems. In these environments, multi-swarm approach has proved to be fruitful.

Parrot *et al* [80] have adjusted the number and size of swarms dynamically. The swarm is divided into species subpopulations. based on their similarity. Each specie has a dominant particle, called seed. The species seeds are updated every iteration and serve as neighborhood best for the respective specie group. It is worthwhile to note that species are formed adaptively at each iteration, keeping in view the feedback from the multimodal fitness function. The Species PSO has proved to be effective under both the static and dynamic multiobjective environments.

2.4.4.8 PSO for Constrained Optimization

Most of the engineering problems are constrained problems. The basic PSO algorithm does not include any mechanism for handling the constrained optimization problems. With the passage of time, however, researchers have contributed to this end as well. There are two approaches to deal with constrained optimization using PSO. One way to optimize constrained problems using PSO is to incorporate constraints in the fitness function using penalty functions. Venter *et al* [81] have proposed to penalize the particles with violated constraints. The motive behind this was to reset the velocity of the violating particles.

The second option is to deal with fitness and constraints separately. This approach is quite helpful in the sense that there is no major change in the algorithm and also no limit on the form or number ofconstraints. After updating the position each particle is checked to see whether or not it is in feasible region [82]. If it is, then either the particle is reset to its p_{best} [83] or its previous position or it is randomized [84].

Parsopoulos *et al* [85] converted the constrained problem into non-constrained problem by incorporating a penalty function which is non-stationary. The application of PSO to the reformulated problem resulted in promising results. Ray et *al* [86] suggested to use information sharing at multiple levels so that a particle not performing better may get its direction corrected with the help of its nearest neighbor that is in better performer list (BPL). In order to deal with constraints. there is a constraint matrix which is used in conjunction with Pareto ranking scheme to create a BPL. The results were found to be quite encouraging. Other PSO schemes for handling constrained optimization may be found in [87]-[89].

2.4.4.9 PSO for Noisy function

Unlike dynamic problems, in noisy problems the fitness function does not change, however, its evaluation is noisy. This means that if the same position is visited again. the fitness value may be different. Parsopoulos and Vrahatis [90] investigated the behavior of PSO under Gaussian random noise. They added noise to the fitness function and performed random rotation to the search space. The results were quite encouraging in the sense that the performance of PSO was not affected by the presence of noise. Pugh [91] compared the noise resistant PSO with the original PSO and found that the noise resistant PSO keeps itself improving while the standard PSO stops evolutionary process after a few iterations. Also the noise resistant PSO takes less number of iterations than the standard PSO to reach the convergence.

2.4.4.10 Discrete PSO

The original PSO algorithm was developed for real number space. Later on discrete versions were also introduced.

Binary PSO, like real number PSO was proposed by Kennedy and Eberhart [92]. The purpose was to extend the standard PSO algorithm to work for binary number space since a number of problems are defined over binary number space. Thus the particle position is in the form of binary digits, 0 and 1. The velocity of the particle is, however, real-valued and limited to the interval [0, 1], thus treating it as probability. The velocity is updated as follows,

$$v_{im}(n) = v_{im}(n-1) + \varphi_1 .(p_{im} - b_{im}(n-1)) + \varphi_2 .(p_{gm} - b_{im}(n-1))$$

where φ_1 and φ_2 are the stochastic acceleration constants. Thresholding of velocity is accomplished by employing sigmoid function, defined as follows,

$$S(v_{im}) = \frac{1}{(1 + \exp(-v_{im}))}$$

In order to avoid the threshold being too close to 0 or 1, velocity is typically set to a maximum value V_{max}. Numerical value of V_{max} is 'usually set between 4 and 6.

Position of a particle is updated using the following

$$\begin{aligned} &if(rand(\ ) < S(v_{im})), \quad then \quad b_{im} = 1 \\ &else \quad b_{im} = 0 \end{aligned}$$

A number of problems require solution in the form of integers instead of binary numbers. For such problems integer PSO was proposed by Parsopoulos et *al* [93]. The equations developed for real number space are used for integer PSO. Once the position is determined it is rounded to the nearest integer value. Integer PSO has shown better performance than other methods like Brach and Bound method [94].

2.4.4.11 Gaussian PSO

The Gaussian PSO was proposed by Krohling [95]. While applying the classical PSO algorithm, one has to specify the constants regarding inertia weight, V_{max} etc. Gaussian PSO was proposed to avoid these initializations so that the acceleration constants are automatically initialized using Gaussian distribution. Inertia is set to zero so that there is no limit on maximum velocity. The Gaussian PSO was tested on well known benchmark functions and performed better than the classical PSO algorithm.

2.4.4.12 Dissipative PSO

Inspired by the concept of self-organizing dissipative systems. this version of PSO was proposed by Xie *et al* [96]. They have introduced the concept ofentropy, thus creating a dissipative structure which is far from equilibrium. The negative entropy is introduced in the system by incorporating an additional chaos in the velocity of particles as given below:

$$if(rand() < c_v), \quad then \quad v_{id} = rand().V_{max} \qquad (2.4.11)$$

Where c_v and $rand$ are both random numbers between 0 and 1. The non-linear interactions among the particles help establish equilibrium. The DPSO was tested on multimodal benchmark functions and gave good results.

2.4.4.13 PSO with Passive Congregation (PSOPC)

PSO with Passive Congregation (PSOPC) was proposed by He *et al* [97]. The focus is on avoiding local minima and at the same time enhance the convergence speed. This is achieved by incorporating congregation coefficient into the velocity update formula,

$$v_i(t) = \varphi_w.v_i(t-1) + \varphi_1.r_1.(p_i - x_i(t-1)) + \varphi_2.r_2.(p_g - x_i(t-1)) + \varphi_3.r_3.(X - x_i(t-1)) \qquad (2.4.12)$$

Here φ_3 is the passive congregation coefficient and r_1, r_2 and r_3 are random numbers between 0 and 1. X is any particle selected randomly from the swarm.

2.4.4.14 Cooperative **PSO**

Van den Bergh *et al* [98] introduced cooperative PSO. It employs multiple swarms to implement the concept of cooperative PSO. CPSO was inspired by the idea given by Potter [99] who suggested to split the search space into multiple smaller search spaces for GA's thus originating the cooperative co-evolutionary genetic algorithm (CCGA). Van *et al* introduced two versions of CPSO. One was named as CPSO-H_K and the other CPSO-S_K. The CPSO-S_K is exactly like CCGA. The strategy is to divide a swarm having n-dimensional vectors into n swarms consisting of k dimensional vectors. Thus each swarm optimizes one component of the solution vector. The number k is termed as split factor. A particle in a swarm is evaluated through credit assignment strategy.

CPSO-H_K is a combination of PSO and CPSO-S_K such that one iteration of CPSO-S_K is followed by one iteration of standard PSO algorithm. The cooperative approach to PSO has shown better than standard PSO when the dimensionality of the problem is increased.

2.4.4.15 Concurrent **PSO**

Concurrent PSO is also based on cooperation and has been proposed by Baskar and Sunganthan [100]. In this scheme the search space is partitioned into two sub-swarms implicitly.

2.4.4.16 **Hierarchical PSO**

Hierarchical PSO El-Abd *et al* [101] combines both implicit and explicit splitting of the search space using CPSO and CONPSO and has outperformed both schemes.

2.4.4.17 Angle Modulated PSO

It is well known that as the problem dimension increases, it puts more pressure on the performance of the optimization algorithms and as a result the algorithm converges to suboptimum solution. With this fact in mind, Pampara *et al* [102] proposed the Angle modulated PSO which reduces a high dimension binary valued optimization problem to four dimensional continuous valued problems. The AMPSO algorithm has been tested and found to give better results as compared to binary valued PSO.

2.4.4.18 Fully Informed PSO

Mendes and Kennedy [103] proposed fully informed PSO. The motive behind was that the PSO algorithm can perform better if a particle is made to share information with all other particles in the population instead of sharing with its neighbors only.

2.4.4.19 Principle Component PSO

Voss [104] proposed principle component PSO. The motive behind was to improve convergence of PSO on high dimensional problems. PCPSO uses covariance matrix and gives more importance to the current directions the particles are flying in, instead of the previous directions. PCPSO has been tested to give superior performance on 30 dimension standard functions, 100 dimension sphere, Rosenbrock and Voss functions.

2.4.4.20 Attractive Repulsive PSO

Attractive-Repulsive PSO was proposed by Riget and Vesterstrom [105] to overcome the premature convergence problem in PSO. The proposed PSO uses one of the two phases while updating the velocity; attraction and repulsion. In attraction phase it acts exactly like the original PSO. In repulsion phase. the velocity update is done by performing a subtraction (instead of addition) in the velocity update formula. This is described as the particles are attracted towards one another during attraction phase while pull one another during repulsion phase. The authors have claimed that the proposed version of PSO performs quite well on multimodal functions.

2.4.4.21 **Gregarious** PSO

Pasupuleti and Battiti [106] proposed gregarious PSO for avoiding premature convergence. They suggested re-initializing the particle when it is stuck at some local optima. Also they proposed to dynamically change step size keeping in view the feedback from previous iterations. The GPSO has been shown to outperform the basic PSO on a number of benchmark functions.

2.4.4.22 **Soft** PSO-I (SPSO-1)

The special feature of this type of PSO is that it works with the real numbers and the final result is taken in binary form. This type of PSO has been successfully used by Choudhry [107] and Zubair[108] for multiuser detection problems. The swarm is initialized at small real random numbers uniformly in [0, 1]. Particle position is updated through soft decision as follows.

$$\begin{aligned}&if(rand(\ ) > S(v_{im})),\quad then\quad b_{im} = -1 + S(v_{im})\\&else\quad b_{im} = S(v_{im})\end{aligned}$$

where

$$S(v_{im}) = (1/1 + \exp(-\gamma v_{im}))$$

It must be kept in mind that-, is one of the important parameters of SPSO. In binary PSO $\gamma = 1$. If $\gamma \to \infty$ we have pure step function. Thus with $\gamma = 0.25$, the variation of

$S(v_{im})$ is very gradual around $v_{im} = 0$. But as γ is increased, it becomes more and more abrupt tending towards more hardness. It has been observed after test and trial. that gradually increasing γ from 0.25. results in better performance. Upon the completion of the iterations, hard decision is carried out on the global best particle to get the final binary valued optimized solution.

2.4.4.23 Soft PSO (SPSO)-2

Choudhry [109] has also used another version of sofl PSO, called **SPSO-2.** This version has proved to be even better in terms of performance. The swarm is initialized at soft values. The particle position is updated as follows,

$$b_{im}(n) = b_{im}(n-1) + \tanh(\gamma v_{im}) \quad (2.4.13)$$

where $0 < \gamma < 1$. Like SPSO-I, the final result is obtained by taking a hard decision on the global best particle.

2.4.5 Applications of PSO

Despite the fact that PSO is comparatively new addition to the optimization techniques, it has shown its usefulness in a number of areas.

2.4.5.1 Artificial Neural Nehvorks

The use of PSO for finding the optimum weights for ANN. has been investigated by a number of researchers. The major focus of the researchers was on synaptic weights and the network topology. This has been done for supervised as well as unsupervised learning. A number of papers have reported the comparison and supremacy of the PSO over traditional back propagation algorithm [110]-[111] for finding the weights of neural networks. The obvious reason behind this is that PSO works on non-differentiable functions. One of the earliest uses of PSO was to train a Recurrent Neural Model [112]-[113]. Later on PSO was used for training product Units in Feedforward Neural Networks [114]-[116], neural network control for nonlinear processes [117], design of radial basis function [118], for training of support vector machines [119], fault diagnosis [120] etc.

2.4.5.2 Image Processing

PSO has been successfully applied to various disciplines of image processing. These include, image classification [121][122], image clustering [123][124], image segmentation [125], image noise cancellation [126], pattern matching [127], image restoration [128][129], texture synthesis [130], and various others [131]-[133].

2.4.5.3 Signal Processing

PSO has been used for solving problems relating to various sub-fields of signal processing like signal detection [134], source separation [135], filter design [136]-[139], speech coding [140], estimation [141]-[143] and multiuser detection [144]-[150].

2.4.5.4 Antennas

There is a considerable research interest in the application of PSO for antenna design [151]-[156]. PSO has been used for antenna arrays [157]-[167], for yagi array antenna [168], monopole antenna [169], and horn antenna [170].

2.4.5.5 Power

PSO has been successfully applied to solve problems pertaining to various areas in Power systems. One of such areas is Reactive Power and Voltage Control. Here the problem is to keep the voltage limited to a specified range. There exist a number of traditional methods to achieve it in static sense. A dynamic approach for achieving this is called volt/var control (VVC). Traditional optimization techniques have been used for VVC problem, but as the system grows, the dimensionality of the problem increases and the traditional techniques show signs of strain. A number of researchers have applied PSO for solving VVC problem and the results were found to be better than other contemporary solutions [171]-[175].

Another challenging problem in power systems is known as Economic Dispatch (ED) problem. In this problem it is desired to hit upon a set of optimal parameters for generational units such that their operational cost is minimized while meeting the other constraints. Researchers have applied evolutionary programming, Tabu search, artificial neural networks and GA. More recently a great interest has been shown by the researchers in applying PSO for ED problem [176]-[184]. Another area of interest is feeder reconfiguration which is a technique used for enhancing the quality as well as price of the service while meeting the constraints of reliability. PSO has been applied for feeder reconfiguration [185]-[187] and various other applications in power systems [187]-[194].

Control is another area where PSO has proved its strength. The specific applications include the design of PI and PID controllers [195]-[197] and AGC tuning [198].

2.4.5.6 Robotics

One of the major uses of PSO in robotics is the design of fuzzy neural network [199]-[201]. Also some researchers have used PSO for path planning [202]-[203] and other purposes [204]-[208].

CHAPTER 3

PSO ASSISTED MULTIUSER DETECTION FOR MC-CDMA

3.1 Introduction

In this chapter multiuser detection (MUD) of multicarrier CDMA (MC-CDMA) systems. in both synchronous and asynchronous environments, is discussed. The first section gives a brief history and major contributions by various researchers for MUD in MC-CDMA systems. The second section addresses the MUD of multicarrier system in synchronous environment using particle swarm optimization (PSO) for slow fading Rayleigh channel. In the third section MUD of MC-CDMA system for asynchronous system using PSO has been discussed. It has been shown that the PSO performs better than the Genetic algorithm (GA) in terms of bit error rate (BER).

3.2 Background

The most important part of the fourth generation (4G) systems is the wireless Internet and high speed multimedia services. This demands a high data rate. which is restricted by time varying dispersive fading channels and intersymbol interference (ISI). Fading is caused by multipath mobile channels. Different replica of the same signal reach the receiver after reflecting from buildings, trees, hills, and other large still structures or moving objects. These different copies of the signal add up at the receiver resulting in a signal that has replicas with different delays and different attenuations. This phenomenon results in frequency selective fading. Special' attention has to be given for receiving systems in these cases. The conventional systems fail to meet such a demanding situation. One of the ways to fight the multipath fading and ISI is to employ equalizer at the receiver.

With the increasing data rates, the equalizers become more and more complicated. Another solution is multicarrier transmission i.e. to use a number of carriers instead of a single one. The idea of multicarrier modulation was first used in Collins Kineplex system [209]. Later on Chang [210][211] proposed OFDM in 1966 and got it patented in 1970. Saltzberg [?l?] analyzed the performance of the idea presented by Chang for dispersive channel and emphasized on reducing the cross talk between adjacent channels rather than perfecting individual channels.

OFDM is a special case of multicarrier modulation. The idea behind OFDM is to split the available channel bandwidth into a number of orthogonal frequency flat channels. As a result of this the data rate on each subchannel will be decreased by expanding the symbol duration. The net effect will be very small or no equalization at the receiver. Hence the working principle of the OFDM is that a high bit rate input is split into parallel lower bit rates and transmitted across the channel. This results in an increase in the symbol duration and hence ISI is reduced considerably.

The difference between tnulticarrier tnodulation (MCM) and OFDM is that in OFDM the carriers are orthogonal to each other which may not be true in MCM. Also OFDM is similar to frequency division multiple access (FDMA) in which bandwidth is divided into a number of channels. However, the difference lies in the carrier spacing. Two adjacent channels in OFDM which are spaced I kHz apart will be 30 kHz apart in

FDMA. Initially the idea of OFDM could not get appropriate appreciation and as a result, there was no significant research in this direction. The reason for this lack of interest was the heavy computational complexity. The major milestones in the history of OFDM occurred in 1971. when discrete Fourier transform (DFT) was proposed for modulation and demodulation purposes [213]. This eliminated the use of tillers and oscillators at the receiver. The main contributions to OFDM research from 1980 to 1990 were focused on high speed OFDM systems [214]-[221]. In the present decade the OFDM is being used for high data rate applications in wired, as well as, wireless communication standards. An example is the asymmetric digital subscriber lines (ADSL). wireless local area networks (WLAN). such as 802.11[222], terrestrial digital vidw broadcasting (DVB-T) [223].

OFDM requires an extensive level of synchronization, both in time and frequency. However, by combining OFDM with CDMA the symbol rate in each subcarrier can be decreased so that longer symbol duration makes it easier to quasi-synchronize the transmissions.

Different combinations of MCM and CDMA have been proposed. In the first technique, known as multicarrier CDMA (MC-CDMA), which was introduced in 1993 [224]-[225], the *data* is spread in frequency domain. Figure 3.1 shows transmitter diagram for k^{th} user in a typical MC-CDMA system. Each user is assigned a spreading sequence of length N. After serial to parallel conversion of data, each user bit is multiplied with one chip of the spreading sequence and sent on one of the M orthogonal carrier.

Schnell [226] used maximum likelihood detector for MC-CDMA. which may be discarded due to its exponentially growing computational complexity with the number of users. The performance of iterative detection was analyzed in [227]-[228]. Park *et al* [229] discussed the parallel interference cancellation for multipath fading channels.

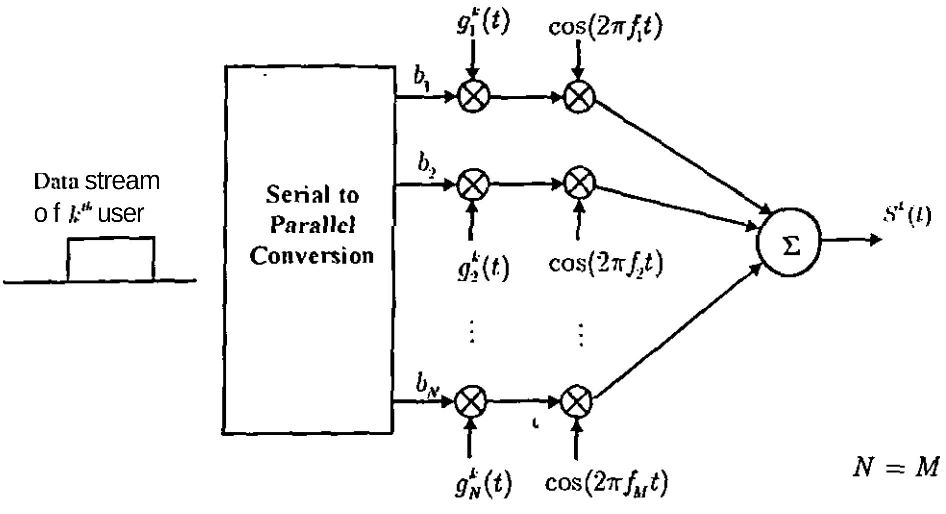


Fig. 3.1: Transmitter for k^{th} user for MC-CDMA system

Hara and Prasad [230] investigated the performance of MC-CDMA in frequency selective Rayleigh fading channels. Tat [231] proposed optimized transmitter and receiver for MC-CDMA system. The receiver optimization was performed in a decentralized manner while transmitter optimization was performed through either centralized or decentralized control of the powers of different carriers. Based on interference cancellation technique, a novel receiver was proposed by McCormick [232]. Miller [233] proposed chip based matched filter in context of MMSE optimization for the detection for MC-CDMA system. Petre *et al* [234] proposed a combination of **MMSE** and partial PIC scheme for MC-CDMA. The effect of the overlapping between successive carriers of a MC-CDMA system on the performance in a multipath fadinp channel was investigated by Said [235]. Zhong [236] proposed partial sampling MMSE

(PS-MMSE) receiver. Unlike conventional MMSE receiver. PS-MMSE demodulates the received signal at a sampling rate significantly higher than symbol rate. This helped to eliminate the problem of timing estimation in the classical MMSE receiver. Kalofonos [237] and Shi [238] investigated the multipath Rayleigh fading channels. Kafle and Sesay [239] presented iterative receivers based on subspace approach. Brunel [240] presented a novel maximum likelihood detection scheme. This scheme. as opposed to the classical optimum maximum likelihood detector (OMLD) which has exponential complexity, had polynomial complexity. Brunel modeled the MC-CDMA system as a sphere packing lattice and a low-complexity optimum lattice decoder. the sphere decoder. is applied to jointly detect all users. The uplink in a wireless mobile system is asynchronous. The timing mismatch in asynchronous systems results in damaged orthogonality among spreading codes and subcarriers as well. Consequently, this mismatch significantly degrades receiver's BER performance. Liu *et al* [241] investigated the performance of asynchronous MC-CDMA for frequency selective multipath fading channels.

Recently Jeffery [242] proposed a low complexity successive interference cancellation (SIC) scheme for multipath fading channels. He has presented two different models for MC-SIC. The first model is an uncoded system which is equipped with an interference cancellation block. The second system uses low-rate super-orthogonal codes for spreading, and each coded symbol is placed on a subcarrier rather than simple replicates of each bit. MC-CDMA is also a candidate for ultra wideband communication. This is evident from the work carried out by Jiangzhou [243]. By assuming Nakagami fading channel, he has analyzed the performance of MC-CDMA in the presence of narrowband interference. Junqiang *et al* [244] presented MAP based iterative ***MUD*** for coded MC-

CDMA systems at much reduced computational cost than other coded MC-CDMA systems. Another variation of the MC-CDMA was proposed recently by Xiaodong *et al* [245]. The idea was to divide the users into groups and then assigning unique spreading codes to the users in the group. However, one group of users was transmitted on one frequency. Due to increase in processing power and decrease in processing cost, researchers are now attracted towards using evolutionary techniques for multiuser detection of MC-CDMA systems. Techniques like Genetic algorithms and Ant colony optimization (ACO) have recently been used for the purpose. Wei and Hanzo [246] have used Genetic algorithm for Synchronous as well as asynchronous [247] MC-CDMA systems. Samir *et al* [248]-[251] have used ACO for MC-CDMA systems

MC-DS-CDMA was introduced in [252]. The transmitter for this system for k^{th} user is shown in Fig 3.2. Data stream of the k^{th} user is first converted from serial into a parallel (frame) and then each bit in the frame is spread in time domain. Each spread sequence then modulates one of the M orthogonal carriers. This technique is particularly useful for uplink.

Kondo and Milstein [253] investigated a MC-DS-CDMA system in which they applied repetition coding to transmit bandlimited DS-CDMA waveforms. whose bandwidths summed up to that of a comparable single-carrier DS-CDMA system. Xu *et al* [254] presented MMSE interference suppression for MC-DS-CDMA systems. Douglas [255] carried forward the work done by Kondo and presented convolutionally coded MC-DS-CDMA Systems for a Multipath Fading Channel. Lin proposed successive interference cancellation (SIC) scheme for MC-DS-CDMA system [256], and then analyzed [257] its performance using band-limited spreading waveforms to prevent self-interference.

Namgoong [258] investigated the performance of MC-DS-CDMA system for frequency selective fading channel by using tapped delay line (TDL) channel model. The performance of generalized MC DS-CDMA system over Nakagami-m fading channels

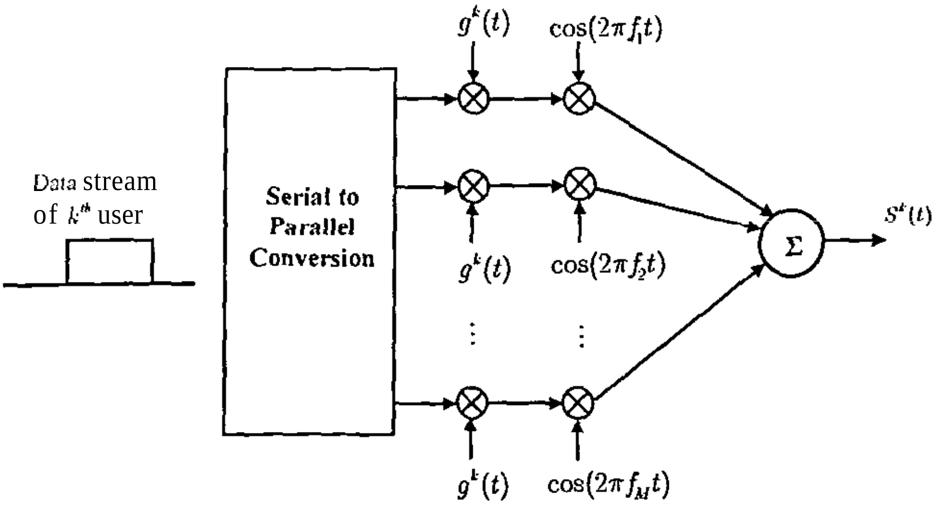


Fig. **3.2:** Transmitter for k^{th} user for MC-DS-CDMA system

and effect of subcarrier spacing on the performance was explored by Liang and Hanzo [259]. They also studied the effect of using various chip waveforms for the generalized MC-DS-CDMA system [260]. Some other contributions in this regard were made by Sener and Hanzo *et al* [261] -[264].

The third scheme in time domain spreading is the multitone CDMA (MT-CDMA). This scheme was introduced by Vandendorpe [265]-[267]. MT-CDMA has not been given much attention. A few researchers have investigated various aspects of MT-CDMA [268]-[277].

3.3 Problem formulation of MUD of Synchronous MC-CDMA System

The system model for MC-CDMA system is shown in Fig. **3.3.** Since we are considering synchronous MC-CDMA, we consider only one time slot. Each bit of the k^{th} user,

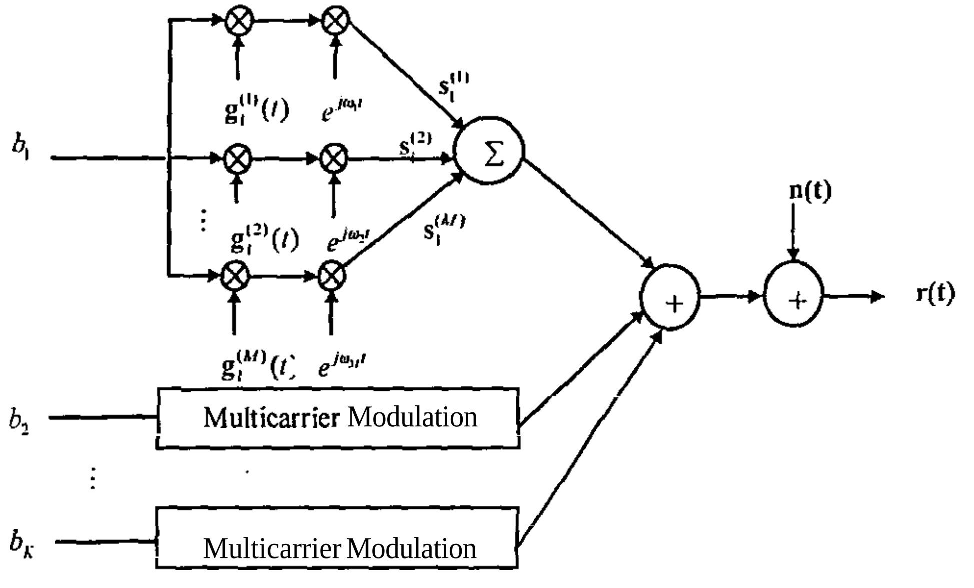


Fig.3.3: Transmitter of MC-CDMA System

b $\in \{1,-1\}$ is spread in parallel on M spreading signatures, denoted by $g_k^{(m)}(t)$, $m = 1,2\ldots M$. Length of each signature is N, given by $N = T_b/T_c$ where T_c is chip interval and T_b is bit interval. The bit b_k spread by $g_k^{(m)}(t)$ is mapped onto the m^{th} subcarrier $e^{j\omega_m t}$. The transmitted symbol of k^{th} user on m^{th} subcarrier is given as

$$s_k^{(m)}(t) = A_k g_k^{(m)}(t) b_k e^{j\omega_m t} \qquad (3.3.1)$$

where A_k is the amplitude of the k^{th} user. The composite signal traveling on the m^{th} sub carrier. is given as

$$s^{(m)}(t) = \sum_{k=1}^{K} A_k g_k^{(m)}(t) b_k e^{j\omega_m t} \tag{3.3.2}$$

Symbol $s^{(m)}(t)$ propagates through Rayleigh flat fading (RFF) channel whose complex gain is given as $\alpha^{(m)} e^{j\varphi_m}$, where $\alpha^{(m)}$ is Rayleigh distributed channel gain and phase φ_m has flat distribution over the interval $[0, 2\pi]$. The received signal on m^{th} subcarrier is given as

$$r_m(t) = \sum_{k=1}^{K} A_k g_k^{(m)}(t) b_k \alpha^{(m)} e^{j(\omega_m t + \varphi_m)} + n^{(m)}(t) \tag{3.3.3}$$

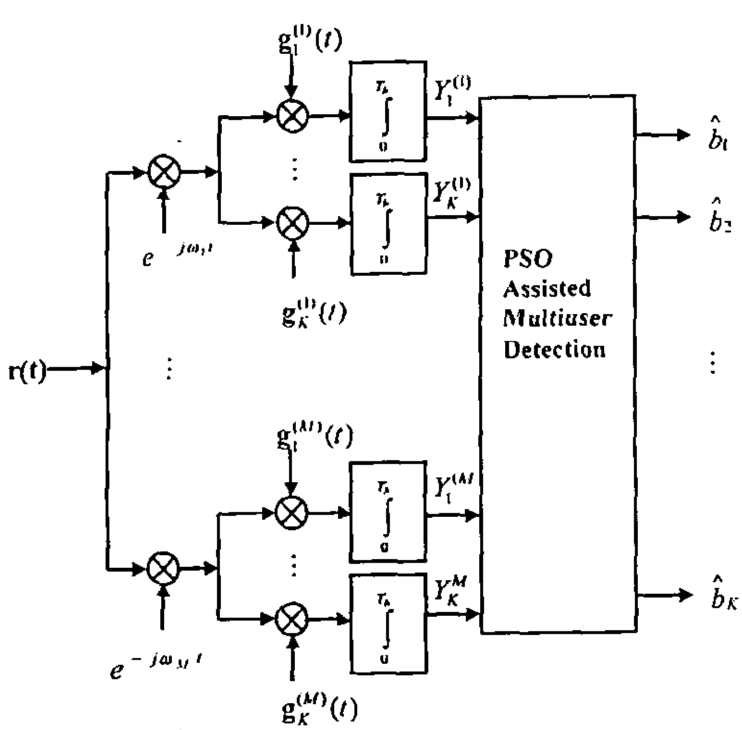


Fig 3.4: Proposed Synchronous MC-CDMA Receiver

The receiver model is shown in Fig. 3.4. The received signal is treated on carrier basis at the receiver end. After demodulation of the m^{th} carrier and passing through matched filter $g_j^{(m)}(t)$, we get.

$$Y_j^{(m)} = \int_0^T r'_{(m)}(t) g_j^{(m)} dt = \sum_{k=1}^{K} \mathrm{A}_k b_k \rho_{jk}^{(m)} \alpha^{(m)} e^{j\phi_m} + \mathrm{z}_j^{(m)} \tag{3.3.4}$$

where $r'_{(m)}$ is demodulated version of $r_m(t)$. This can also be written in matrix form as follows

$$\mathbf{Y}^{(m)}(i) = [y_1^{(m)}(\mathrm{i}) \ y_2^{(m)}(\mathrm{i}) \ \ldots y_K^{(m)}(i)] \tag{3.3.5}$$

$$\mathbf{Y}_m = \mathbf{R}^{(m)} \mathbf{H}^{(m)} \mathbf{A}_b + \mathbf{z}_m \tag{3.3.6}$$

where

$$\mathbf{H}^{(m)} = \alpha_m e^{j\phi_m} \mathbf{I}$$

$$\mathbf{A}_b = [A_1 b_1 \ A_2 b_2 \cdots A_K b_K]^T$$

$$\mathbf{z}_m = \left[z_1^{(m)} \ z_2^{(m)} \cdots z_K^{(m)}\right]^T$$

$$\rho_{jk} = \int_{\tau_k - \tau_j}^{T_b} g_j(t) g_k(t - \tau_j - \tau_k) dt \qquad \text{for } j < k$$

$$\rho_{kj} = \int_0^{T_b + \tau_k - \tau_j} g_j(t) g_k(t + \tau_j - \tau_k) dt \qquad \text{for } k < j$$

$$\rho'_{kj} = \int_0^{\tau_j - \tau_k} g_j(t) g_k(t - T_b + \tau_j - \tau_k) dt \qquad \text{for } k < j$$

and

$$\mathbf{R}^{(m)} = \begin{bmatrix} \rho_{11}^{m} & \rho_{12}^{m} \cdots \rho_{1K}^{m} \\ \rho_{21}^{m} & \rho_{22}^{m} \cdots \rho_{2K}^{m} \\ \rho_{K1}^{m} & \rho_{K1}^{m} \cdots \rho_{KK}^{m} \end{bmatrix}$$

For optimum **MUD** in **MC-CDMA,** the objective function to be maximized is given as

$$\begin{aligned} J(\mathbf{b}) &= \sum_{m=1}^{M} J^{(m)}(\mathbf{b}) \\ &= \sum_{m=1}^{M} \left\{ 2\,\mathrm{Re}\left[\mathbf{A}_b^T \mathbf{H}^{*(m)} \mathbf{Y}^{(m)}\right] - \mathbf{A}_b^T \mathbf{H}^{(m)} \mathbf{R}^{(m)} \mathbf{H}^{*(m)} \mathbf{A}_b \right\} \end{aligned} \qquad (3.3.7)$$

Thus there are 2^{K} possible combination of b out of which one has to be chosen

$$\mathbf{b}_{opt} = \arg\left\{\max_{\mathbf{b}}\left[J(\mathbf{b})\right]\right\} \qquad (3.3.8)$$

To avoid this exhaustive search which increases exponentially with the number of users, GAs were invoked for near optimum solution. In the same spirit. PSO is proposed for near optimum solution with much less complexity and hence without any need for exhaustive search. Specifically two versions of PSO have been used for the purpose. One is the conventional PSO, known as hard PSO (HPSO) that uses hard decisions on the particles. The second version of PSO, known as sot? PSO applies soft decision on the particles.

3.3.1 HPSO Assisted MUD for Synchronous MC-CDMA

PSO assumes that each possible solution is a particle in the swarm. Any i^{th} particle or solution is written as

$$\mathbf{b}_i = [b_{i1} \;\; b_{i2} \;\; \ldots b_{im} \ldots b_{iK}] \tag{3.3.9}$$

where K is the number of users and b_{im} is the position of m^{th} user on i^{th} particle. Each particle position b_{im} has a corresponding velocity v_{im}.

The algorithm for PSO assisted MC-MUD has the following steps:

Step1: Initialization of the particles in the swarm can be done in two different manners. One way is to create particles at random without bias i.e. $\mathbf{b}_i = [b_{i1} \;\; b_{i2} \cdots b_{iK}] \quad 1 \le i \le p$

where p is the number of particles in the swarm and it depends on the number of users in the system. Second way is to create population of particles with bias. We proceed through the second way in which we use maximum ratio combining principle for matched tilter output and then carry out the hard decision. The output of k^{th} user is given

$$\hat{b}_k = \operatorname{sgn}\left\{ \sum_{m=1}^{M} Y_k^{(m)} \alpha_k^{(m)} e^{j\varphi_m} \right\} \tag{3.3.10}$$

once we have the output of K users, $\mathbf{b} = [\hat{b}_1 \cdots \hat{b}_K]$, we treat it as particle and create the rest of the $(\mathbf{p}-1)$ particles through toggling one of the bits at random.

Step 2: Using (3.3.7) as fitness function, the fitness of each particle is calculated. Particle with the highest fitness is taken as best performer of the population. Also looking at the history of each particle we record their corresponding personal best position.

Step 3: The velocity v_{im} of each particle corresponding to each bit position b_{im} is updated using personal best and global best particles,

$$v_{im}(n) = v_{im}(n-1) + \varphi_1(p_{im} - b_{im}(n-1)) + \varphi_2(p_{gm} - b_{im}(n-1)) \qquad (3.3.11)$$

As explained earlier. φ_1 and φ_2 are the weights of the stochastic acceleration terms that pull each particle towards p_{im} and p_{gm} position. We have taken $\varphi_1 = \varphi_2 = 2$. We suggest a fundamental change in (3.3.11). We take $\varphi_1 = \varphi_2 = \varphi$ and introduce a factor β

$$v_{im}(n) = v_{im}(n-1) + \beta\varphi(p_{im} - b_{im}(n-1)) + (1-\beta)\varphi(p_{gm} - b_{im}(n-1)) \qquad (3.3.12)$$

By varying β we can decide to prefer personal or collective intelligence. One can change this parameter to give best possible results. The velocity v_{im} has the following bounds

$$|v_{im}| \leq V_{max}$$

where V_{max} has been kept as 4 in simulations.

The decision to be made by the particle position to be $+1$ or -1 depends upon the particles predisposition, which is obtained by v_{im}. If v_{im} is high, the individual is more probable to acquire a value of $+1$. This is given by the statement

$$if(rand(\ ) < S(v_{im})), \quad then \quad b_{im} = -1 \quad else \quad b_{im} = 1$$

where

Step 3: The velocity v_{im} of each particle corresponding to each bit position b_{im} is updated using personal best and global best particles,

$$v_{im}(n) = v_{im}(n-1) + \varphi_1(p_{im} - b_{im}(n-1)) + \varphi_2(p_{gm} - b_{im}(n-1)) \qquad (3.3.11)$$

As explained earlier. φ_1 and φ_2 are the weights of the stochastic acceleration terms that pull each particle towards p_{im} and p_{gm} position. We have taken $\varphi_1 = \varphi_2 = 2$. We suggest a fundamental change in (3.3.11). We take $\varphi_1 = \varphi_2 = \varphi$ and introduce a factor β

$$v_{im}(n) = v_{im}(n-1) + \beta\varphi(p_{im} - b_{im}(n-1)) + (1-\beta)\varphi(p_{gm} - b_{im}(n-1)) \qquad (3.3.12)$$

By varying β we can decide to prefer personal or collective intelligence. One can change this parameter to give best possible results. The velocity v_{im} has the following bounds

$$|v_{im}| \leq V_{max}$$

where V_{max} has been kept as 4 in simulations.

The decision to be made by the particle position to be $+1$ or -1 depends upon the particles predisposition, which is obtained by v_{im}. If v_{im} is high, the individual is more probable to acquire a value of $+1$. This is given by the statement

$$if(rand(\ ) < S(v_{im})), \quad then \quad b_{im} = -1 \quad else \quad b_{im} = 1$$

where

$$S(v_{im}) = \frac{1}{1+\exp(-v_{im})}$$

Step4: The above steps are repeated until the minimum bit error condition is satisfied or maximum number of iterations has been performed.

3.3.2 SPSO Assisted MUD for Synchronous MC-CDMA

In SPSO the particles in the swarm are initialized at soft values instead of hard values. This is done by using $\tanh(.)$ instead of $\mathrm{sgn}(.)$ in (3.3.10), i.e.

$$\hat{b}_k = \tanh\left(\sum_{m=1}^{M} Y_k^{(m)} \alpha^{(m)} e^{-j\phi_m}\right) \tag{3.3.13}$$

The rest of the panicles are created by flipping the sign or by adding a very small number to a randomly selected location of a particle. The fitness of particles is evaluated by using (3.3.7). Velocity is updated by using (3.3.12). Like HPSO velocity is bounded above and below. Particle position is updated through soft decision as follows,

$$if(rand(\ ) > S(v_{im})).\quad then\quad b_{im} = -1 + S(v_{im})\quad else\quad b_{im} = S(v_{im}) \tag{3.3.14}$$

Where
$$S(v_{im}) = \frac{1}{1+\exp(-\gamma v_{im})} \tag{3.3.15}$$

It must be kept in mind that γ is one of the important parameters of SPSO. In HPSO $\gamma = 1$. If $\gamma \rightarrow \infty$ we have pure step function. Thus with $\gamma = 0.25$, the variation of $S(v_{im})$ is very gradual around $v_{im} = 0$. But as γ is increased it becomes more and more abrupt tending towards more hardness. We have observed after test and trial that by gradually increasing γ from 0.25 the results are better in terms of bit error rate than keeping the γ static ($\gamma = 1$) as in HPSO. Upon the completion of the iterations. hard decision is carried out on the global best particle to get the estimated data bits.

3.3.3 Performance of PSO assisted MUD for Synchronous MC-CDMA System

In order to test the performance of PSO thoroughly, simulations have been carried out for systems with various number of users and for various computational complexities. The product of number of particles, P, and number of iterations, Y. has been taken as computational complexity. The reason behind this is to compare the proposed schemes with the results obtained by using Genetic Algorithm by H. Wei [246] who has defined P.Y as complexity. Throughout the simulations, 32-chip Walsh codes have been used for spreading the data bits and each user bit is sent on 4 different carriers. The channel is assumed to be Rayleigh distributed with flat phase uniformly distributed between $[0, 2\pi]$. These results have been compared with the results of GA for same computational complexity. Bit error rate (BER) has been taken as performance criteria for these simulations.

Fig. 3.5 shows simulation results in terms of BER performance. for a system with 10 users. Here the hard PSO (HPSO) is used with computational complexities of 200. 400 and 600. Fig. 3.6 shows simulation results for the same system with the same complexities when SPSO is used. SPSO has given better BER results and this is because of two reasons. First reason is that the panicles have soft values which gives flexibility and the second reason is that during the execution, these soft values are gradually forced towards the limits i.e. +1 or −1 by increasing the value of γ in (3.3.15). This procedure seems natural instead of forcing a particle to have a hard value of +1 or -1.

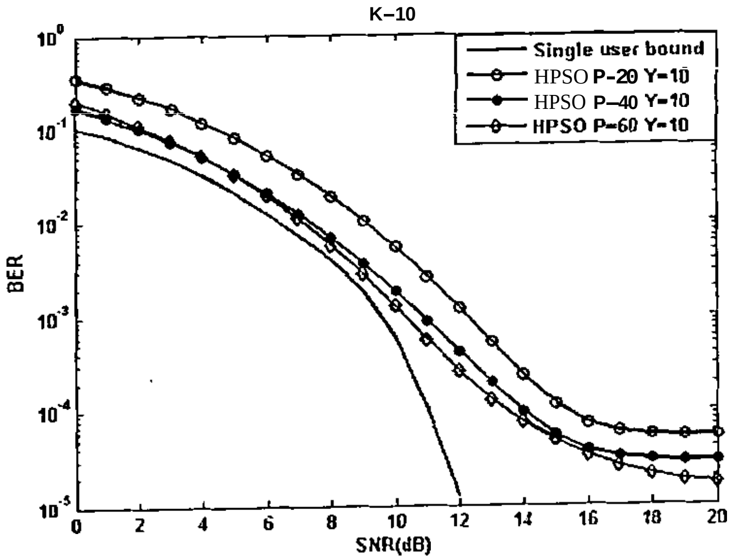


Fig. 3.5: RER Performance of HPSO For synchronous MC-CDMA system with 10 users

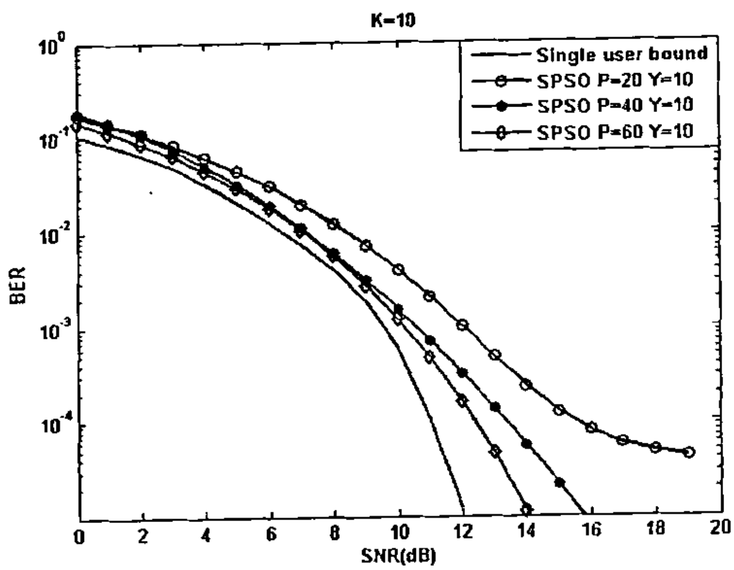


Fig. 3.6: BER Performance of SPSO for synchronous MC-CDMA system with 10 users

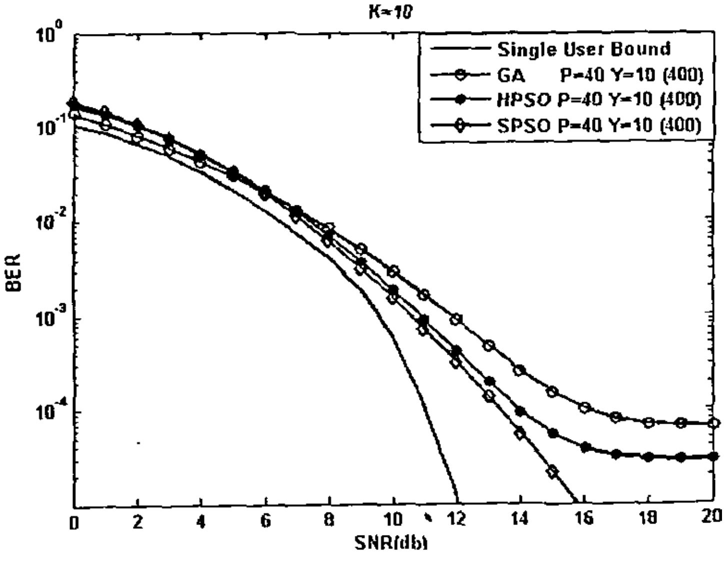


Fig. 3.7: Performance Comparison of GA. SPSO. and HPSO for synchronous MC-CDMA system with 10 users

Performance comparison of HPSO-MUD, SPSO-MUD, and GA-MUD for 10 users is shown in fig. 3.7. It is evident that both the hard and soft versions of PSO perform better than GA.

Fig. 3.8 shows results for systems with 15 users, when HPSO is used. The computational complexities of the results in this simulation are 600, 800, and 1000. Fig. 3.9 shows results for the same environment when SPSO has been used. Since no results ere available for GA-MUD for 15-user system. therefore no comparison is provided. However it is quite obvious that the results obtained by using SPSO are better than those obtained by HPSO. Similarly, fig. 3.10 shows the BER performance of HPSO for a system with 20 users. The computational complexities used for this simulation are 1000, 1500 and 2000. Fig. 3.1 I shows results for the same system, under the same environment when SPSO is used.

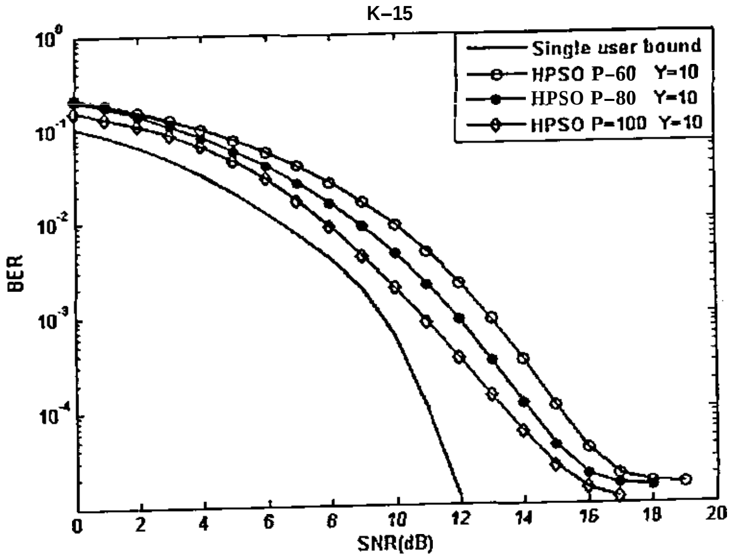


Fig. 3.8: BER Performance of HPSO for synchronous MC-CDMA system with 15 users

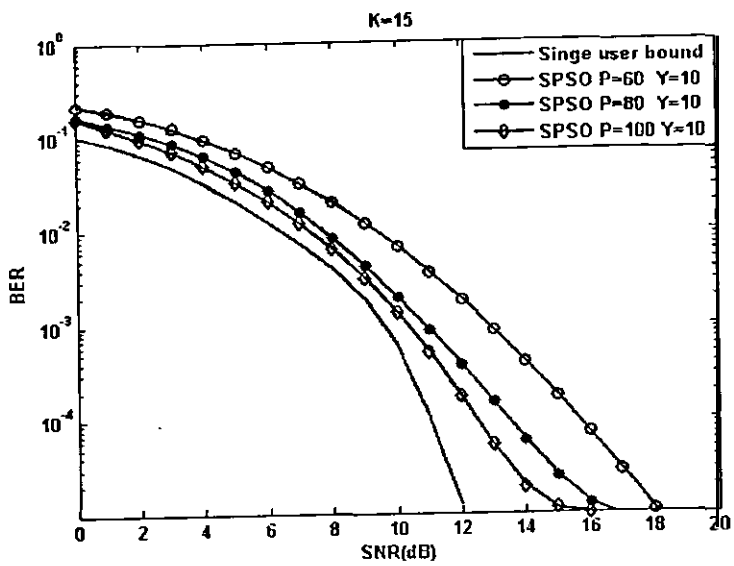


Fig. 3.9: BER Performance of SPSO for synchronous MC-CDMA system with 15 users

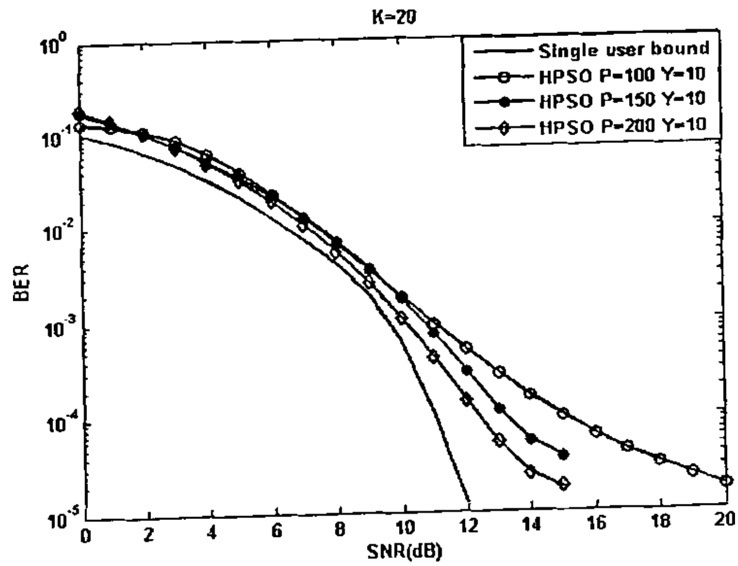


Fig. 3.10: BER Performance of HPSO for synchronous MC- CDMA system with 20 users

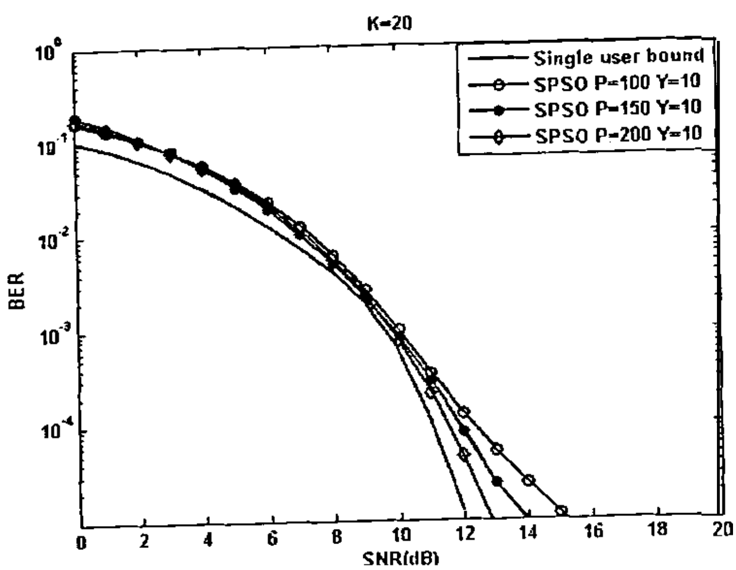


Fig. 3.11: BER Performance of SPSO for synchronous MC-CDMA system with 20 users

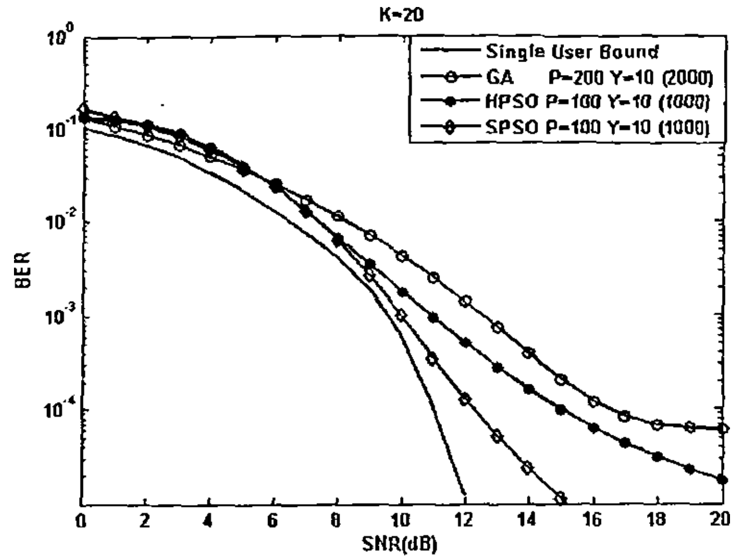


Fig. 3.12: Performance Comparison of GA, SPSO, and HPSO for synchronous MC-CDMA system with 20 users

Fig. 3.12 presents comparison of HPSO, SPSO and GA under the same simulation conditions. Once again SPSO is leading the other two.

The nest two graphs, Fig. 3.13 and Fig. 3.14, show BER performance of HPSO and SPSO for a system with 32 users. The computational complexities for both of these simulations are 1600, 2400 and 3200. One can see an obvious deterioration in BER performance as compared to the previous results. As the number of users increase. the computational cost goes up exponentially [1]. From this point of view PSO-MUD has a computational complexity that is almost negligible as compared to the optimum detector by Verdu [1] that has computational cost of 2^{32} for this system. Since no results were available for GA's for 32 user system, hence it has not been included in the simulations.

The computational complexity of SPSO and HPSO deserves a comment. The complexity of SPSO is slightly more than that of HPSO.

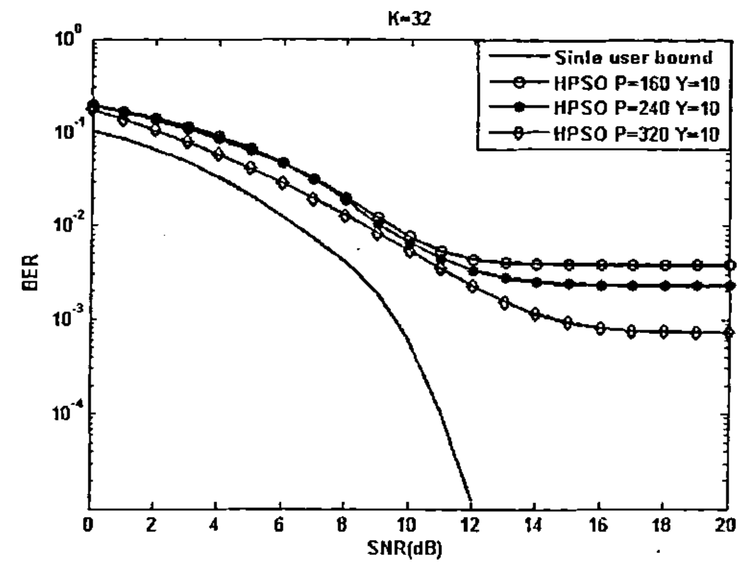


Fig. 3.13: BER Performance of HPSO for synchronous MC-CDMA system with 32 users

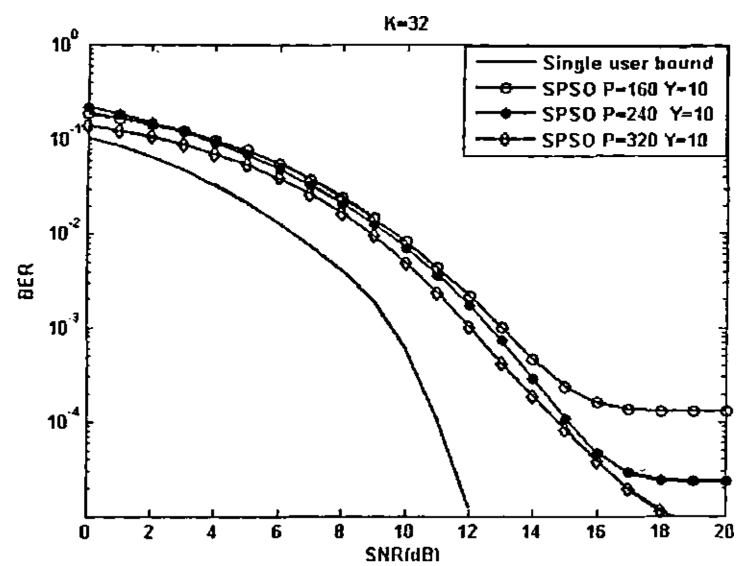


Fig. 3.14: BER Performance of SPSO for synchronous MC-CDMA system with 32 users

The reason is that with $\gamma = 0.25$ in SPSO, the calculation of $S(v_{im})$ will take little more time because of an additional multiplication required for v_{im}. Moreover. since $b_{in} = -1 + S(v_{in})$ or $b_{im} = S(v_{in})$, it will take more storage than in the case of HPSO where $b_{im} = \pm 1$.

3.4 Problem formulation of MUD of Asynchronous MC-CDMA System

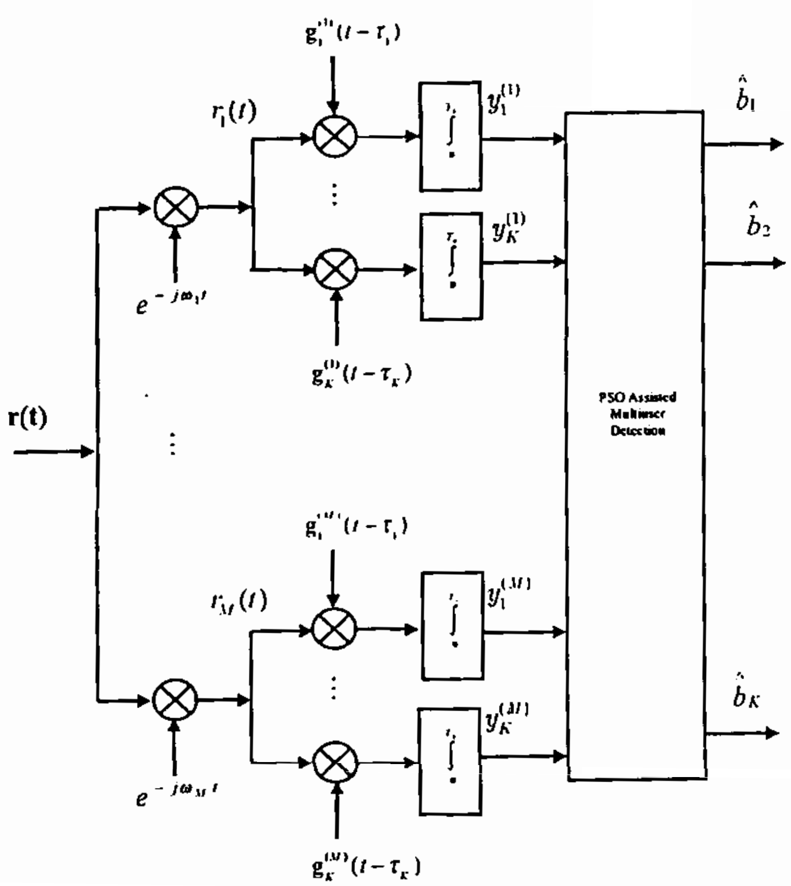
The uplink in a wireless communication system is inherently of asynchronous nature. This is because of timing mismatch between the incoming signal and the locally generated spreading sequence and carrier frequency. As a result the performance in terms of BER is degraded considerably.

In this section we discuss an asynchronous MC-CDMA system with K users. The transmitter for such a system is the same as shown in section 3.3. In a typical wireless asynchronous system a bit of interest faces interference due to two neighboring bits. Therefore in order to detect the bit of interest, the two neighboring bits have to be estimated first. The receiver for the system is shown in Fig. 3.15. We define a transmission delay τ_k associated with k^{th} user. For the sake of simplicity and without loss of generality it can be assumed that $0 \le \tau_1 \le \tau_2 \le \cdots \le \tau_K < T_b$. For the sake of simplicity it is assumed that *for* any k^{th} delay, $\tau_k = l_k T_c$, where $0 \le l < N$. We also assume that the delays are known at the receiver. The channel is taken to be slowly fading with Rayleigh distribution. It is further supposed that an additive white Gaussian noise (AWGN) corrupts the composite signal. The receiver, shown in Fig. 3.15, employs a bank of filters, matched to the delayed signatures. The received signal at the base station on m^{th} subcarrier after coherent detection is given by

$$\mathbf{r}_m(t) = \sum_{i=-\infty}^{\infty} \sum_{k=1}^{K} A_k g_k^{(m)}\left(t - iT_b - \tau_k\right)\, b_k \alpha_k^{(m)} e^{j\phi_{km}} + \mathbf{n}_m(t) \qquad (3.4.1)$$

For m^{th} subcarrier, the output of the matched filter, at i^{th} interval, can be written in vector form as follows,

$$\mathbf{Y}^{(m)}(i) = [y_1^{(m)}(i)\ \ y_2^{(m)}(i)\ \ \ldots y_K^{(m)}(i)] \qquad (3.4.2)$$



where

Fig. 3.15: Proposed Asynchronous MC-CDMA Receiver

$$y_k^{m}(i) = \int_{iT_b-\tau_k}^{(i+1)T_b+\tau_k} r_m(t) g_k^{m}(t - iT_b - \tau_k)dt \qquad (3.4.3)$$

At any observarion interval. the current bit of interest $b(i)$ gets overlapped with the previous bit $b(i-1)$, and the nest bit $b(i+1)$. As a result only three terms survive in (3.4.3),

$$\begin{aligned} y_k^{(m)} &= \sum_{j=1}^{K} (A_j \alpha_j^{(m)} b_j(i) e^{-j\varphi_m}) \int g_j^{(m)}(t - iT_b - \tau_j) g_k^{(m)}(t - iT_b - \tau_k)dt \\ &+ \sum_{j=1}^{K} (A_j \alpha_j^{(m)} b_j(i-1) e^{-j\varphi_m}) \int g_j^{(m)}(t - iT_b + T_b - \tau_j) g_k^{(m)}(t - iT_b - \tau_k)dt + \\ &+ \sum_{j=1}^{K} (A_j \alpha_j^{(m)} b_j(i+1) e^{-j\varphi_m}) \int g_j^{(m)}(t - iT_b - T_b - \tau_j) g_k^{(m)}(t - iT_b - \tau_k)dt \end{aligned}$$

$$(3.4.4)$$

Using vector notation, the output of the matched filter at i^{th} interval can be written as

$$\mathbf{Y}^{(m)}(i) = \mathbf{R}_m[1]\mathbf{H}^m\mathbf{A}\mathbf{b}(i-1) + \mathbf{R}_m[0]\mathbf{H}^m\mathbf{A}\mathbf{b}(i) + \mathbf{R}_m^T[1]\mathbf{H}^m\mathbf{A}\mathbf{b}(i+1) + \mathbf{n}_m \quad (3.4.5)$$

where

$$\mathbf{R}_m[0] = \begin{vmatrix} 1 & \rho_{12} & \rho_{13} \cdots \rho_{1K} \\ \rho_{21} & 1 & \rho_{23} \cdots \rho_{2K} \\ \vdots & \vdots & \vdots \\ \rho_{K1} & \rho_{K2} & & 1 \end{vmatrix} \qquad (3.4.6)$$

and

$$\rho_{jk} = \int_{\tau_k - \tau_j}^{T_b} g_j(t) g_k(t - \tau_j - \tau_k) dt \qquad \text{for } j < k$$
$$\rho_{kj} = \int_{0}^{T_b + \tau_k - \tau_j} g_j(t) g_k(t + \tau_j - \tau_k) dt \qquad \text{for } k < j \tag{3.4.7}$$

and

$$\mathbf{R}_m[1] = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ \rho'_{21} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ \rho'_{K1} & \rho'_{K2} & & & 0 \end{bmatrix} \tag{3.4.8}$$

where

$$\rho'_{kj} = \int_{0}^{\tau_j - \tau_k} g_j(t) g_k(t - T_b + \tau_j - \tau_k) dt \qquad \text{for } k < j \tag{3.4.9}$$

Unlike the synchronous case. one has to deal with three vectors simultaneously, $\mathbf{b}(i-1)$, $\mathbf{b}(i)$ and $\mathbf{b}(i+1)$. Keeping (3.4.5) into account. the objective function of OMLD for the m^{th} subcarrier is given as

$$J^{(m)}(\mathbf{b}(i)) = \arg\left\{\min_{\mathbf{b}(i-1)\mathbf{b}(i+1)} E[\mathbf{n}_m^T \mathbf{n}_m]\right\} \tag{3.4.10}$$

where $\mathbf{n}_m$ is noise sampling vector given by,

$$\mathbf{n}_m = \mathbf{Y}^{(m)}(i) - \mathbf{R}^{(m)}[1]\mathbf{W}_m\mathbf{A}\mathbf{b}(i-1) - \mathbf{R}^{(m)}[0]\mathbf{W}_m\mathbf{A}\mathbf{b}(i) - \mathbf{R}^{T(m)}[1]\mathbf{W}_m\mathbf{A}\mathbf{b}(i+1) \tag{3.4.11}$$

For optimum MUD for CDMA the overall objective function to be minimized becomes

$$J(\mathbf{b}) = \sum_{m=1}^{M} J^{(m)}(\mathbf{b}(i)) \tag{3.4.12}$$

The detection of $\mathbf{b}(i)$ requires the knowledge of both $\mathbf{b}(i-1)$ and $\mathbf{b}(i+1)$. If we assume that $\mathbf{b}(i-1)$ is already known to us from the previous observation interval. we have to find the optimal $\mathbf{b}(i)$ and $\mathbf{b}(i+1)$. But $\mathbf{b}(i+1)$ is not known in advance. For optimal detection of $\mathbf{b}^{(i)}$, there are 2^{2K} combination of $\mathbf{b}$ out of which one has to be chosen.

$$\mathbf{b}_{opt}(i) = \arg\left\{\min_{\mathbf{b}(i),\mathbf{b}(i+1)}\left[J(\mathbf{b})\right]\right\} \tag{3.4.13}$$

It is obvious that the computational requirement for this problem is three times more than that of MUD of synchronous **MC-CDMA** system described in the previous section. Among a number of suboptimal approaches like Successive Interference Cancellation (SIC), Parallel Interference Cancellation (PIC), and evolutionary computing algorithms, we propose PSO as one of the options, in the next subsection.

3.4.1 HPSO Assisted MUD for Asynchronous MC-CDMA

Any j^{th} particle in the swarm is written as concatenation of $b_j^{(i)}$ and $b_j^{(i+1)}$

$$\mathbf{b}_j = [b_{j1}^{(i)}\ b_{j2}^{(i)}\ \ldots b_{jK}^{(i)}, b_{j1}^{(i+1)} b_{j2}^{(i+1)} \ldots b_{jK}^{(i+1)}] \tag{3.4.14}$$

where K is the number of users and $b_{jm}^{(t)}$ is the position of the j^{th} particle and m^{th} user in time slot t, which is either i or $i+1$. Each particle position $b_{jm}^{(t)}$ has a corresponding velocity $v_{jm}^{(i)}$. Both of these parameters are updated using individual best as well as the global best particles so far. Our algorithm for PSO assisted MUD for asynchronous MC-CDMA has the following steps:

Step1: We have adopted the biased initialization by employing the maximum ratio combining principle and performing hard decisions on the matched filter output. The output of k^{th} user is given as

$$\hat{\mathbf{b}}_k^{(i)} = \operatorname{sgn}\left(\sum_{m=1}^{M} \mathrm{Y}_k^{(m)} \alpha_k^{(m)} c^{j\varphi_{km}}\right) \tag{3.4.15}$$

The output of data vector $\hat{\mathbf{b}}^{(i)} = \left[\hat{b}_1^{(i)} \ldots \hat{b}_K^{(i)}\right]$ for i^{th} time slot is taken as first K bits of the first particle. The first K bits, for the rest of the particles are created through toggling one of the bits of $\hat{\mathbf{b}}^{(i)}$ at random. This constitutes the population for i^{th} particle. For $\mathbf{b}^{(i+1)}$ to be used in (3.4.11) a number of approaches have been investigated in the literature. Wei and Hanzo [247] have used maximum ratio combining as well as random initialization for $\mathbf{b}^{(i+1)}$. We have used MRC for $\hat{\mathbf{b}}^{(i+1)}$. Thus the last K bits of each particle in the population, corresponding to panicle $\mathbf{b}^{(i+1)}$ are created by randomly flipping $\hat{\mathbf{b}}^{(i+1)}$.

Step 2: The fitness of each particle is determined using (3.4.10) as fitness function. The particle having the highest fitness is chosen as global best panicle of the population. On the basis of the previous fitness of each particle, the corresponding individual best position is updated.

Step 3: The velocity $v_{jm}^{(i)}$ of each particle corresponding to each bit position $b_{jm}^{(i)}$ is revised using individual best and global best particles,

$$v_{jm}^{(i)}(n) = v_{jm}^{(i)}(n-1) + \varphi_1(p_{jm} - b_{jm}^{(i)}(n-1)) + \varphi_2(p_{gm} - b_{jm}^{(i)}(n-1)) \quad (3.4.16)$$

where φ_1 and φ_2 are the weights of the stochastic acceleration terms that pull each particle towards p_{jm} and p_{gm} position. We have taken $\varphi_1 = \varphi_2 = 2$. The second term on the right in (3.4.16) represents the individual intelligence while the third term represents the global intelligence of the particles. We propose a fundamental change in (3.4.16) . In order to have flexible contributions from individual and global intelligence for the updation ofparticle velocity, we introduce a factor β .

$$v_{jm}^{(i)}(n) = v_{jm}^{(i)}(n-1) + \beta\varphi(p_{jm} - b_{jm}^{(i)}(n-1)) + (1-\beta)\varphi(p_{gm} - b_{jm}^{(i)}(n-1)) \qquad (3.4.17)$$

By varying β we can decide to prefer individual or collective intelligence. One can change this parameter to give *best* optimal. To avoid divergence, the velocity $v_{jm}^{(i)}$ has the following bounds

$$\left|v_{jm}^{(i)}\right| \le V_{max}$$

where V_{max} has been kept as **4** in this dissertation. In calculating v_{jm}^{i} in (3.4.17)

$$if \;\; v_{jm}^{(i)} > V_{max}, \;\; v_{jm}^{(i)} = V_{max}$$

and

$$if \;\; v_{jm}^{(i)} < -V_{max}, \;\; v_{jm}^{(i)} = -V_{max}$$

The decision about the particle position to be $+1$ or -1 depends upon the current velocity of the particle $v_{jm}^{(i)}$. The higher the value of $v_{jm}^{(i)}$, the more the probability for $+1$, i.e.

$$if(rand(\ ) < S(v_{jm}^{(i)})), \quad then \quad b_{jm}^{(i)} = -1 \quad else \quad b_{jm}^{(i)} = 1$$

Step4: Using the updated particles, new fitnesses are calculated. Individual and global best particles are updated. The steps 1 to 4 are repeated until the minimum error condition is satisfied or maximum number of iterations has been reached. Upon the termination of the algorithm, global best particle is picked up as estimated data vector $\hat{\mathbf{b}}^{(i)}$. Once the data vector $\hat{\mathbf{b}}^{(i)}$ has been estimated. it becomes $\mathbf{b}^{(i-1)}$ for the nest data vector $\mathbf{b}^{(i)}$. The whole population of particles. made for the detection of $\mathbf{b}^{(i)}$ is discarded. The data vector $\mathbf{b}^{(i-1)}$ is determined as $\mathbf{b}^{(i)}$ in the nest iteration.

3.4.2 SPSO Assisted MUD for Asynchronous MC-CDMA

As described earlier in section 3.3.2, for SPSO, the particles in the swarm are initialized at soft values instead of hard limits. This is done by using $\tanh(.)$ instead of $sgn(.)$ in (3.4.15) i.e.

$$\hat{b}_k = \tanh\left(\sum_{m=1}^{M} Y_k^{(m)} \alpha^{(m)} e^{-j\phi_m}\right) \qquad (3.4.18)$$

The rest of the population is created by perturbing the first particle. The fitness of particles is evaluated by using (3.4.10). Velocity is updated by using (3.3.12). Like

HPSO, the velocity of particles is bounded above and below. Particle position is updated through soft decision as follows,

$$if(rand(\ ) > S(v_{jm}^{(t)}),\quad then\quad b_{jm} = -1 + S(v_{jm}^{(t)})\quad else\quad b_{jm} = S(v_{jm}^{(t)})$$

where

$$S(v_{jm}^{(t)}) = \frac{1}{1+\exp(-\gamma v_{jm}^{(t)})}$$

The role of γ is the same as described in section 3.3.2. Upon the completion of ***the*** algorithm, a hard decision is carried out on the global best particle to get the estimated data bits.

3.4.3 Performance of PSO assisted MUD for Asynchronous MC-CDMA System

Simulations have been carried out for the asynchronous MC-CDMA systems with different number of users. Walsh codes are used for spreading the data bits. Each user bit is sent on 4 different carriers. The channel is assumed to be Rayleigh distributed having flat phase distribution between 0 and 2π. BER is taken as performance criteria. **Fig.** 3.16 shows performance of PSO-MUD for a system with 10 users for computational complexities of 200. 400 and 600. Fig. 3.17 shows the performance when SPSO is used in the same scenario. For this simulation, 8-chip Walsh code has been used. The performance of SPSO is better than that of HPSO which is in agreement to the synchronous case in section **3.3.** Both SPSO and HPSO have also been compared with GA-MUD [278] in fig. 3.18. The performance of PSO is comparable to GA at low

E_b / N_o ratio. However, the PSO gives better performance after 12dB. It is worth mentioning here that in the real world communication, the ratio E_b/N_o is kept greater than 12dB.

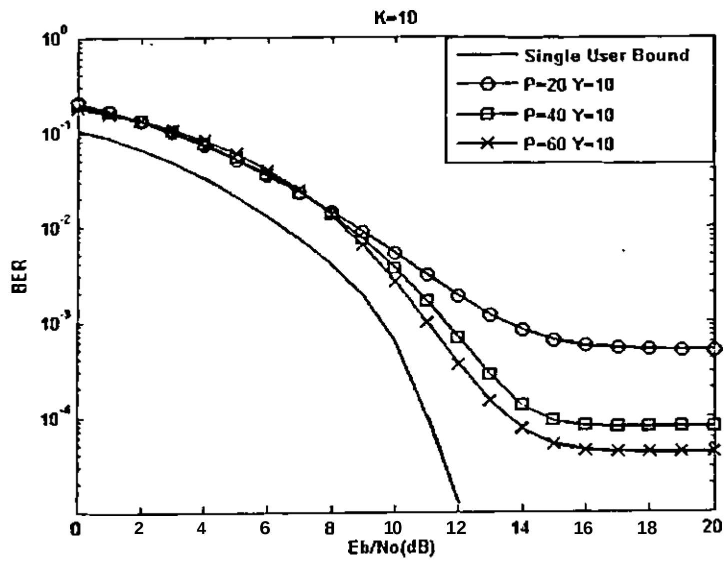


Fig. 3.16: **BER** Performance of HPSO for Asynchronous MC-CDMA system with **10** users

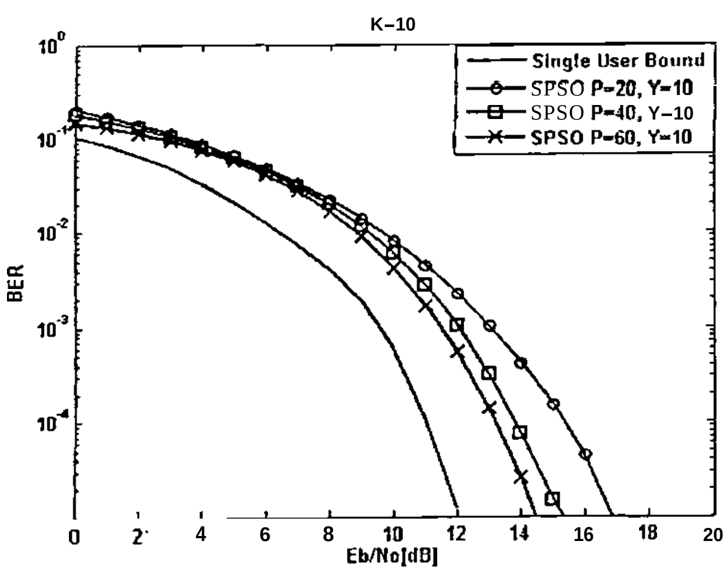


Fig. 3.17: **BER** Performance of SPSO for asynchronous MC-CDMA system with 10 users

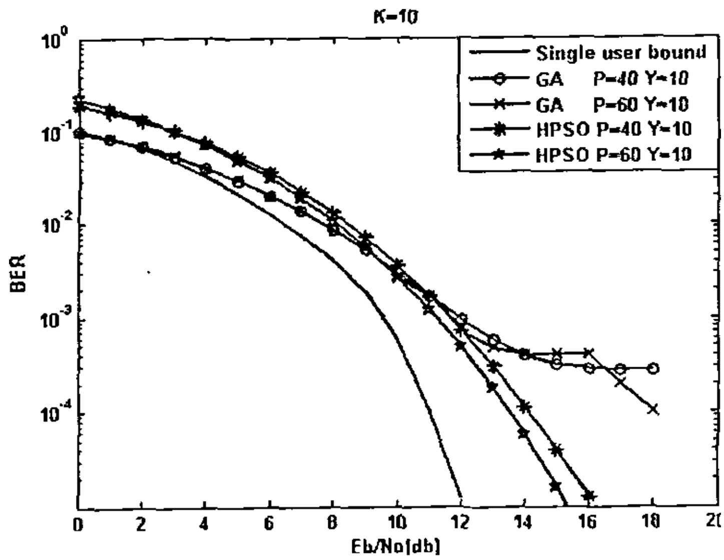


Fig. 3.18: Performance Comparison of GA and HPSO for asynchronous MC-CDMA system with 10 users

Fig. 3.19 presents the BER performance of HPSO-MUD for 15-user system. The computational complexities used are 600, 800 and 1000. For this system again, 8-chip Walsh code has bee" used for spreading the data bits. Obviously, for the same computational complexity, the results of synchronous case are better than those of asynchronous case. Fig 3.20 shows the performance of SPSO for same environment.

Fig. 3.21 presents simulation for 70-user system using HPSO. Here 16-chip Walsh code has been used. All other simulation parameters are the same. The computational complexities, used are, 1000. 1500, and 2000. The nest figure, Fig. 3.22 is for the same. system but using SPSO.

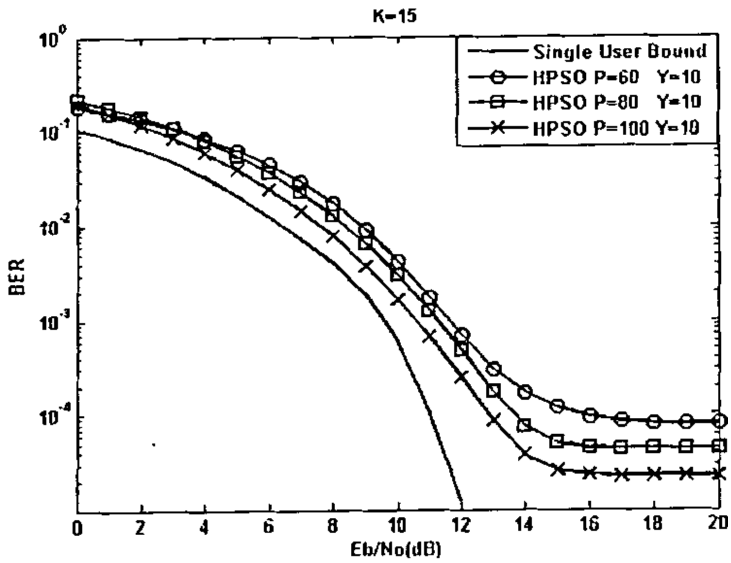


Fig. 3.19: BER Performance of HPSO for asynchronous MC-CDMA system with 10 users

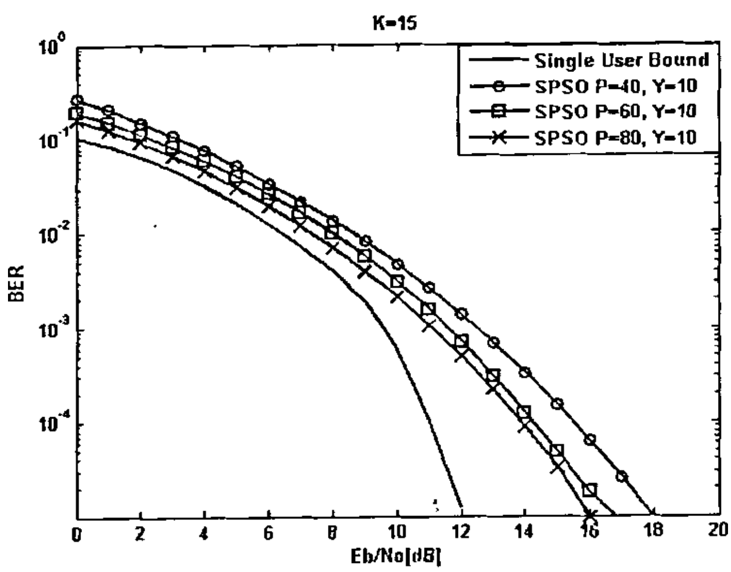


Fig. 3.20: BER Performance of SPSO for Asynchronous MC-CDMA system with 10 users

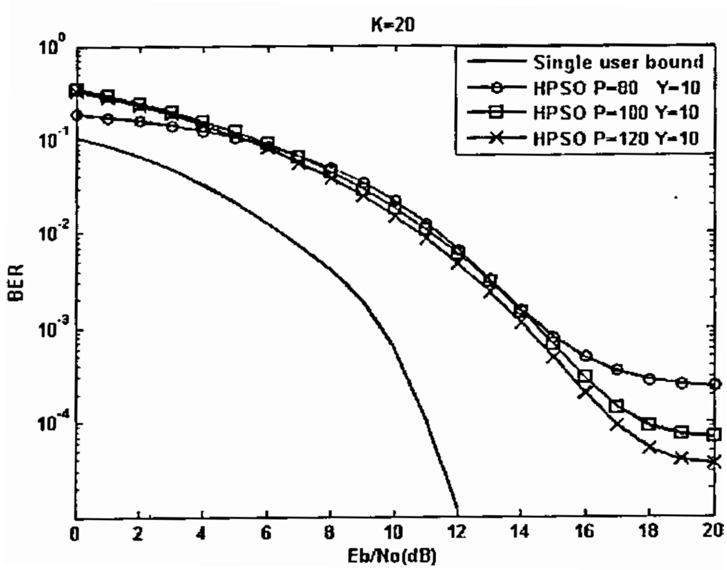


Fig. 3.21: BER Performance of HPSO for Asynchronous MC-CDMA system with 20 users

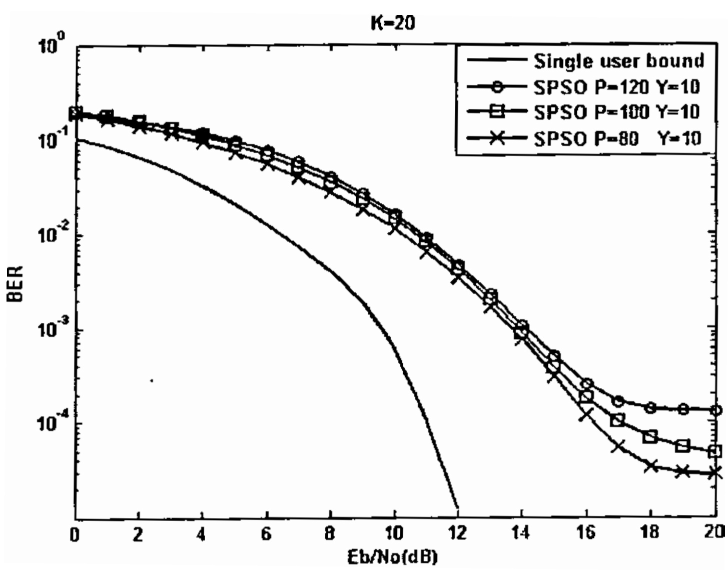


Fig. 3.22: BER Performance of SPSO for Asynchronous MC-CDMA system with 20 users

The performance of PSO as compared to **GA,** keeps on improving with SNR. Especially the SPSO is outperforming the other competitors. The next two figures show the simulation results for 32-user system for HPSO and SPSO respectively. Here a Walsh code of 32 chips has been used for spreading the data bits. The computational complexities for both HPSO and SPSO cases are 1600. 2400 and 3200. The results with **GA-MUD** for this system were not available. hence its comparison with PSO is not provided.

The BER performance for 32 users in asynchronous detection is obviously quite lower than the performance for synchronous detection with same number of users. The reason behind this is worth mentioning. The asynchronous case is quite different from synchronous case because in asynchronous case the search space is 2^{3K} instead of 2^{K} as in synchronous detection. The computational cost increases explosively by increasing the number of users and is much larger than that required in synchronous detecfion. However. the computational requirement of PSO is almost negligible in comparison with the optimum detector proposed by Verdu [I],but still best among the best among other suboptimum techniques.

The performance of PSO as compared to GA, keeps on improving with SNR. Especially the SPSO is outperforming the other competitors. The next two figures show the simulation results for 32-user system for HPSO and SPSO respectively. Here a Walsh code of **32** chips has been used for spreading the data bits. The computational complexities for both HPSO and SPSO cases are 1600,2400 and 3200. The results with GA-MUD for this system were not available, hence its comparison with PSO is not provided.

The BER performance for **32** users in asynchronous detection is obviously quite lower than the performance for synchronous detection with same number of users. The reason behind this is worth mentioning. The asynchronous case is quite different from synchronous case because in asynchronous case the search space is 2^{3K} instead of 2^{K} as in synchronous detection. The computational cost increases explosively by increasing the number of users and is much larger than that required in synchronous detection. However, the computational requirement of PSO is almost negligible in comparison with the optimum detector proposed by Verdu [1], and is better among the other suboptimum techniques.

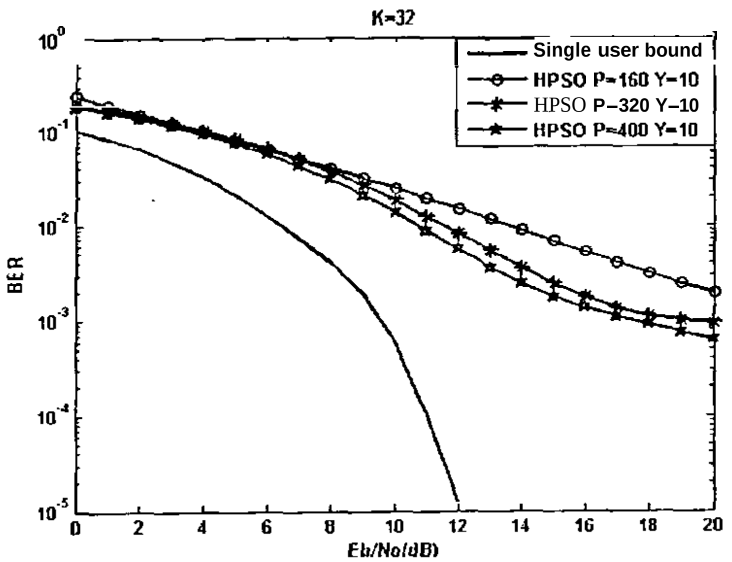


Fig. 3.23: BER Performance of HPSO for Asynchronous MC-CDMA system a svstem with **32** users

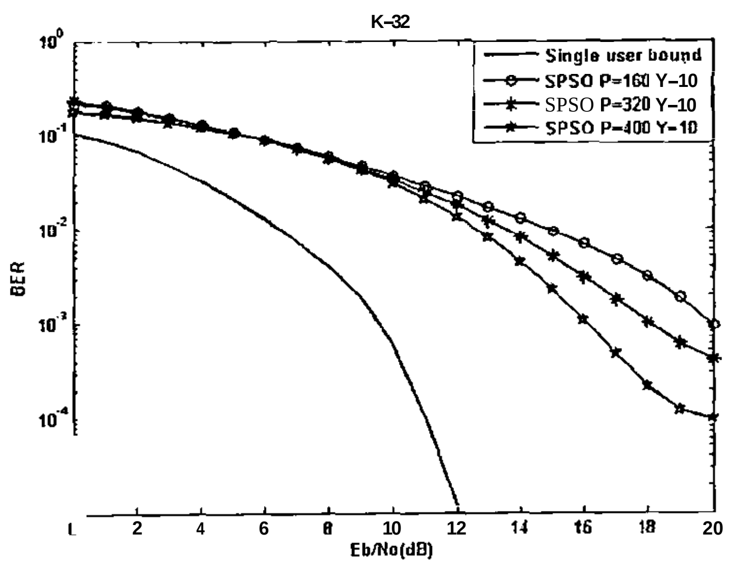


Fig. **3.24: BER** Performance of SPSO for Asynchronous MC-CDMA svstem a svstem with **32** users

CHAPTER 4

RADIAL BASIS FUNCTIONS AND PSO ASSISTED MULTIUSER DETECTION FOR DS-CDMA

4.1 Introduction

The two major factors which contribute to erroneous detection in a DS-CDMA system are the MAI and noise. The former appears due to non-orthogonal spreading codes while the latter is the induced by the channel. Since the traditional detectors give a soft output, the presence of MAI and channel noise make these soft decisions non-linearly separable. This chapter describes the use of Radial Basis Functions along with PSO for multiuser detection in a synchronous environment.

Section 4.2 sheds light on basics, brief history and applications of radial basis functions. Section 4.3 discusses the use of PSO-assisted multiuser detection using radial basis functions for synchronous DS-CDMA system. Section 4.4 discusses the simulation results. Section 4.5 concludes the chapter.

4.2 Background

Radial Basis functions (RBF) were initially used for solving real multivariate interpolation problems [279]. First of all it was Cover [280] who argued that a complex pattern classitication problem turns into linearly separable problem if it is taken to high dimensionality. This can be very easily understood by considering the XOR problem.

Another reason for having a high dimensional non-linear layer is the motive to increase the capacity of the network. RBF neural networks are known to be deterministic global non-linear optimization tool. The application areas of RBF networks include pattern classification [281]-[282], signal classification [283], function approximation [284]-[285], image restoration [286], time series prediction and modeling [287]-[289], and interference cancellation [290].

Fig. 4.1 shows the simplest form of a radial basis function network. It has three-layers, an input layer, one hidden layer and one output layer. The input layer obviously, connects the network to the environment. The hidden layer is a high dimensional layer that is non-linear in nature, while the output layer is linear.

There is a large class of radial basis functions. A few of these include multiquadrics. inverse multiquadrics and Gaussian functions. Multiquadrics are non-localized while Gaussian and inverse multiquadrics are localized radial basis functions. By localized we mean that these functions give a strong response near the centers and as the vector x moves away from the center, the response weakens.

In our work we have used Gaussian functipns as the activation function of the hidden layer units. Mathematically it is given by

$$\varphi_j(\mathbf{x}) = \exp\left[-\frac{\|\mathbf{x}-\mathbf{c}_j\|^2}{2\sigma_j^2}\right] \qquad j = 1,2,\cdots,N \qquad (4.2.1)$$

where $\mathbf{x} = [x_1, x_2, \cdots x_N]$ is the input vector.

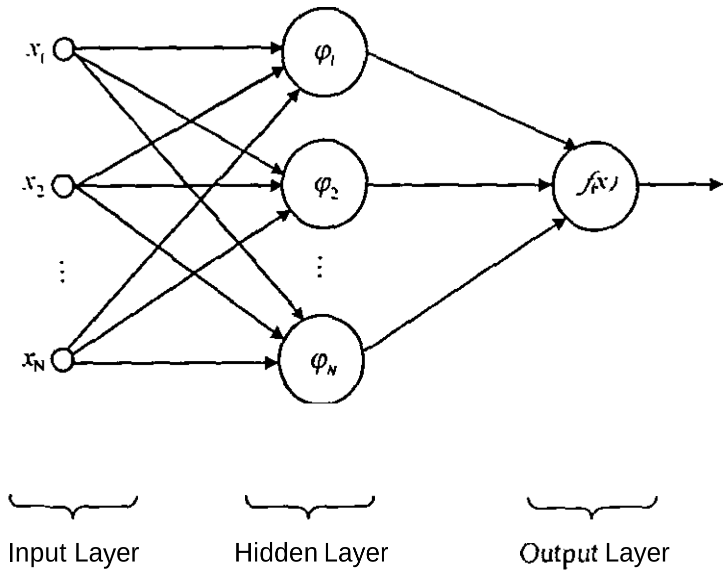


Fig.4.1: Radial basis function network

c_j is the centre and σ_j is the spread of j^{th} radial basis function. It is obvious from (4.2.1) that the argument of the activation function of each hidden unit in an RBF network computes Euclidean norm between the input vector and the center of that unit.

RBF have been trained using various learning schemes. These are orthogonal least squares method [291]-[292] and hybrid learning [293].

The use of Neural Networks for multiuser detection was first proposed by Aazhang [294]. Later on Radial basis functions for multiuser detection were proposed by Mitra [295]. Other contributions to the topic were provided by Cruickshank [296], Tanner [297]-[298], and Sessler [299]. The high computational complexity of RBF network has resulted in a lack of interest in its use. Some researchers have used other tools in conjunction with RBF network to get the better results at reduced complexity. Wei and

Hanzo [300] have used Genetic algorithms to reduce the computational complexity. We have proposed PSO in this dissertation, to cut down this computational cost. A number of other researchers [301]-[307] have also contributed for using the RBF in multiuser detection.

4.3 Problem formulation of RBF-assisted MUD for Synchronous DS-CDMA System

Consider the DS-CDMA system whose transmitter model is given in Fig.4.2. Each of the K users has been assigned a spreading code of length N. The transmitted composite signal of K users for this system is given as

$$\mathbf{S} = \sum_{k=1}^{K} \mathbf{s}^{(k)} = \sum_{k=1}^{K} A^{(k)} b^{(k)} g^{(k)} \tag{4.3.1}$$

where $A^{(k)}$ is the amplitude oftransmitted bit. $b^{(k)} \in \{1,-1\}$ and $\mathbf{g}^{(k)} = [g_1^k \; g_2^k \cdots g_N^k]$ is a spreading code for k^{th} user. with $g_i^{(k)} \in \{1,-1\}$. T_b and T_c are bit and chip intervals respectively, with $T_b = NT_c$. The received vector at the intput of the matched filter is given as

$$\mathbf{r} = \mathbf{S} + \mathbf{n} \tag{4.3.2}$$

Elements of vector $\mathbf{n}$ are independent and identically distributed (iid) random variables with Gaussian distribution. Channel is assumed to be lion-dispersive memoryless AWGN. The RBF assisted multiuser detector is given in Fig.4.3 with input vector as $\mathbf{r}$.

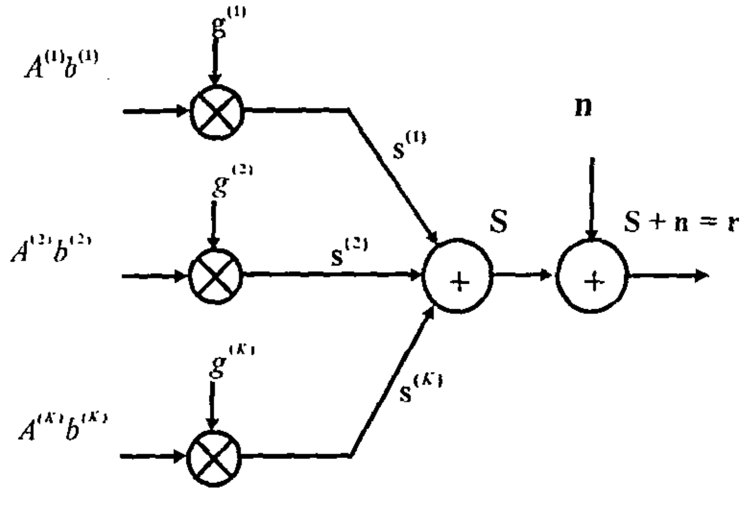


Fig.4.2: System Model of Transmitter

For the composite received signal r and vector centre c_j a typical RBF is given by

$$\varphi_j(\mathbf{r}) = \exp\left\{-\frac{\|\mathbf{r}-\mathbf{c}_j\|^2}{2\sigma_j^2}\right\} \quad j = 1,2,\dots,2^K \tag{4.3.3}$$

The basic radial basis function network has been expanded to give multiple outputs for the multiuser detector. There are a total of K outputs corresponding to K users in the system. The weighted outputs From RBF functions are added to give the k^{th} component vector.

$$\mathbf{y} = \left[y^{(1)}\ y^{(2)} \dots y^{(k)} \dots y^{(K)}\right]$$

$$y(k) = \sum_{j=1}^{M} w_{jk}\varphi_j(r) \tag{4.3.4}$$

$\cdot_{j,k}$ is the weight connecting j^{th} RBF to k^{th} summer and $M = 2'$. The above equation can be written in matrix form as,

$$\mathbf{y} = \mathbf{W\Phi} \tag{4.3.5}$$

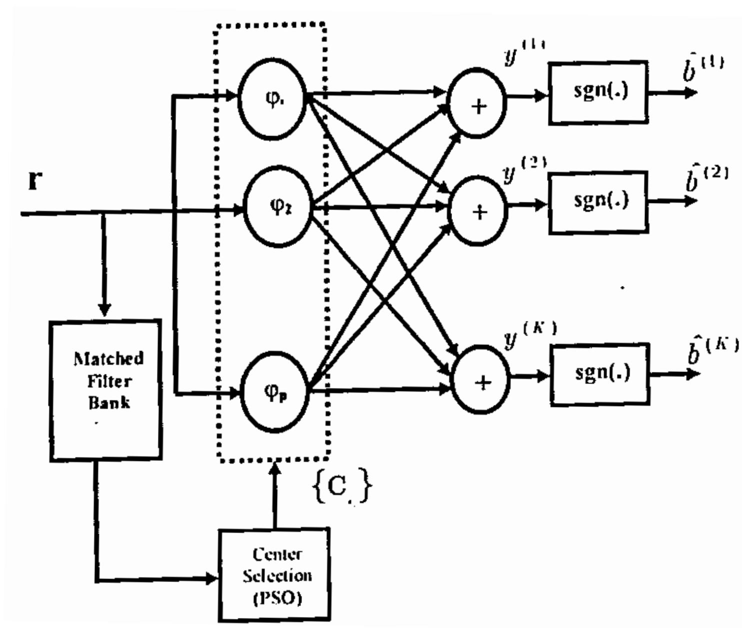


Fig. 4.3: RBF-PSO assisted receiver

The weights in RBF network, as given in [300], can belong to a limited set $w_{i,k} \in (1,-1)$. We can have majority decision rule if we choose the weights as follows:

$$w_{ik} = \begin{cases} 1 & , b_i^{(k)} = 1 \\ -1 & , b_i^{(k)} = -1 \end{cases} \tag{4.3.6}$$

For users with binary signaling, we can have 2^K possible combinations of bits. All of these can be used for formulating RBF centers

$$\mathbf{c}_i = \sum_{k=1}^{K} A^{(k)} b_i^{(k)} \mathbf{g}^{(k)} \quad \text{where } i = 1, 2, \cdots 2^K \tag{4.3.7}$$

where $c_i = \left[c_i^{(1)}\ c_i^{(2)}\ c_i^{(3)}, \cdots, c_i^{(N)}\right]$ can be called as a noiseless channel output corresponding to the transmitted K data bits $b_i = \left[b_i^{(1)}\ b_i^{(2)}\ b_i^{(3)}, \cdots, b_i^{(N)}\right]$. One may go for a straight forward approach by using all 2^K possible centers. but computational complexity for large K will be unbearable. To reduce this complexity. we have suggested PSO [147]. They are simpler to implement and faster in convergence as compared to CA.

4.3.1 HPSO for RBF-assisted MUD

The RBF centers are chosen so as to maximize the objective function as given by (4.3.3). This implies that specific sel of centers $\mathbf{c}_i$ that have components closest to vector $\mathbf{r}$. will be the channel output. In our proposed receiver the output of the bank of matched filters is followed by a hard decision. which is taken as input particle b_i for the PSO. Rest of the population is generated by flipping sign of a single position of b_i at random. For evaluating the fitness of all the particles (4.3.3) serves as fitness function. So the particles in the population serve as candidate RBF centers. After evaluating the fitness of all the centers, global best particle is chosen. As the PSO algorithm progresses the entire particles move with their respective velocities towards the global best particle. This movement or the convergence towards the global best particle is controlled by two factors. This can be seen in the following equation,

$$v_i^{(m)}(n) = v_i^{(m)}(n-1) + \varphi_1 \beta (p_i^{(m)} - b_i^{(m)}(n-1)) + \varphi_2 (1-\beta)(p_g^{(m)} - b_i^{(m)}(n-1)) \quad (4.3.8)$$

where $p_i^{(m)}$ denotes the personal best position, $b_i^{(m)}$ is the current position. and $p_g^{(m)}$ is the position of global best panicle. φ_1 and φ_2 are the stochastic acceleration terms. The second term on the right side of the above equation represents the individual intelligence of a particular particle, while the third term represents the collective or global intelligence. The contribution of the individual and global intelligence to the movement towards the global best can be finely controlled by using β. A value of 0.5 implies equal contribution of the individual and global intelligence. The velocity of a particle is then used to determine its nest position. This is done by using the following

$$if(rand(\ ) < S(v_i^{(m)}), \ then \ b_i^{(m)} = 1 \ \ else \ b_i^{(m)} = -1 \qquad (4.3.9)$$

After a finite number of PSO cycles, we get the best possible **RBF** centers. These centers contribute thmugh majority decision principle to give the estimated K user bits.

4.3.2 SPSO for RBF-Assisted MUD

The idea behind Soft PSO has already been esplained in the previous chapter. The application of soft PSO results in BER performance better than hard PSO.

4.3.3 Simulation Results and Discussion

The performance of PSO has been investigated by comparing it with other suboptimal techniques as well as with itself for different number of users and different computational complexities. The definition of computational complexity is the same as described in the previous chapter i.e. product of number of particles P and irerations Y. Throughout tlie simulations, the channel is assumed to be additive white Gaussian noise (AWGN) nature. For spreading the data bits, 31-chip Gold codes have been used.

Fig. 4.4 shows BER performance for a system with 10 users using both soft and hard PSO for computational complexities of 100 and 200. In both the cases, the performance of SPSO is superior to that of HPSO. Fig. 4.5 shows the comparison of GA-RBF [278] assisted MUD with the proposed HPSO-RBF assisted MUD [147] for computational complexities of 200 and 400. At low **SNR,** the performance of GA seems to be on better ride. However, as the SNR values reach in tlie range used in real world communication. i.e. greater than 10dB, PSO performs better. In this range of SNR i.e. $> 12\ dB$, it outperforms the CA based algorithm. Application of SPSO instead of HPSO has resulted in even better performance. which is quite obvious from Fig. 4.6. It shows BER performance for a system with 15 users for computational complexities of 100, 200 and 400.

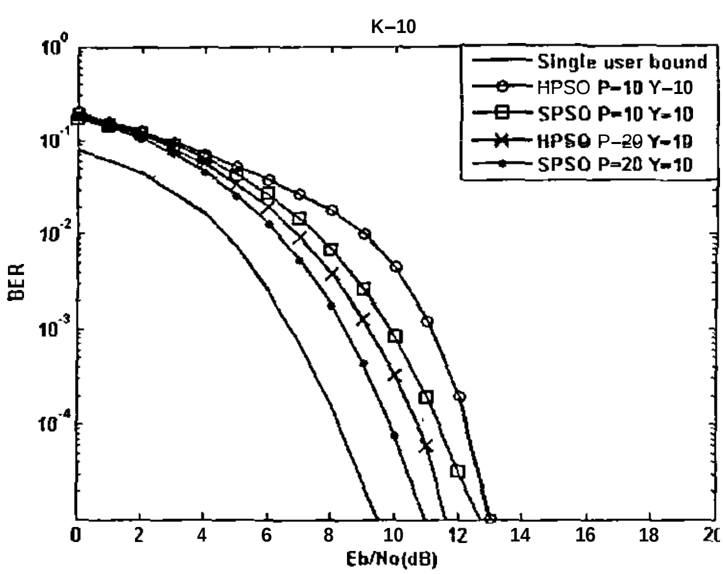


Fig. 4.4: BER Performance of PSO-RBF assisted MUD for

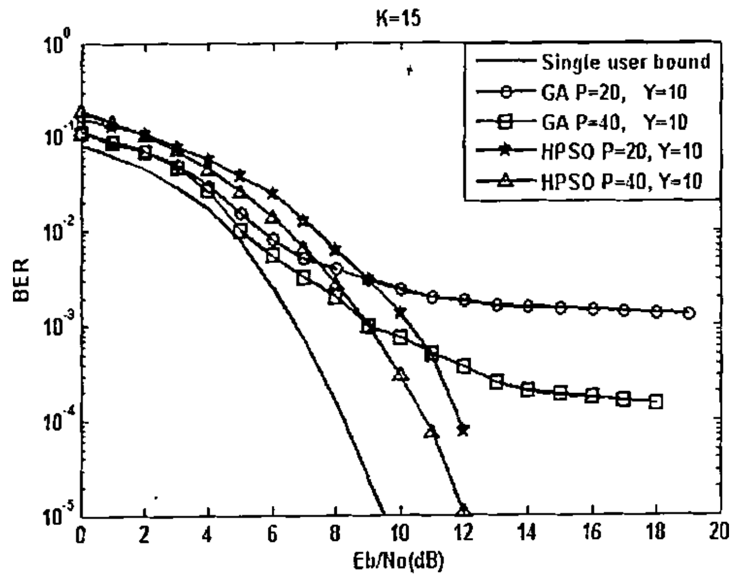


Fig. 4.5: Performance Comparison of PSO-RBF assisted MUD with GA assisted MUD system with 15 users

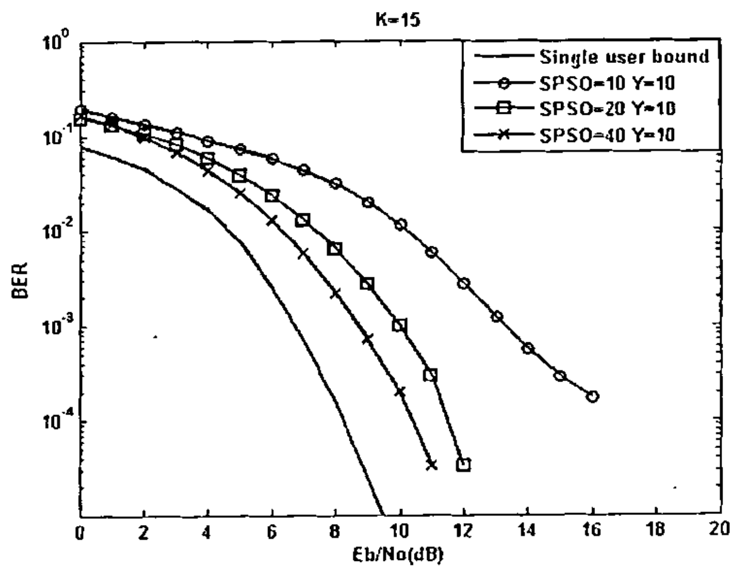


Fig. 4.6: BER Performance of SPSO-RBF assisted MUD for system with 15 users

Fig. 4.7 presents comparison of PSO with other suboptimal techniques including decorrelating detector, LMMSE detector and partial PIC detector for a system with 20 users. For PSO, we have used 100 particles and 50 iterations. As we can observe PSO for 20 users outperforms the other suboptimal techniques. Fig. 4.8 shows BER performance using SPSO for the same 20-user system. This simulation is based on computational complexities of 800, 1000 and 1200.

Fig. 4.9 shows simulation results for a system with 32 users using PSO-RBF approach. No simulation results were available for any other suboptimal technique for 32-user system. For this system, the performance of HPSO-MUD has been investigated with three complexities, each having two possible combinations. For example, the complexity 1000 has been tested with $P = 125$, $Y = 8$ and $P = 100$, $Y = 10$. It has been observed that having more number of iterations for the same computational complexity, results in better BER performance. The same observation is there for SPSO-RBF assisted MUD. as shown in Fig. 4.10.

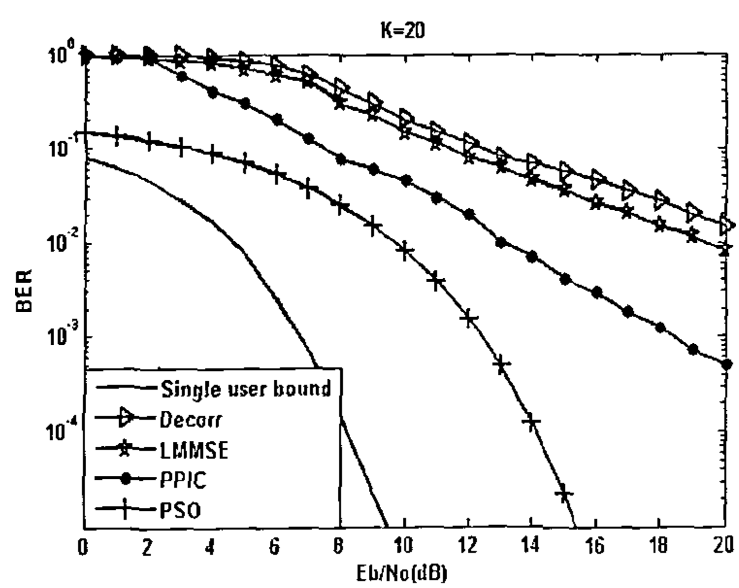


Fig. 4.7: Performance Comparison of RBF-PSO assisted MUD with different suboptimal techniques for system with 20 users

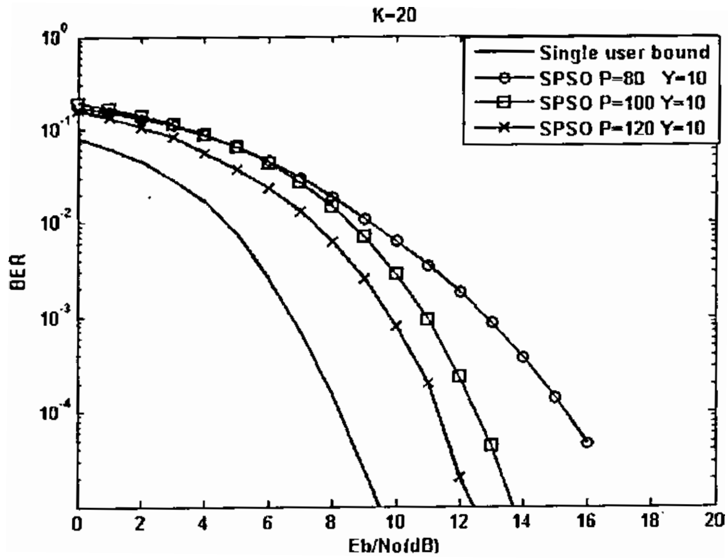


Fig. 4.8: BER Performance of SPSO-RBF assisted MUD for system with 20 users

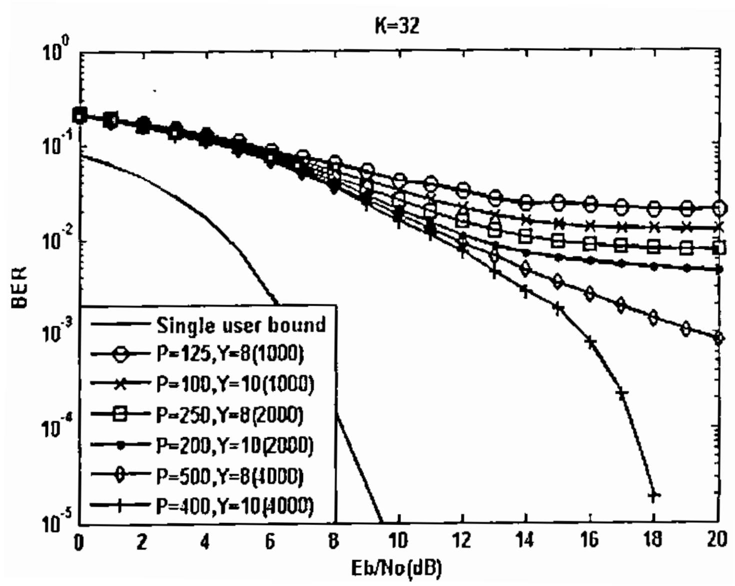


Fig. 4.9: Performance of HPSO-RBF assisted MUD for system with **32** users

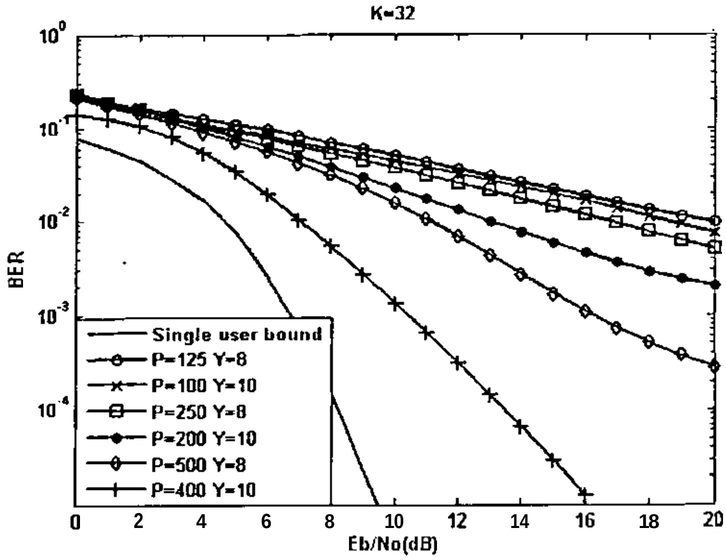


Fig. 4.10: Performance of **SPSO-RBF** assisted **MUD** for system **with** 32 users

CHAPTER 5

JOINT CHANNEL AND DATA ESTIMATION USING PARTICLE SWARM OPTIMIZATION

5.1 Introduction

This chapter investigates the use of PSO for joint channel and data estimation. Section 5.2 covers tlie background of the topic. Section **5.3** presents the system model and problem formulation. Section 5.4 explains the application of PSO for joint channel and data estimation. Section *5.5* discusses simulation results

5.2 Background

In real world communication the channel behavior is not completely known a priori. In other words the channel is not fixed and its behavior varies over time. Typical examples are the wireless channels or radio channels, underwater acoustic channels and the dial-up telephone network. In these kinds of environments one cannot design an optimum receiver. The time dispersive channel behavior results in intersymbol interference (ISI). A bulk of research work can be found that addresses tlie issue of fighting tlie ISI. The techniques used to fight tlie ISI are known as channel equalization tecliniques.

Broadly there are two families oftechniques for channel equalization. First major family relies on tlie use of training signals [308]. Prior to tlie actual data transmission. a predefined data sequence is transmitted and received. At the receiver the coefficients of the equalizer are adjusted accordingly. Once this is done, the actual data transmission begins. This simple technique cannot be used in many practical systems due to a number ofreasons including bandwidth and time wastage.

The second major family of equalization techniques which does not use training sequences is known as blind channel equalization. Blind channel equalization has been an active area of research. One can find a good review in [309]. Sato [310] is considered to be the pioneer of blind channel equalization. Blind equalization techniques can be divided in three categories. The algorithms in the first category are based on higher order statistics of the output data [311]-[315]. These algorithms were quite robust, but suffered from considerable storage requirement for large data samples and heavy computational load which has made their use difficult in fast changing environments like mobile communication. The bussgang algorithms [316]-[318] are also considered to be in this class. because they exploit the higher order statistics implicitly. These algorithms were relatively easy to implement but suffered from slow convergence.

The second category of blind equalization has algorithms is based on second order statistics of the output data [319]-[324]. Another category of blind channel equalization has algorithms that jointly use second and higher order statistics, called hybrid algorithms [325].

In the third major family of equalization joint data and channel estimation is done. One approach to this problem is to use **MLSE,** but it has huge computational complexity. A number of researchers have contributed towards lowering this computational cost. Ghosh [326] has used batch iterative method for the purpose, while Seshadri [327] has used blind maximum likelihood sequence estimation (MLSE) on Viterbi algorithm (VA) where VA operates on many trellises instead of one. Javad *et al* [328]-[331] have used

Kalman filtering to jointly estimate channel and data. Joint estimation of channel and data has also been investigated for multiple input multiple output (MIMO) [332], single input multiple output (SIMO) [333]-[335], OFDM [336]-[341] and Space time communication [342]-[344]. Chen [345] has used two level approach. At the upper level he has used micro-genetic algorithm to estimate the channel and at the lower level the Viterbi algorithm to estimate the data.

5.3 Problem Formulation

We assume that the channel to be estimated is a linear channel modeled by finite impulse response (FIR) filter having N real valued taps. Also the channel is assumed to be additive white Gaussian noise (AWGN) channel. Furthermore the data encoding is supposed to be M-ary PAM. The k^{th} received sample is given by,

$$r(k) = \sum_{i=0}^{N-1} h_i s(k-i) + n(k) \qquad (5.3.1)$$

where h is the channel impulse response and s is the transmitted symbol sequence and n is the noise. Also we assume that,

$$\sum_{i=0}^{N-1} h_i^2 = 1$$

For P received samples in one batch, in vector matrix form we have

$$\mathbf{r}_{1\times P} = \mathbf{a}_{1\times h}.\mathbf{S}_{h\times P} \qquad (5.3.2)$$

where P is the number ofsamples we received in one batch. The ML crirerion for jointly estimating the channel parameters and the data requires us to minimize the following metric.

$$J_{ML}(\hat{h},\hat{S}) = \sum_{i=1}^{P}\left|r(i) - \sum_{j=0}^{N} h_j S(i-j)\right|^2 \tag{5.3.3}$$

$$J_{ML}(\hat{h},\hat{S}) = \sum_{i=1}^{P}\left(r^2(i) - 2r(i)\sum_{j=0}^{N-1} h_j S(i-j) + \sum_{j=0}^{N-1} h_j S(i-j)\sum_{k=0}^{N-1} h_k S(i-k)\right)$$

$$= \sum_{i=1}^{P} r^2(i) - 2\sum_{i=1}^{P} r(i)\sum_{j=0}^{N-1} h_j S(i-j) + \sum_{i=0}^{P}\left(\sum_{j=0}^{N-1} h_j S(i-j)\sum_{k=0}^{N-1} h_k S(i-k)\right)$$

where the first term can be ignored. Equivalently the following metric can be maximized to obtain an estimate of the channel parameters and the data,

$$C_{ML}(\hat{h},\hat{S}) = 2\sum_{i=1}^{P} r(i)\sum_{j=0}^{N-1} h_j S(i-j) - \sum_{i=0}^{P}\sum_{j=0}^{N-1}\sum_{k=0}^{N-1} h_j h_k S(i-j)S(i-k)$$

In vector matrix form,

$$C_{ML}(\hat{\mathbf{h}},\hat{\mathbf{S}}) = 2\mathbf{r}^T\mathbf{S}\mathbf{h} - \mathbf{h}^T\mathbf{S}^T\mathbf{S}\mathbf{h}$$

$$= 2\mathbf{r}.(\mathbf{S}\mathbf{h}) - \|\mathbf{S}\mathbf{h}\|^2 \tag{5.3.4}$$

The equation (5.3.4) serves as the cost function for the particle swarm optimization algorithm.

5.4 PSO for Joint Channel and Data Estimation

There are two basic versions of PSO algorithm. One is for binary data and the other is for real valued data. We have used the real-valued PSO algorithm for the estimation of channel parameters and binary PSO for detecting the data. The fast convergence properties of PSO and its successful application to various problems in engineering make it a potential tool for reducing the computational complexity in joint estimation problem. The PSO algorithm in our work has been applied at two levels iteratively. At the top level channel parameters are estimated by using real valued PSO. Once an estimate is obtained it is used at the lower level together with the binary valued PSO to estimate the data. The complete algorithm has been explained in the following.

Step 1: Two separate populations/swarms are initialized at random values, one for data b called data swarm and the other for channel parameters h. known as channel swarm. The dimensions of b and h depend on the size of the channel output sample taken. and the number of channel parameters respectively. All the combinations of channel swarm and data swarm are tested using (5.3.4) as fitness function. The best combination of b and h is chosen. A new population for b is created by perturbing the b chosen above. This population serves as the data swarm for the binary PSO algorithm for detecting the data bits. Similarly the vector h chosen above is perturbed to create channel swarm for real-valued PSO for estimating the channel parameters. Again the best combination of b and h is chosen and is called global best data vector and global best channel vector.

Step 2: Real-valued or continuous PSO starts with the population created above. Each particle in the channel swarm is a candidate solution for channel parameters. Again (5.3.4) is used as fitness function, but this time b is fixed at the global best data vector and PSO algorithm works on the channel swarm. The velocity of the m^{th} location of i^{th} particle is updated using the following,

$$v_m^i(n) = v_m^i(n-1) + 2\varphi_1(h_m^p - h_m^i) + 2\varphi_2(h_m^g - h_m^i) \qquad (5.4.1)$$

where h^p denotes the personal best and h^g denotes the global best panicle. Particle position is updated by using the following,

$$h_m^i(n) = h_m^i(n) + \tanh(v_m^i(n)) \qquad (5.4.2)$$

This process is repeated till a fixed number of iterations have been executed. At the end of the iterations, best particle is chosen as the estimated solution for channel parameters.

Step 3: The next step is to use the soft version of binary PSO, called SPSO, for estimating the data bits. Again we use the (5.3.4) as fitness function. This time, the channel vector estimated above through PSO is fixed and the binary PSO algorithm works on the data swarm. The global best particle of the data swarm is chosen and the velocity is updated wing the following,

$$u_m^i(n) = u_m^i(n-1) + 2\varphi_1(b_m^p - b_m^i) + 2\varphi_2(b_m^g - b_m^i) \qquad (5.4.3)$$

where b_m^i denotes the m^{th} location of i^{th} particle. Particle position is updated. using the following,

$$if(rand(\,) > S(u_{im})),\quad then\quad b_{im} = -1 + S(u_{im})\quad else\quad b_{im} = S(u_{im}) \qquad (5.4.4)$$

r

where $$S(u_{mn}) = (1/1 + \exp(-\gamma u_{mn}))$$

After a tised number of iterations for the data swarm, the algorithm stops. The global best particle from the data swarm is chosen as the estimated data vector.

Step 4: Next sample of the received signal is taken and the execution goes to the step 2

5.5 Results and Discussion

We have used three different channels to test our algorithm. The impulse response of these channels is:

Channel 1: $\mathbf{h}_1 = [0.407 \quad 0.815 \quad 0.407]$

Channel 2: $\mathbf{h}_2 = [-0.21 \quad -0.50 \quad 0.7\% \quad 0.36 \quad 0.21]$

Channel 3: $\mathbf{h}_3 = [0.227 \quad 0.46 \quad 0.688 \quad 0.46 \quad 0.227]$

The channels are taken from [345]. We have used two of the most widely used criterion to judge the performance of PSO algorithm for channel and data estimation. For joint estimation we have used mean square error (MSE), detined by,

$$\mathrm{MSE} = \frac{1}{P}\sum_{i=0}^{P}\left(r(i) - \sum_{j=0}^{N-1}\hat{h}_j\hat{s}(i-j)\right)^2 \qquad (5.5.1)$$

Specifically for channel parameters we have used mean tap error (MTE), defined by.

$$\mathrm{MTE} = \left\|\pm\hat{\mathbf{h}} - \mathbf{h}\right\|^2 \qquad (5.5.2)$$

As explained above, we have applied continuous PSO for channel coefficients and binary PSO for estimating data the bits. in a repetitive manner. A big loop repeats the two PSO algorithms one after the other. If the outer loop iterates for (say) 10 times and both the inner loops iterate for five times each. the total number of iterations will be 250.

Fig.5.1 shows the performance comparison of PSO and GAVA [345] technique in terms of MTE for channel I. We have used the data sample of size 50. The number of particles for the channel swarm is 15 and for data swarm is **100.** The number of cycles for both the PSO algorithms is 5. The graph shows the results for two different SNR values. It is obvious that for higher SNR, i.e. 30dB, the PSO takes over GAVA even for less number of iterations.

Fig, 5.2 shows the performance comparison of PSO and GAVA technique in terms of MTE for channel 2 which is five tap channel. All other parameters for this simulation are the same as used for channel I above. One can observe that PSO performs much better as the number of iterations is increased. Similarly fig. 5.3 shows simulation results for channel 3

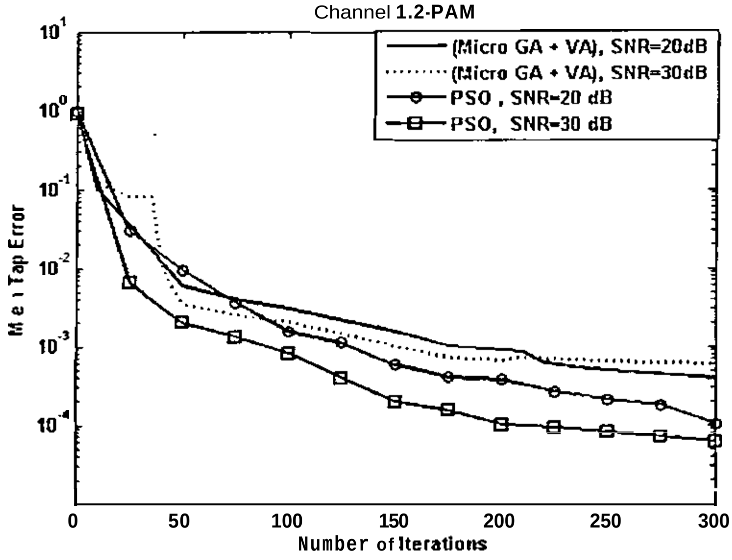


Fig. 5.1: Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel I for 2 PAM.

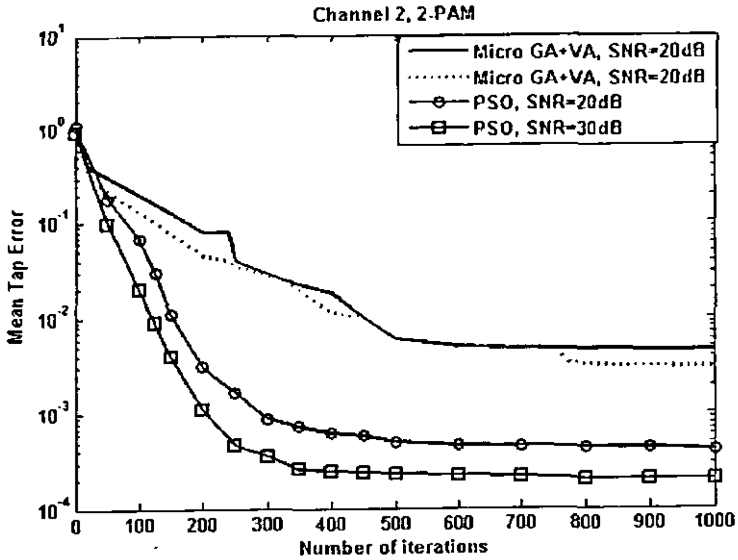


Fig. 5.2: Performance comparison of PSO and GAVA technique in term; of MTE against the number of iterations for channel 2 for 2 PAM.

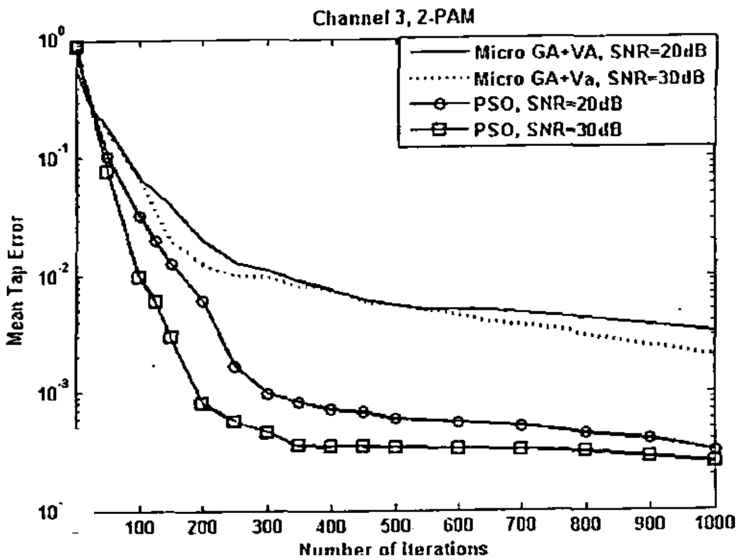


Fig. 5.3: Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 3 for 2 PAM.

Yet another way to investigate the performance of the proposed scheme is the MSE. Since no results were' available for comparison with **GAVA** for MSE (defined as a function of number of iterations). hence only the performance of PSO is given in the following results. Fig. 5.4 shows the MSE for channel **I** at three different SNR levels, i.e. 10, 20 and 30 dB. Fig. 5.5 shows MSE for channel 2 and Fig. 5.6 shows MSE for channel 3 against number of iterations. The parameter used is SNR whose values are kept 10. 20 and 30 dB respectively. For these simulations, we have used the sample size of 50. The number of particles for the channel swarm is 10 and for data swarm is 100. The number of cycles for both the PSO algorithms is again taken to be ***5.*** The same task has been performed for channel 3 by keeping the simulation parameters the same. It is important to note that as the number of iterations is increased, MSE drops quite fast. Since no results were available for comparison with **GAVA** for MSE, hence only the performance of PSO is given in the following results.

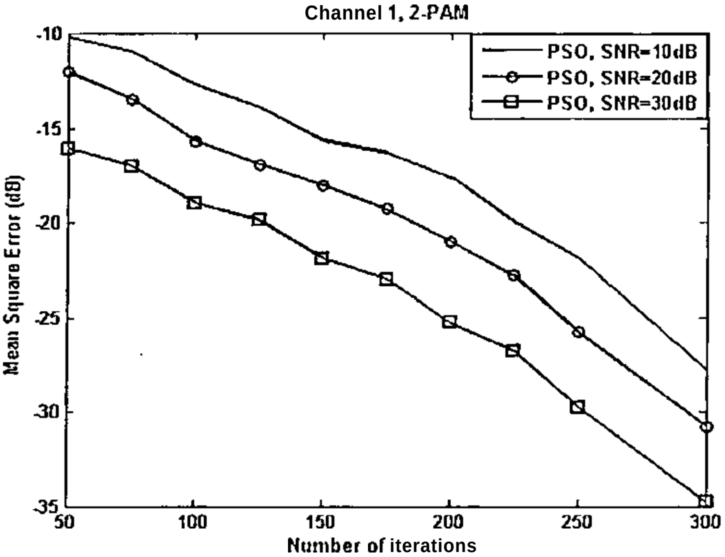


Fig. 5.4: Performance of PSO in terms of MSE against the number of iterations for channel **I**

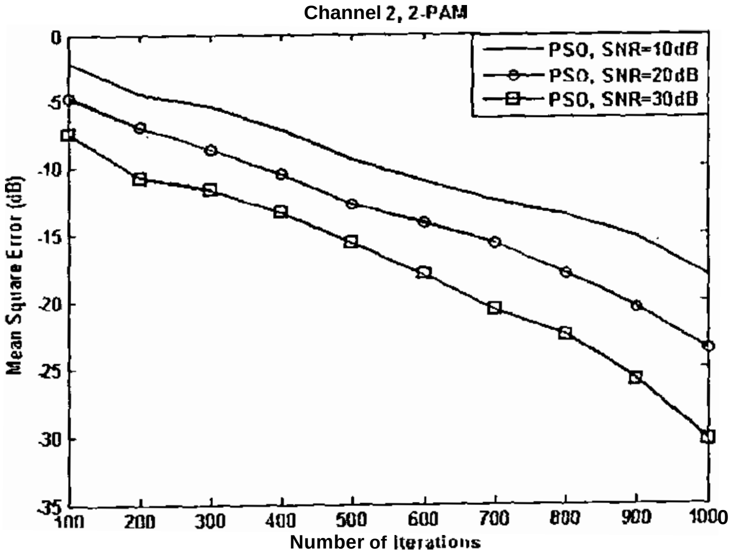


Fig. 5.5: Performance of PSO in terms of MSE against the number of iterations for channel **2**

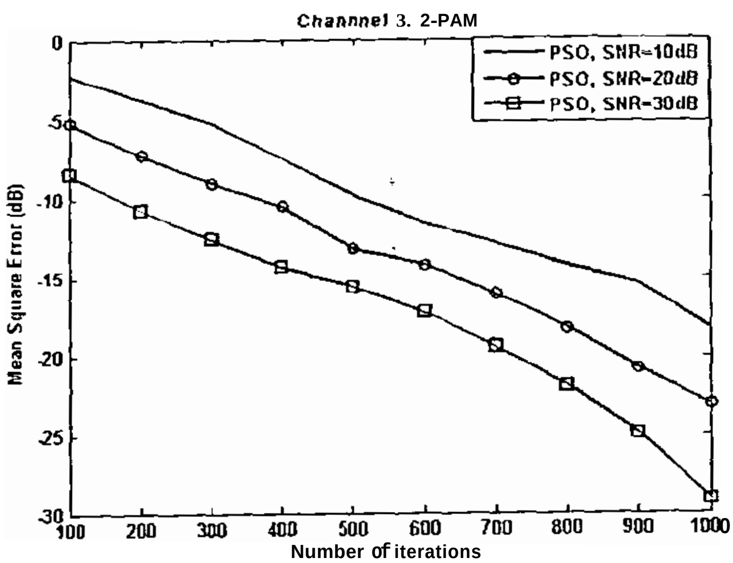


Fig. 5.6: Performance of PSO in terms of MSE against the number of iterations for channel 3

In order to thoroughly test the performance of PSO, we have also investigated the MTE for all the above mentioned channels for 8-PAM. These results have also been compared with the results given in [345], where GAVA technique has been used. All other simulation parameters are the same as used previously.

The comparison of the GAVA and PSO approach from computational complexity point of view is quite complicated since the former approach involves two different algorithms. one GA and the other VA. The computational complexity of this approach as given in [345] is $N_{VA} \times C_{VA}$ where N_{VA} is the number of Viterbi algorithm (VA) calls and C_{VA} is the complexity of VA. We have used product of the number of runs of both the PSO algorithms as complexity. However for the sake of comparison, in every simulation we have used the same sample size, as used in [345].

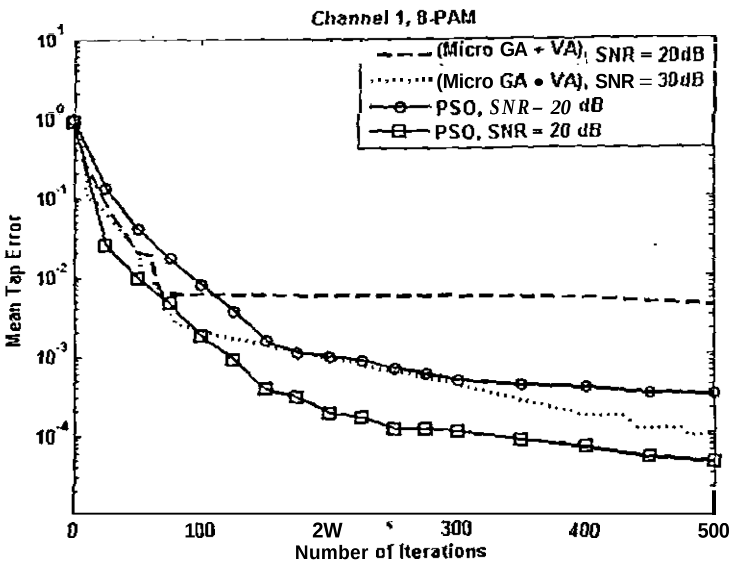


Fig. 5.7: Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 1 for 8 PAM.

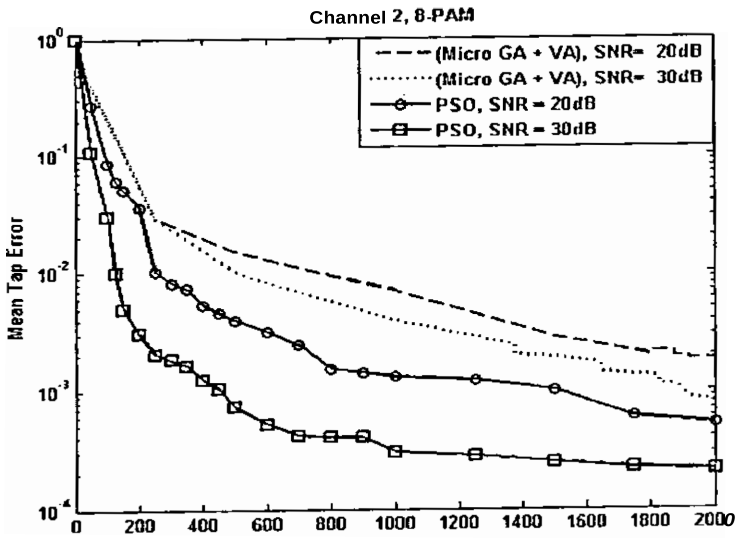


Fig. 5.8: **Performance** comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel 2 for 8 PAM.

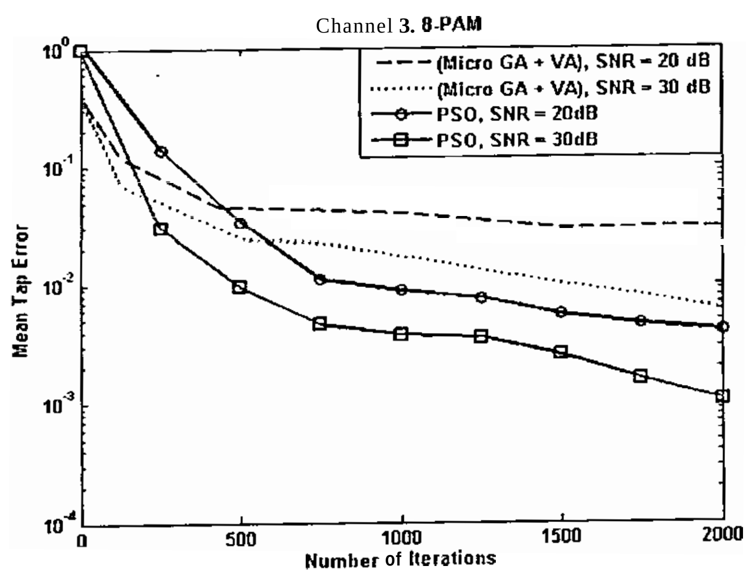


Fig. 5.9: Performance comparison of PSO and GAVA technique in terms of MTE against the number of iterations for channel **3** for 8 PAM.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Conclusion

This dissertation is concerned with the application of PSO to different problems in the field of communication. There exist various algorithms with different computational costs and implementation complexities to optimize or solve the real world problems. The winner is the one that does the job with minimum computational complexity and cost. A number of evolutionary algorithms have been applied to various problems in communication. In this dissertation the PSO has been applied to four different problems. The reasons behind choosing PSO are manifold. Firstly, PSO is relatively a new technique. Secondly, it is perhaps the most widely and critically acclaimed evolutionary algorithm and is currently under investigations. One can see a large number of variants proposed by a number of researchers. Thirdly PSO is very easy to implement and requires quite less number of parameters and at the same time it is characterized by an excellent and fast convergence.

Standard binary PSO (HPSO) and a proposed variant namely SPSO has been employed on four different problems. In HPSO the particles take the discrete values while for SPSO, the particle values are real numbers. The flexibility in values is mainly the reason for terminology of soft PSO. The first application of PSO in this dissertation is MUD of multicarrier CDMA system in synchronous as well as in asynchronous environment.

There are two major issues, addressed in this dissertation, for designing a receiver in CDMA systems. The first one is related to multiple access interference. The optimal approach to multiuser detection requires computational cost that grows exponentially with the number of users. Such a large number of calculations cannot be performed in real time applications. Among many suboptimal approaches, evolutionary computing is one solution. PSO has teen successfully employed for MUD and shown to outperform GA, which is one of the best known evolutionary computing techniques. For the same computational complexity, PSO-MUD has outperformed the GA-MUD. Considering the uplink, PSO has been employed for MUD in asynchronous MC-CDMA system and has shown promising results.

Another problem addressed in this dissertation is the PSO assisted MUD along with RBF. Here the MUD problem has been treated as pattern classification problem. Thus a linearly non separable problem has been recast into a separable problem by transforming it to higher dimensions using RBF. PSO is used to reduce the computational cost and the results show the superiority of PSO over GA and other suboptimal techniques.

The fourth problem is the joint data and channel and data estimation. Computationally, this is again a very expensive problem. PSO offers a suboptimal solution. The joint application of continuous and discrete PSO has given a performance, better than the other two approaches, namely, micro GA and blind trellis search technique.

6.2 Future Work

Our future work is in three dimensions. First, the application of PSO to a number of other areas like, channel coding, OFDM, SIMO and MIMO channels etc. Second, is to develop and propose other variants so as to ensure faster convergence without getting stuck in local minima. Third, is to apply these variants to the problems.

Other than communication, the other signal processing areas could be a potential hanle grounds for PSO. For example one may apply PSO in the field of image processing and pattern classification. For this purpose radial basis functions, other than those, discussed in this dissertation. may be used. It will need a special variant tailored in accordance with the need of environment.

Similarly PSO may be useful in the problem of speech coding and recognition. Source coding could be another juicy area for the application of PSO. Another very interesting area might be artificial intelligence and machine learning. The ever increasing computing power is going to put a pressure on researchers to develop more and more intelligent machines. With enormous computing capability at hand the race will be among those who develop smarter learning algorithms and hence learning machines. Since the PSO is based on social thinking, its role may be investigated in an environment where machines jointly learn from the environment. Another advantage is that evolutionary computing algorithms in general and PSO in particular are well suited for parallel machines. This may also be joined with GA to give some more robust learning algorithm. Another dimension could be the use of PSO to train various types of artificial neural networks for developing learning algorithms.

REFERENCES

[1] S. Verdu. "Multiuser Detection." *Cambridge Press* 1998.

[2] E. Polak, "Optimization: Algorithms and Consistent Approximations." ***Springer Verlag, New York, USA*** 1997.

[3] J. Spall, "Introduction to Stochastic Search and Optimization," ***first edition. Wiley-Interscience***, 2003.

[4] X. Yao, "An overview of evolutionary computarion," *Chinese Journal of Advanced Software Research*, vol. 3, no. 1, pp. 12-29. 1996.

[5] T. Back. D. B. Fogel, Z. Michalewicz. "Evolutionary Computation." CRC Press. 2000.

[6] L. J. Fogel, A. J. Owens. M. J. Walsh, "Artificial Intelligence through Simulated Evolution," ***John Wiley***, 1966.

[7] L. Shi and G. Xu., "Self-adaptive evolutionary programming and its application to multi-objective optimal operation of power systems", ***International Journal of Electric Power Systems Research***, vol. 57, no.3. pp. 181-187. April 2001.

[8] L. L. Lai. J. T. Ma. "Application of evolutionary programming to transient and subtransient parameter estimation." *IEEE Transactions on Energy Conversion,* vol. 11, pp. 523-530. September 1996.

[9] J. Graf and W. Banzhaf, "Interactive evolution of images". ***Proceedings of the Fourth Annual Conference on Evolutionary Programming***, editor D. B. Fogel, 1995.

[10] Coates. Paul. "Using Genetic Programming and L-Systems to Explore 3D Design Worlds." *CAAD Futures '97, Kluwer Academic Publishers. Munich*, 1997.

[11] I. Rechenberg, "Cybernetic Solution Path of an Experimental Problem." *Library translation* pp. 1122, Royal Aircraft Establishment, Farnborough. UK. 1965.

[12] I. Rechenberg. Evolution strategie, "Optimierung technischer Systeme nach Prinzipien der Biologischen Evolution," *Frommann-Holzboog, Stuttgart*, 1973.

[13] H-P. Schwefel, "*Evolutionsstrategie und numerische Optimierung*," Ph.D. thesis, TU Berlin, 1975.

[14] T. Hatanaka, K. Uosaki, H. Tanaka, Y. Yamada. "System parameter estimation by evolutionary strategy," *Proceedings of the 35th SICE Annual Conference*, pp. 1045 – 1048. 1996.

[15] J. Louchet, "Stereo analysis using individual evolution strategy", *Proceedings of 15th International Conference on* Pattern *Recognition* . vol. I . pp. 908 –911, 2000.

[16] G. W. Greenwood, A. Gupta, K. McSweeney, "Scheduling tasks in multiprocessor systems using evolutionary strategies," Proceedings *of the* First IEEE Conference *on Evolutionary Computation*, vol.1, pp345 –349, 1994.

[17] H. De Garis, "Genetic programming: modular evolution for Darwin machines." *Proceedings of the International Conference* on *Neural Networks*, pp.194-197. Washington. DC: Lawrence Erlbaum, 1990.

[18] J. R. Koza. "Evolving a computer program to generate random numbers using the genetic programming paradigm," ***Proceedings of the Fourth International Conference on Genetic Algorithms,*** pp. 37-44. La Joolla, CA: Morgan Kaufmann.

[19] J. Koza. "Genetic Programming: on the Programming of Computers by Means of Natural Selection and Genetics." ***MIT Press. Cambridge, MA, USA***, 1992.

[20] J. R. Koza. "Genetic Programming II: Automatic Discovery of Reusable Programs." ***The MIT Press.*** 1994.

[21] J. R. Koza. H. B. Bennett. D. Andre and M. A. Keane, "Genetic Programming III: Darwinian Invention and Problem Solving," ***Morgon Kaufmann Publishers***, 1999

[22] H. J. Bremermann, 'The evolution of intelligence. The nervous system as a model of its environment," ***Technical report, no.1, contract no. 477(17), Department of Mathematics. University Washington, Seattle***, July, 1958.

[23] J.H. Holland. "Adaptation in Natural and Artificial Systems." ***MIT Press, second edition.*** 1992.

[24] J. Holland. "'Adaptation in Natural and Artificial Systems." ***University of Michigan Press***, Ann Arbor. MI. 1975.

[25] L. Caponetti. N. Abbatista, G. Carapella. "A genetic approach to edge detection." ***Proceedings of IEEE international conference of image processing,*** vol.1, pp. 318 –322,1994.

[26] R. Lukac. K.N. Plataniotis, B. Smolka. A.N. Vene-tsanopulos. "Color image filtering and enhancement based on genetic algorithms." ***Proceedings of the inter-***

national symposium on circuits and systems, ISCAS '04, vol.3, pp. 913 – 916. 2004.

[27] G. L. Bosco, "A genetic algorithm for image segmentation," *Proceedings of 11th international conference on image analysis and processing*, pp.262-266, 2001.

[28] D. Marcano, "Synthesis of Linear and Planar Antenna Arrays Using Genetic Algorithms," *IEEE International Symposium on Antennas and Propagation*, vol. 3, pp. 1688-1691, July 1997.

[29] S.J. Flockton and M.S. White. "The application of genetic algorithms to infinite impulse response adaptive filters." *IEE Colloquium Digest 1993/039*, pages 9/1-9/4. IEE. 1993.

[30] J. Kennedy and R. C. Eberhart, "Particle swarm optimization." *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942–1948, Piscataway. NJ, USA, 1995. IEEE Press.

[31] R. C. Eberhart, P. Simpson, and R. Dobbins, "*Computational Intelligence PC Tools*," chapter 6. pp. 212-226. Academic Press Professional. 1996.

[32] J. Kennedy. "Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance." *Proceedings of IEEE Congress on Evolutionary Computing*, vol. 3, pp. 1931–1938, July 1999.

[33] J. Kennedy and R. Mendes. "Population structure and particle swarm performance," *Proceedings of IEEE Congress on Evolutionary Computing*, vol. 2, pp. 1671–1676, May 2002.

[34] H. Fan and Y. Slii. "Study on Vmas of particle swarm optimization." *Proceedings of Workshop on Particle Swarm Optimization*, Purdue School of Engineering and Technology. April 2001.

[35] J. Kennedy and R.C. Eberharr, "A Discrete Binary version of the Particle Swarm Algorithm." ***Proceedings of conference on Systems, Man, and Cybernetics***, pp.4101-4109, 1997.

[36] A. Abido, "Particle swarm optimization for multimachine power system stabilizer design," ***Proceedings of Power Engineering Society Summer Meeting (PES)***, vol. 3, pp. 1346–1351, July 2001.

[37] A. Abido. "Optimal power flow using particle swarm optimization." *International Journal of Electrical Power and Energy Systems*, vol. 24, no. 7. pp. 563–571, Oct. 2002.

[38] Y. Slii and C. Eberhart, "A Modified Particle Swarm Optimizer," ***Proceedings of IEEE International Conference of Evolutionary Commutation***, pp.69-73. May 1998,

[39] Y. Shi and R. Eberhart, "Parameter selection in particle swarm optimization," ***Proceedings of 7th International Conference on Evolutionary Program***, pp. 591–600, May 1998.

[40] Y. Slii and R. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings* of ***IEEE Congress on Evolutionary Computation***, vol.3 pp.-1950. July 1999.

[41] M. Clerc, "The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization," ***Proceedings of the Congress on Evolutionary Computation***, vol. 3, pp. 1951-1957, July 1999.

[42] M. Clerc and J. Kennedy. "The Particle Swarm: Explosion. Stability and Convergence in a Multi-Dimensional Complex Space," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp.58-73, February 2002.

[43] R. C. Eberhart and Y. Shi. "Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization," ***Proceedings of IEEE Congress on Evolutionary Computation***, vol.1, pp.84-88, July 2000.

[44] E. Ozcan and C. Mohan, "Particle swarm optimization: Surfing the waves," ***Proceedings of IEEE Congress on Evolutionary Computation***, vol. 3. pp. 1939–1944. July 1999.

[45] P. N. Sunganthan, "Particle Swarm Optimization with Neighborhood Operator." *Proceedings of the **Congress on** Evolutionary Computation*, pp.1958-1962, July 1999.

[46] R. Storn and K. Price, "Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," *Journal of **Global Optimization***, vol. 11, pp. 341 – 359. December 1997.

[47] W. Zhang and X. Xie, "DEPSO: Hybrid particle swarm with differential evolution operator." *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, vol. 4. pp. 3816–3821, October 1003.

[48] H. Talbi and M. Batouche, "Hybrid particle swarm with differential evolution for multimodal image registration," Proceedings of *IEEE International Conference Industrial Technology*, vol. 3, pp. 1567–1572, December. 2004.

[49] P. Moore and G. Venayagamoorthy, "Evolving combinational logic circuits using particle swarm, differential evolution and hybrid DEPSO, " *International* Journal *of Neural Systems*, vol. 16. no. 2, pp. 163–177. 1006.

[50] P. J. Angeline, "Using Selection to Improve Particle Swarm Optimization." *Proceedings of International Conference on Evolutionary Computation*, pp. 84-89, May 1998.

[51] V. Miranda and N. Fonseca, "EPSO evolutionary particle swarm optimization, a new algorithm with applications in power systems." *IEEE/PES* Conference and Exhibition on Transmission and *distribution*, Asia *Pacific*, vol. 2. pp. 745–750, October 2002.

[52] V. Miranda and N. Fonseca. "New Evolutionary Particle Swarm Algorithm (EPSO) Applied to Voltage/Var Control." *Proceedings of PSCC'02- Power Systems Computation Conference*, June. 2002.

[53] M. Lovbjerg, T. K. Rasmunssen, and T. Krink, "Hybrid Particle Swarm Optimizer with Breeding and Subpopulations." Proceedings *of the Genetic and* Evolutionary Computation Conference, San Francisco, USA. July 2001.

[54] A. El-Dib, H. Youssef, M. El-Metwally, and Z. Osman, "Load flow solution using hybrid particle swarm optimization." Proceedings *of IEEE International*

Conference on Electrical, Electronic, Computer Engineering, pp. 742–746. September 2004.

[55] J. Robinson, S. Sinton and Y. Rahmat Samii. "Particle swarm. genetic algorithm. and their hybrids: optimization of a profiled corrugated horn antenna." *Proceedings of IEEE international symposium on antennas and propagation* vol.2, pp. 314–317.2002.

[56] R. Poli, C. Di Chio, and W. B. Langdon, "Exploring extended particle swarms: a genetic programming approach," *Proceedings of the 2005 conference on genetic and evolutionary computation*, pp. 169–176. 3005.

[57] R. Poli, W. B. Langdon. and O. Holland. "Extending panicle swarm optimization via genetic programming." *Proceedings of the 8th European conference on genetic programming*, vol. 3447, pp. 291–300, 1005.

[58] J. D. Schaffner, "*Multiple Objective Optimization with Vector Evaluated Generic Algorithms*", Ph.D. thesis, Vanderbilt University, Nashville. Tennessee.

[59] J. D. Schaffner, "Multiple objective optimization with vector evaluated genetic Algorithms," *Proceedings of on International Conference on Genetic Algorithms and Their Applications*. pp. 93-100. 1985.

[60] C. Coello. "A comprehensive survey of evolutionary-based niultiobjective optimization," *Knowledge and Information Systems*. vol. I. no. 3. pp. 129-156, 1999.

[61] C. M. Fonseca and P. J. Fleming. "An overview of evolutionary algorithms in multiobjective optimization." *International Journal of Evolutionary Computation*, vol.3, pp. 1-16. 1995.

[62] K. Parsopoulos and M. Vrahatis, "Particle swarm optimization method in multiobjective problems," *Proceedings of ACM Symposium on Applied Computing*, pp. 603-607, March 2002.

[63] Hu, X. and Eberhart, R. C., "Multiobjective optimization using dynamic neighborhood panicle swarm optimization." Proceedings of IEEE Congress *on Evolutionary Computation*, pp. 1677-1681.2002.

[64] T. Ray and K. Liew, "A swarm metaphor for multi-objective design optimization." *Engineering Optimization*, vol. 34, no. 2, pp. 141–153. March 2002.

[65] C. Coello and M. Lechuga. Mopso, "A proposal for multiple objective particle swarm optimization." *Proceedings of Congress* on *Evolutionary Computation*, pp. 1051-1056.2002.

[66] C. Coello. G. Pulido, and M. Lechuga. "Handling multiple objectives with particle swarm optimization," IEEE *Transactions* on *Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, June 2004.

[67] S. Mostaghim and J. Teich, "Strategies for finding local guides in multi-objective particle swarm optimization." *Proceedings of IEEE Swarm Intelligence Symposium*, pp. 2633, 2003.

[68] X. Hu and R. Eberhart, "Multiobjective optimization using dynamic neighborhood particle swarm optimization," *Proceedings of Congress on Evolutionary Computation*, vol. 2, pp. 1677-1681, May 2002.

[69] K. Parsopoulos and M. Vrahatis, "Recent approaches to global optimization problems through particle swarm optimization." *Natural Computing*, vol. 1, pp. 235-306, May 2002.

[70] J. J. Liang, A. K. Qin, Ponnuthurai Nagaratnam, S. Baskar, "Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions." *IEEE Transactions on Evolutionary Computation*, vol. 10. no.3 pp.281-295, Jun 2006.

[71] K. E. Parsopoulos, and *hl.* N. Vrahatis, "Particle swarm optimizer in noisy and continuously changing environments," *Artificial intelligence* and *soft computing*, IASTED/ACTA Press, Anaheim, CA. USA, pp. 289-294. 2001.

[72] R. C. Eberhart and Y. Shi, '-Tracking and optimizing dynamic system with particle swarms," *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 94-97, 2001.

[73] Hu, X., and Ebcrhart. R. C., "Adaptive particle swarm optimization: detection and response to dynamic systems." *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1666-1670. 2002.

[74] A. Carlisle and G. Dozier, "Adapting particle swarm optimization to dynamic environments." *Proceedings of International* Conference on *Artificial Intelligence* pp. 429-434, 1000.

[75] A. Carlisle and G. Dozier, "Tracking changing extrema with particle swarm optimizer." *Proceedings of World Automation Congress*, pp. 265-270, 2002.

[76] T. Blackwell and P. J. Bentley. "Don't push me! Collision-avoiding swarms." *Proceedings of the IEEE congress on evolutionary computation*, vol. 2, pp. 1691–1696, May 2002.

[77] T. M. Blackwell and P. J. Bentley, "Dynamic search with charged swarms," *Proceedings* of Genetic and Evolutionary *Computation* Conference, pp. 19–26. 2002.

[78] T. K. Das, S. R. Jetti, and G. K. Venayagamoorthy, "Optimal design of a SVC controller using a small population based PSO." *Proceedings of IEEE Swarm Intelligence Symposium*, pp. 156–161, May 3006.

[79] T. Das and G. Venayagamoortliy, "Optimal design of power system stabilizers using a small population based PSO." Proceedings of IEEE *PES General Meeting*, June 2006.

[80] D. Parrot, X. Li, *Locating and Tracking Multiple Dynamic Optima by a Particle Swam Model Using Speciation." *IEEE Transactions on Evolutionary Computation*, vol. 10, no.4, pp. 440-458, August 2006.

[81] G. Venter and J. Sobieszczanski-Sobieski, "Particle swarm optimization." 43rd AIAAIASMUASCEIAHAIASC Structures. Structural Dynamics and Materials Conference, Denver. Colorado. USA. 2002.

[82] X. Hu and R. C. Eberhart, "Solving constrained nonlinear optimization problems with particle swarm optimization." *Proceedings of the Sixth World*

Multiconference on Systemics, Cybernetics and Informatics 2002 (SCI ZOOZ), Orlando. USA. 2002

[83] A. I. El-Gallad. M. E. El-tlawary, and A. A. Sallam, "Swarming of intelligent particles for salving the nonlinear constrained optimization problem." *International Journal* of *Engineering Intelligent systems for Electrical Engineering and Communications*, vol. 9, no. 3,pp. 155-163, September 2001.

[84] Y. del Valle, J. C. Hernandez, G. Venayagamoorthy, and R. G. Harley, "Multiple STATCOM allocation and sizing using particle swarm optimization." *Proceedings of IEEE* **PES** *Power Systems Conference and Exposition*, pp. 1884–1891, October 2006.

[85] K. E. Parsopodos and M. N. Vrahatis, M. N. 'Particle swarm optimization method for constrained optimization problems." *Proceedings of the 2nd Euro-international Symposium on Computational Intelligence*, pp.?14-220 ,2002.

[86] T. Ray. and K. M. Liew. "A swarm with an effective information sharing mechanism for unconstrained and constrained single objective optimization problem." *Proceedings of IEEE Congress on Evolutionary Computation*, pp. 75-80. 2001.

[87] L. Schoofs and B. Naudts, 'Swarm Intelligence on Binary Constraint Satisfaction Problem." *Proceedings of IEEE Congress on Evolutionary Computation*, Honolulu, Hawaii USA. 2002.

[88] G. Coath and S. Halgamgue "A Comparison of Constraint-handling Methods for the Applications of Particle Swarm Optimization to Constrained and Nonlinear

Optimization Problems," *Proceedings* of *IEE* Congress *on Evolutionary Computation* 2003, pp. 2419-2425, 2003.

[89] Z. Gaing, "Particle Swarm Optimization to Solving the Economic Dispatch Considering the Generator Conwraints." *IEE Transactions on Power Systems*. vol. 18. no.3 pp.1187-1195. 2003.

[90] K. E. Parsopodos and M. N. Vrahatis. "Particle swarm optimizer in noisy and continuously changing environments." *Artificial intelligence and* soji *computing*. pp. 289-294, IASTED/ACTA Press. 2001.

[91] J. Pugh, *A.* Martinoli, and Y. Zhang, ,"Particle swarm optimization for unsupervised robotic learning," *Proceedings* of IEEE *swarm* intelligence *symposium* pp. 92–99, June 2005.

[92] J. Kennedy and R. Eberhart. "A discrete binary version of the particle swarm algorithm," *Proceedings of IEEE International Conference on Systems. Man. and Cybernetics: Computational Cybernetics and Simulation (ICSMC)*, vol. 5, pp. 4104–4108. October 1997.

[93] K. Parsopoulos and M. Vrahatis, "Recent approaches to global optimization problems through particle swarm optimization," *Natural Computing*, vol. 1, pp. 235–306, May 2002.

[94] E. Laskari, K. Parsopoulos, and M. Vrahatis, "Particle swarm optimization for integer programming," Proceedings of *IEEE* Congress on *Evolutionary Computation*, vol. 2, pp. 1582–1587, May 2002.

[95] R. A. Krohling, "Gaussian Swarm: A Novel Particle Swarm Optimization Algorithm," *Proceeding of the IEEE Conference on Cybernetics and Intelligent Systems*, pp.372-376. December, 2004.

[96] X. Xie, W. Zhang, and Z. Yang, "A dissipative particle swarm optimization," *Proceedings of IEEE Congress an Evolutionary Computation*, vol. 2, pp. 1456–1461, May 2002.

[97] S. He, J. Wen, E. Prempain, Q. Wu, J. Fitch, and S. Mann, "An improved particle swam optimization for optimal power flow," *Proceedings of IEEE International Conference on Power Systems Technology*, pp. 1633–1637, November 2004.

[98] F. Bergh and A. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, June 2004.

[99] M. A. Potter and K. A. de Jong, "A cooperative co evolutionary approach to function optimization," in The Third Parallel Problem Solving From Nature. Berlin, Germany: Springer-Verlag, pp. 249–257, 1994

[100] S. Baskar and P. N. Suganthan, "A novel concurrent particle swarm optimization," *Proceedings of IEEE Congress on Evolutionary Computation*, vol. 1, pp. 792–796, June 2004.

[101] M. El-AM and M. S. Kamel, "A hierarchal cooperative particle swarm optimizer," *Proceedings of IEEE Symposium on Swarm Intelligence*, pp. 43-47, May 2006.

[102] G. Pampara, N. Franken, A. P. Engelbrecht, "Combining Particle Swarm Optimization with angle modulation to solve binary problems." *IEEE Congress on Evolutionary Computation*, vol.1, pp.89-96, 2005.

[103] R. Mendes, J. Kennedy, and J. Neves, "The Fully Informed Particle Swarm: Simpler, Maybe Better," *IEEE Transactions On Evolutionary Computation*, vol. 8, no. 3, pp.204-210, June 2004

[104] M. S. Voss, "Principal Component Panicle Swarm *Optimization:* a Step towards Topological Swarm Intelligence," *IEEE Congress on Evolutionary Computation*, vol. I, pp.298-305, September 2005.

[105] J. Riget, and J.S. Vesterstrom, "A Diversity Guided Particle Swarm Optimizer—the ARPSO", EVA-Life Technical *Report.* (http://www,evalife.dk), *2001.*

[106] P. Sardinians, R. Battiti, "The gregarious particle swarm optimizer (G-PSO)," *Proceedings of 8th annual conference on Genetic and evolutionary computation*, pp. 67-74,2006.

[107] M. A. S. Choudhry, M Zubair, A. Naveed, and I. M. Qureshi, "Near Optimum Detector for DS-CDMA System using Particle Swarm Optimization." *IEICE Transactions on Communications*, vol.E90-B,No.11,pp.3278-3282 Nov. 2007.

[108] M. Zubair, M. A. S. Choudhry A. N. Malik and I. M. Qureshi, "Particle Swarm with Soft *Decision* for Multiuser detection of Synchronous Multicarrier CDMA". *IEICE Transactions on Communications*, vol.E91-B, No.5, pp.1640-1643. May. *2008*

[109] M. Aamer Saleem Choudhry, "*New Schemes of MUD for Synchronous DS-CDMA and its Overloaded Systems*," Ph.D. thesis. M.A. Jinnah University 2007.

[110] G. Gudise and G. Venayagamoorthy, "Comparison of panicle swarm optimization and backpropagation as training algorithms for neural networks." *Proceedings of IEEE Swarm Intelligence Symposium*, pp. 110–117, April 2003.

[111] S. Mohaghegi, Y. del Valle, G.K. Venayagamoorthy and R.G. Harley. "A comparison of PSO and backpropagation for training RRF neural networks for identification of a power system with STATCOM," *Proceedings of IEEE Swarm Intelligence Symposium*, vol. , Issue , 8-10 pp. 381 – 384, June 2005.

[112] J. Salerno, "Using the Particle Swarm Optimization Technique to Train a Recurrent Neural Model," *Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence*, pp.45-49, 1997.

[113] C.-F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design." *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 2, pp. 997–1006, 2004.

[114] A Ismail, AP Engelbrecht, "Training Product Units in Feedforward Neural Networks using Particle Swarm Optimization," *Proceedings of the IEEE International Conference on Artificial Intelligence*, Durban, South Africa, 1999.

[115] F. Van den Bergh, AP Engelbrecht, "Using Cooperative Particle Swarm Optimization to Train Product Unit Neural Networks," *IEEE International Joint Conference on Neural Networks*, Washington DC, USA, 2001.

[116] F. Van den Bergh, and A.P. Engelbrecht, "Training product unit networks using cooperative panicle swarm optimizers," *Proceedings* of *INNS-IEEE International* Joint Conferenceon *Neural Networks* 2001, Wnshington DC, USA. 2001

[117] Y. Song, Z. Chen, and Z. Yuan. "New chaotic PSO-based neural network predictive control for nonlinear process." *IEEE Transactions on Neural Networks*, vol. IS. no. 2, pp. 595–601, 2007.

[I IS] C.-M. Huang and F.-L. Wang. "An RBF network *with* OLS and EPSO algorithms for real-time power dispatch." *IEEE Transactions on Power Systems*, vol. 22, no. 1, pp. 96–104, 2007.

[119] U. Paquet, and A.P. Engelbrecht, "Training support vector machines with particle swarms," Proceedings of *the International* Joint Conference *on Neural* Networks. pp. 1598,2003.

[120] C. Xiang, X. Huang, D. Huang, J. Hu, "Wavelets Neural Network Based on Particle Swarm Optimization Algorithm for Fault Diagnosis." icicic, pp. 320-323, *First International Conference on Innovative Computing, Information and Control* - Volume III (ICICIC'06), 2006

[121] M. Omran, A. Salman and AP Engelbrecht, "Image Classification using Particle Swarm Optimization," *Proceedings* of *the 4th Asia Pacific Conference on Simulated Evolution and Learning*, Singapore 2002.

[122] K. Chandramouli, E. Izquierdo, "Image Classification using Chaotic Panicle Swarm Optimization." *IEEE International Conference on Image Processing*, pp. 3001-3004, October 2006.

[123] M. Omran, A. Engelbrecht and A. Salman, "Particle Swarm Optimization Method for Image Clustering," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, pp. 297-322, no. 3, 2005.

[124] M. Omran, A. Salman and A. P. Enpelbrecht, "Dynamic Clustering using Particle Swarm Optimization with Application in Unsupervised Image Classification." *Fifth World Enformatika Conference*, Prague, Czech Republic, 2005.

[125] C.-C. LAI, "A Novel Image Segmentation Approach Based on Particle Swarm Optimization," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* E89-A(1) pp.324-327, 2006.

[126] Y.-C. Chen, H.-C. Wang, T.-J. Su, "Patticle Swarm Optimization for Image Noise Cancellatioa," *First International Conference on Innovative Computing, Information and Control* - Volume I (ICICIC'06), pp. 587-590, 2006.

[127] P.-Y. Yin, "Particle swarm optimization for point pattern matching," Journal of Visual *Communication and Image Representation*, vol. 17, Issue 1, pp. 143-162. February 2006.

[128] H. Talbi and M. Batouche, "Hybrid panicle swarm with differential evolution for multimodal image registration," *Proceedings of IEEE International Conference on Industrial Technology*, vol. 3, pp. 1567–1572, December 2004.

[129] M. Wachowiak, R. Smolikova, Y. Zheng, J. Zurada, and A. Elmaghraby, "An approach to multimodal biomedical image registration utilizing particle swarm optimization." *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 289–301, 2004.

[130] Y. Zhang, Y. Meng, W. Li, and Y. Pang "Texture synthesis using particle swarm optimization,", *Proceedings of International Conference on Communications, Circuits and Systems*, vol. 2, pp. 969- 973, June 2004.

[131] W. Slade, H. Ressonr. M. Musavi. and R. Miller, "Inversion of ocean color observations using particle swarm optimization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 9, pp. 1915~1923, 2004.

[132] T. Huang and A. S. Mohan, "A microparticle swarm optimizer for the reconstruction 'of microwave images," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 3, pp. 568~576, 2007.

[133] M. Donelli and A. Massa. "Computational approach based on a particle swarm optimizer for microwave imaging of two dimensional dielectric scatterers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53. no. 5, pp. 1761~1776, 2005.

[134] Y. Zhao, J. Zheng, "Particle Swarm Optimization Algorithm in Signal Detection and Blind Extraction," *International Symposium on Parallel Architectures, Algorithms and Networks*, pp. 37-41, May 2004.

[135] Y. Gao. S. Xie, "A blind source separation algorithm using particle swarm optimization," *Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication*, vol. 1, pp.297-300, 2004.

[136] Wen Wang; Yilong Lu; Fu, J.S., Yong Zhong Xiong, "Particle swarm optimization and finite-element baed approach for microwave filter design," *IEEE Transactions on Magnetics*, vol. 41, pp.1800 –803, May 2005.

[137] D. J. Krusienski, W. K. Jenkins, "Particle swarm optimization for adaptive IIR filter structures," *Congress on Evolutionary Computation*, vol.1, pp. 965-970 June 2004.

[138] D. J. Krusienski, W. K. Jenkins, "The application of particle swarm optimization to adaptive IIR phase equalization," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 693-, May 2004.

[139] D. J. Krusienski, W. K. Jenkins, "A particle swarm optimization-least mean squares algorithm for adaptive filtering,", *Proceedings of Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 241 – 245, November 2001.

[140] G. K. Venayagamoorthy and W. Zha, "Comparison of non-uniform optimal quantizer designs for speech coding with adaptive critics and particle swarm," *IEEE Transactions on Industry Applications*, vol. 43, no. 1, pp. 238–244, 2007.

[141] M. K. Tsatsanis, G. B. Giannakis, "Blind Estimation of Direct Sequence Spread Spectrum Signals in Multipath," *IEEE Transactions on Signal Processing*, vol. 45, pp. 1241–1252, May 1997.

[142] G. Tong, Z. Fang, X. Xu, "A Particle Swarm Optimized Particle Filter for Nonlinear System State Estimation," *IEEE Congress on Evolutionary Computation*, vol., pp. 438 – 442.2006

[143] V. R. Asghari, and M. Ardebilipour. "Spread Spectrum Code Estimation by Particle Swarm Algorithm," *Proceedings of World Academy of Science, Engineering and Technology*, vol. 7, pp.207-210, August 2005.

[144] Z. Lu and S. Yan, "Multiuser Detector Based on Particle Swarm Algorithm," *Proceedings of IEEE Symposium on, Emerging Technologies: Mobile and Wireless Communications*, Shanghai China, July 2004.

[145] C. Liu and Y. Xiao, "Multiuser Detection Using the Particle Swarm Optimization Algorithm," *Proceedings of IEEE International Symposium on Communications and Information Technology*, vol.1 pp.362-365, October 2005.

[146] Y. Zhao and J. Zheng, "Multiuser Detection Employing Particle Swarm Optimization in Space-time CDMA systems," *Proceedings of IEEE International Symposium on Communications and Information Technology*, vol. 2 pp.972-974, October 2005.

[147] M. Zubair, M. A. S. Choudhry A. N. Malik and I. M. Qureshi. "Particle Swarm optimization assisted Multiuser detection along with Radial Basis Function." *IEICE Transactions on Communications*, vol.E90-B no.7 pp.1861-1863 July 2007.

[148] M. Zubair, M. A. S. Choudhry A. N. Malik and I. M. Qureshi, "Multiuser detection for Asynchronous Multicarrier CDMA using Particle Swarm Optimization". *IEICE Transactions on Communications*, vol. E91-B, no.5, pp.1636-1639, May. 2008.

[149] G. Zhen-Qing, X. Yang, L. M. Ho. "Multiuser Detection Based on Particle Swarm Optimization Algorithm over Multipath Fading Channels" *IEICE Transactions on Communications*, vol.E90-B, no.2, pp.421-424, 2007.

[150] K. K. Soo, Y. M. Siu, W. S. Chan L. Yang and R. S. Chen, *R.S.*, "Particle-Swarm-Optimization-Based Multiuser Detector for COMA Communications," *IEEE Transactions on Vehicular Technology*, vol. 56, pp.3006 – 3013, September 2007.

[151] *Y.* Rahmat-Samii, D. *Gies* and J. Robinson. "Particle swarm optimization (PSO): a novel paradigm for antenna designs." The Radio Science Bulletin, vol. 305, pp. 14.22. Seprember 2003.

[152] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE Transactions on Antennas and Prupopttion.* vol. 52, no. 2, pp. 397.407, February 2004.

[153] N. Jin and Y. Rahmat-Samii, "Parallel particle swarm optimization and finite difference time-domain (PSO/FDTD) algorithm for multiband and wide-band patch antenna designs," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 11, pp. 3459.3468. November 2005.

[154] N. Jin and Y. Rahmat-Samii, "A hybrid real-binary particle swarm optimization (HPSO) algorithm: concepts and implementations," *IEEE Antennas and Propagation Society International Symposium*, pp. 1585.1588, July 2007.

[155] N. Jin and *Y.* Rahmat-Samii, 'Advances in particle swarm optimization for antenna designs: real-number, binary, single-objective and multiobjective

[162] T. B. Chen, Y. L. Dong, Y. C. Jiao, and F. S. Zhang, "Synthesis of circular antenna array using crossed particle swarm optimization algorithm," *Journal of Electromagnetic Waves and Applications*, vol. 20, no. 13, pp. 1785–1795, 2006.

[163] M. H. Bataineh, and J. I. Ababneh, "Synthesis of aperiodic linear phased antenna arrays using particle swarm optimization," *Electromagnetics*, vol. 26, pp.531–541, October 2006.

[164] M. Donelli. M., R. Azaro, F. G. B. Natale, and A. Massa, "An innovative computational approach based on a particle swarm strategy for adaptive phased-arrays control," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 3, pp. 888–898, March 2006.

[165] K. R. Mahmoud, , M.E. Adawy and S.M.M. Ibrahem, "A Comparison between circular and hexagonal array geometries for smart antenna systems using particle swarm optimization algorithm," *Progress in Electromagnetic Research*, 72, pp.75-90, 2007.

[166] H. M. Elkamchouchi, M. M. Wagih, "Dynamic Null Steering in Linear Antenna Arrays Using Adaptive Particle Swarm Optimization Algorithm." *Third International Conference on Wireless and Mobile Communications*, p. 24, 2007

[167] S. H. Zainud-Deen, , K. R. Mahmoud, M. El-Adawy, and S. M. M. Ibrahem, "Design of Yagi-Uda antenna and electromagnetically coupled curl antenna using particle swarm optimization algorithm," *22nd National Radio Science Conference*, pp. 115-124, March 15–17.7005.

[168] S. Baskar, A. Alphones, P.N. Sganthan and J.J. Liang, "Design of Yagi-Uda antennas using comprehensive learning particle swarm optimization," *IEEE Microwave Antenna and Propagation*, vol. 152, no. 5, pp. 340-346, October 2005.

[169] L. Zhengyi, G. Ke, "Design of a Printed Monopole Anrenna for Dualband WLAN Application Using PSO," *International Symposium on Antennas, Propagation & EM Theory*, pp.1 – 4, October 2006.

[170] J. Robinson. S. Sinton, and Y. Rahmat-Samii, Y., "Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna," *IEEE Antennas and Propagation Society International Symposium*, vol. 1, pp. 314-317.2002

[171] H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, and Y. Nakanishi. "A particle swarm optimization for reactive power and voltage control considering voltage security assessment," *IEEE Transactions on Power Systems*, vol. 15, no. 4. pp. 1232–1239, November 2000.

[172] A. Mantawy and M. Al-Ghamdi, "A new reactive power optimization algorithm," *Proceedings of IEEE Power Technology Conference*, vol. 4, pp.23-26, June 2003.

[173] W. Zhang and Y. Liu, -'Reactive power optimization based on PSO in a practical power sysrem." *Proceedings of IEEE Power Engineering Society General Meeting*, June 2004, vol. I, pp. 239–243.

[174] G. Coath, M. At-Dabbagh, and S. Halgamuge, "Particle swarm optimization for reactive power and voltage control with grid-integrated wind farms,"

Proceedings of IEEE Power Engineering Society General Meeting, June 2004, vol. 1, pp. 303–308.

[175] B. Zhao, C. Guo, and Y. J. Cao, "A multiagent-based particle swarm optimization approach for optimal reactive power dispatch," *IEEE Transactions on Power Systems*, pp. 1070–1078, May 2005.

[176] J. Park, K. Lee, J. Shin, and K. Y. Lee. "A particle swarm optimization for economic dispatch with non-smooth cost functions," *IEEE Transactions on Power Systems*, vol. 20, no. I, pp. 34-42. February 2005.

[177] K. Y. Lee and J. Park, "Application of particle swarm optimization to economic dispatch problem: Advantages and disadvantage," *Proceedings of IEEE Power Systems Conference Exposition*, pp. 188–192, October 2006.

[178] T. Aruldoss, A. Victoire, and A. Jeyakumar, "Reserve constrained dynamic dispatch of units with valve-point effects," *IEEE Transactions on Power Systems*, vol. *20*, no. 3, pp. 1273–1282, August 2005.

[179] L. Lai, T. Nieh, Y. Ma, Y. Lu, Y. Yang, and H. Braun, "Particle swarm optimization for economic dispatch of units with non-smooth input output characteristic functions," *Proceedings of 13th IEEE International conference on Intelligent Systems Application to Power Systems*, pp. 499–503, 2005.

[180] Z. Gaing, "Particle swarm optimization to solving the economic dispatch considering the generator constraints," *IEEE Transactions on Power Systems*, vol. 18, no. 3, pp. 1187–1195, August 2003.

[181] R. Pancholi and K. Swarup, "Particle swarm optimization for security constrained economic dispatch," Proceedings of *International Conference* on *Intelligent Sensing and Information Processing*, pp. 7–12, January 3004.

[182] B. Zhao, C. Guo, and Y. Cao. "Dynamic economic dispatch in electricity market using particle swarm optimization algorithm," *Proceedings of IEEE 5th World Congress on Intelligent Control and Automation*, pp.5050–5054, June 2004.

[183] A. El-Gallad, M. El-Hawary, A. Sallam, and A. Kalas, "Particle swarm optimizer for constrained economic dispatch with prohibited operating zones." Proceedings *of* IEEE Canadian *Conference* on Electrical and *Computer* Engineering. rol. I. pp. 78–81, May 2002.

[184] A. Kumar. K. Dhanushkodi, J. Kumar, and C. Paul. "Particle swarm optimization solution to *emission and* economic dispatch problem," *Proceedings of Conference on Convergent Technologies for Asia-Pacific Region*, vol. 1, pp. 435–439, October 2003.

[185] R. Chang and C. Lu. "Feeder reconfiguration for load factor improvement." Proceedings of *IEEE* Power. Engineering *Society* Winter Meeting, vol. 3, pp. 980–984, October 2002.

[186] C. Shen and C. Lu, "Feeder reconfiguration for power quality requirement and feeder service quality matching," Proceedings of *IEEE* Transmission and *Distribution Conference and Exhibition 2002: Asia Pacific*, vol. 1, pp. 226–231, October 2002.

[187] W.-C. Wu, M.-S. Tsai. F.-T. Hsu, "A new Binary Coding Panicle Swarm Optimization for Feeder Reconfiguration," Proceedings *of IEEE International* Conference on *Intelligent Systems* Applications *to* Power' Systems. pp. 1-6, November 2007.

[188] X. Jin. J. Zhao. K. Li, and B. Zhang, "Distribution network reconfiguration for load balancing using binary particle swarm optimization," *Proceedings of IEEE International Conference on Power System Technology*, pp. 507~510. November 2004.

[189] W. Kurutach and Y. Tuppadung, "Feeder-switch relocation based upon risk analysis of tree-caused interruption and value-based distribution reliability assessment," Proceedings of *2004* IEEE Region *10* Conference, vol. 3, pp. 577–580. November 2004.

[190] W. Kurutach and Y. Tuppadung, "Feeder-switch relocation for value-based distribution reliability assessment," *Proceedings of 2004 IEEE International Engineering Management Conference on Innovation and Entrepreneurship for Sustainable Development*, pp. 402–406, October 2004.

[191] M. Abido, "Optimal **design** of power-system stabilizers using particle swarm optimization." IEEE *Trans. Energy Conversion*, vol. 17. no. 3, pp. 406413, Sep. 2002.

[192] T. Das and G. Venayagamoorthy, "Bio-inspired algorithms for the design of multiple optimal power system stabilizers: SPPSO and BFA," *Proceedings of IEEE Industry Applications Society 41st Annual Meeting*, vol.2, pp.635-641, October 3006.

[193] A. Karimi, A. Al-Hinai, K. Schodcr. and A. Feliachi. "Power system stability enhancement using backstepping controller tuned by particle swarm optimization technique." *Proceedings of IEEE Power Engineering Society General Meeting*, pp. 1388-1395, January 2005.

[194] T. Okada, T. Watanabe, and K. Yasuda. "Parameter tuning of fixed structure controller for power system stability enhancement," *Proceedings of IEEE-PES* Transmission and Distribution Conference and *Exhibition* 2002: Asia *Pacific*, vol. 1, pp. 162-167. October 3002.

[195] W. Qiao. G. K. Venayagamoorthy, and R. Harley, "Design of optimal PI controllers for doubly fed induction generators driven by wind turbines using particle swarm optimization," *Proceedings of* IEEE *International Joint Conference on Neural Networks*, pp. 1982-1987. July 2006.

[196] Z. Gaing, "A particle swarm optimization approach for optimum design of PID controller in AVR system," IEEE *Transactions on Energy Conversion*, vol. 19, pp.384-391, June 2004.

[197] Y. Zheng, L. Ma. L. Zhsng and J. Qian, "Robust PID Controller Design using Particle swarm optimizer." *Proceedings of* IEEE *International Symposium on Intelligence Control*, pp. 974-979. 2003.

[198] Y. Abdel-Magid and M. A. Abido. "AGC tuning of interconnected reheat thermal systems with particle swarm optimization," *Proceedings of IEEE International Conference* on *Electronics, Circuits, Systems*, vol. 1. pp. 376-379, December 2003.

[199] K. Watanabe, A. Chatterjee, K. Pulasinghe, K. Izurni, and K. Kiguchi. "A fuzzy-neural network based human machine interface for voice-controlled robots trained by a particle swarm optimization," *Proceedings of 4th International Symposium on Advanced Intelligent Systems*, pp.411-414, September 2003.

[200] K. Watanabe, A. Chatterjee. K. Pulasinghe. S-H. Jin, K. Izumi, and K. Kiguchi. "Training of fuzzy-neural network for voice-controlled robot systems by a particle swarm optimization," Proceedings *of International* Conference on *Control, Automation, and Systems*, pp. 1115-1120, October 2003.

[201] A. Chatterjee, K. Pulasinghe. K. Watanabe, and K. Izurni. "A particle-swarm-optimized fuzzy-neural network for voice controlled robot systems." *IEEE Transactions on Industrial Electronics*, vol. 52, no. 6, pp. 1478–1489, **3005.**

[202] M. Saska. M. Macas. L. Preucil and L. Lhotska. "Robot Path Planning using Particle Swarm Optimization of Ferguson Splines". *Proceedings of IEEE/ETFA '06*, pp. **833-839.2006.**

[203] Q. Yuan-Qing, S. De-Bao, L. Ning and C. Yi-Gang, "Path planning for mobile robot using *the* particle swarm optimization with mutation operator." Proceedings of *International* Conference on *Machine Learning and Cybernetics*, **pp. 2473 – 2478.2004.**

[204] J. Pugh, A. Martinoli, and Y. Zhang, Y.. "Particle swarm optimization for unsupervised robotic learning," *Proceedings of IEEE swarm intelligence symposium*, pp. **92–99. 3005.**

[205] J. M. Hereford, "A Distributed Particle Swarm Optimization Algorithm for Swarm Robotic Applications," *IEEE* Congress an *Evolutionary Computation*, pp.1678 – 1685, 2006.

[206] W. Li. L. Yushu, D. Hongbin and X. Yuanqing, "Obstacle-avoidance Path Planning for Soccer Robots Using Particle Swarm Optimization". *Proceedings of* IEEE *International Conference on Robotics and Biomimetrics*, pp. 1233- 1238, 2006.

[207] J. Pugh, A. Martinoli "Inspiring and Modeling Multi-Robot Search with Particle Swarm Optimization." IEEE *Swarm Intelligence Symposium*, pp.332-339, April 2007.

[208] W. Jatmiko, K. Sekiyama, T. Fukuda, "A Mobile Robots PSO-based for Odor Source Localization in Dynamic Advection-Diffusion Environment," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4527-4532, October 2006.

[209] M. L. Doelz, E. T. Heald and D. L. Martin, "Binary Data Transmission Techniques for Linear Systems," Proceedings *of IRE*, vol. 45, pp. 656-616, May 1957.

[210] R. W. Chang, "Synthesis of band-limited orthogonal signals for multi-channel data transmission," Bell System Technical Journal, vol. 45, pp. 1775~1796, December 1966.

[211] R.W. Chang and R. A. Cibby, "A theoretical study of performance of an orthogonal multiplexing data transmission scheme," IEEE *Transactions on Communication Technology*, vol. COM-16, pp. 529–540. August 1968.

[212] B. Saltzberg, "Performance of an efficient parallel data transmission system," IEEE *Transactions an Communication* Technology, vol. 15, pp. 805-813, December 1967.

[213] S. B. Weinstein and P. M. Ebert. "Data transmission by frequency division multiplexing using the discrete Fourier transform." IEEE *Transactions on Communication Technology*, vol. 19. pp. 628–634, Oct. 1971.

[214] B. Hirosaki, "An Analysis of Automatic Equalizers for Orthogonally Multiplexed QAM Systems," IEEE Transactions on *Communications*, vol. 28. pp. 73-83, January 1980.

[215] B. Hirosaki, "An Orthogonally Multiplexed QAM System Using the Discrete Fourier Transform." IEEE *Transactions* on *Communications*, vol. 29. pp. 982-989. July 1981.

[216] L. J. Cimini, "Analysis and simulation of a digital mobile channel using onhogonal frequency division multiplexing." IEEE *Transactions on Communications*, vol. 33. pp. 665–675, July 1985.

[217] B. Hirosaki et al., "A 19.2 kb/s Voice-Band Data Modem Based on Orthogonally Multiplexed QAM Techniques." IEEE *International Conference* on *Communication*, pp. 661-665. Aug. 1985.

[218] Hirosaki, S. Hasegarva, and A. Sabato. "Advanced Group-Band Modem Using Orthogonally Multiplexed QAM Technique," *IEEE Transactions on Communications*, vol. COM-34. pp. 587-592, June 1986.

[219] I. Kalet and N. A. Zervos. "Optimized Decision Feedback Equalization versus Optimized Orthogonal Frequency Division Multiplexing for High-speed Data Transmission Over the Local Cable Network," Proceedings *of* IEEE *International* Conference on Communications, pp. 1080-1085, September 1989.

[220] Telebit Corporation, "Comparative Performance Results for Asymmetrical Duplex. V.32 (extended). and Multicarrier Modems." CCITT SG XVII, Contribution D56, September 1989.

[221] J. A. C. Bingham, "Multicarrier modulation for data transmission: An idea whose time has come." *IEEE Communications Magazine*, vol. 28. pp. 5–14. May 1990.

[222] IEEE P802 LAN/MAN Committee, "The working group for wireless local area networks (WLANs)." http://grouper.ieee.org/groups/802/11/index.html.

[223] ETSI, "Digital video broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television." Draft EN 300 744, V1.2.1. 1

[224] N. Yee. J.-P. Linnartz, and G. Fettweis. "Multicarrier CDMA for Indoor Wireless Radio Networks." *Proceedings of IEEE International Symposium on Personal*, Indoor *and Mobile Radio Communications*, pp. 109-113. September 1999.

[225] K. Fazel and L. Papke, "On the Performance of Convolutionally-Coded CDMA/OFDM for Mobile Communication Systems," *Proceedings of IEEE*

International Symposium on Personal. Indoor and Mobile Radio *Communications*, pp. 468-472, September 1999.

[226] M. Schnell and S. Kaiser. "Diversity Considerations for MC-SSMA Systems in Mobile Communications," *Proceedings of* IEEE *ISSSTA '96*, pp.131-135. September 1996.

[227] S. Kaiser and J. Hagenauer. "Multicarrier CDMA with iterative decoding and soft-interference cancellation," Proceedings of IEEE *GLOBECOM*, pp. 6–10, Noveber1997.

[228] M. S. Akther, J. Asenstorger, P. D. Alexander, and M. C. Reed, "Performance of multicarrier CDMA with iterative detection," Proceedings *of IEEE International Conference on Universal Personal Communications*, pp. 131-135, October 1998.

[229] H. *K.* Park, Eung. BaeKim, Y. Lee, and K. HyonTchah, "Multi-carrier CDMA system with parallel interference cancellation for multipath fading channels," *Proceedings of 9th* IEEE *International Symposium* on *Personal, Indoor and Mobile* Radio *Communications*, vol.?. pp. 513-517, 1998.

[230] S. Hara and R. Prasad, "Design and performance of multicarrier CDMA system in frequency-selective Rayleigh fading channels," IEEE *Transactions* on Vehicular Technology. vol. 48, pp. 1584–1595, September 1999.

[231] T. M. Lok. T. F. Wong, "Transmitter and Receiver Optimization in Multicarrier CDMA Systems." IEEE *Transactions on Communications*, vol. 48, pp.1197–1207. July 2000.

[232] A. C. McCormick, P. M. Grant, and J, S. Thompson, "A novel interference cancellation receiver for uplink multi-carrier CDMA," Proceedings of IEEE *International Symposium* on Spread *Spectrum Techniques* and *Applications*, pp. 155–158, September 2000.

[233] S. L. Miller and B. J. Rainbolt. "MMSE Detection of Multicarrier CDMA." IEEE *Journal on Selected Areas in Communications*, vol. 18. pp.2356-2362. November 2000.

[234] F. Petre, P. Vandenameele, A. Bourdoux, B. M. M. Moonen, and H. DeMan, "Combined MMSE/pcPIC multiuser detection for MC-CDMA." *IEEE Transactions* on *Vehicular* Technology, pp. 770-774, June 2000.

[235] S. M. Elnoubi and A. El-Beheiry. "Effect of Overlapping between Successive Carriers of Multicarrier CDMA on the Performance in a Multipath Fading Channel," IEEE *Transactions* on *Communications*, vol. 49. pp.769–773. May 2001.

[236] P. Zong, K. Wang, and Y. Bar-Ness. "Partial sampling MMSE interference suppression in asynchronous multicarrier CDMA system." IEEE *Journal on Selected* Areas in *Communications*, vol. 19, pp. 1605-1613, August 2001.

[237] D. Kalofonos, M. Stojanovic, and J. Proakis, "Performance of Adaptive MC-CDMA Detectors in Rapidly Fading Rayleigh Channels." IEEE *Transactions on* Wireless *Communications*, vol.2, pp. 229-239, March 2003.

[238] Q. Shi and M. Latva-aho, "Performance analysis of MC-CDMA in Rayleigh fading channels with correlated envelopes and phases," *IEE Proceedings Communications*, vol. 150, pp. 214-220, 2003.

[239] P. Kafle and A. Sesay, "Iterative Semi-Blind Multiuser Detection for Coded MC-CDMA Uplink System," IEEE *Transactions* on *Communications*, vol. 51, pp.1034-1039, July 2003.

[240] L. Brunel, '.Multiuser detection techniques using maximum likelihood sphere decoding in multicarrier CDMA systems." IEEE *Transactions on Wireless Communications*, pp. 949-957, May 1004.

[241] R. Liu. E. Chester, and B. Sharif, "Performance of asynchronous multicarrier CDMA multiuser receiver over frequency selective multipath fading channels." IEEE *Electronics Letters*, vol. 40. pp. 48-49. January 2004.

[242] J. G. Andrews, T. H. Y. Meng, "Performance of Multicarrier CDMA with Successive Interference Cancellation in a Multipath Fading Channel." IEEE *Transactions* on *Communications*, vol. 52, pp.811–822, May 2004.

[243] J. Wang and L. B. Milstein, "Multicarrier CDMA Overlay for Ultra-Wideband Communications," IEEE *Transactions on Communications*. vol. 52. pp.1664-1669. October 2004.

[244] J. Li, K. B. Letaief, "Reduced-Complexity MAP-Based Iterative Multiuser Detection for Coded Multicarrier CDMA Systems." IEEE *Transactions* on *Communications*, vol. 52, pp.1909–1994, November 2004.

[245] X. Cai. S. Zhou, and G. B. Giannakis. "Group-Orthozonal Multicarrier CDMA." *IEEE Transactions on Communications*, vol. 52, pp.90–99, January 2004.

[246] H. Wei and L. Hanzo. "Reduced –Complexity Near-Optimum Genetic Algorithm Assisted Multiuser Detection for Synchronous Multicarrier CDMA". ***Proceedings of Vehicular Technology Conference***, 2004.

[247] H. Wei and L. Hanzo. "Generic algorithm Assisted Multiuser Detection for Asynchronous Multicarrier CDMA." ***Proceedings of the IEEE Vehicular Technology Conference.*** October 2004.

[248] S. L. Hijazi, A. Best. B. Natarajan, and S. Das, "Novel low-complexity DS-CDMA multiuser detector based on ant colony optimization," *Proceedings of 60th IEEE Vehicular Technology Conference*, vol. 3, pp. 1939 - 1943. September 200

[249] S. L. Hijazi, A. Best. B. Natarajan, and S. Das. "Novel low-complexity MC-CDMA multiuser detector based on ant colony optimization," ***IEEE International Conference at Wireless and Mobile Computing, Networking and Communications***, August 2005.

[250] S. L. Hijazi and B. Natarajan, "Near-Optimal Multiuser Detection in Asynchronous MC-CDMA via the ant colony approach." *Proceedings of IEEE 2nd International Symposium on Wireless Computing*, vol. issue 5-7 . February 2007.

[258] J. Namgoong and J. S. Lehnert, "Performance of Multicarrier DS/SSMA Systems in Frequency-Selective Fading Channels," IEEE Transactions on *Wireless Communications*, vol.1 pp.236-244, April 2002.

[259] L. L. Yang. and L. Hanzo. "Performance of Generalized Multicarrier DS-CDMA Over Nakagami-m Fading Channels," IEEE *Transactions* on *Communications*, Vol.50 pp.956-966. June 2002.

[260] L. -L. Yang. and L. Hanzo. "Performance of Generalized Multicarrier DS-CDMA Using Various Chip Waveforms." *IEEE Transactions* on *Communications*. vol.51 pp.956-966, May 2003.

[261] S. Sener, 1. Develi and N. Karaboga, "Further performance analysis of the generalized MC DS-CDMA system in Nakagami-m fading channels." Computers and Electrical Engineering, doi: 10.1016/j.compeleceng. 2008.03.001, (in press).

[262] L.-L. Yang, H. Wei, and L. Hanzo, "A Multicarrier DS-CDMA System using Both Time-domain and Frequency-Domain Spreading." *Proceedings of the* IEEE *Vehicular Technology Conference*, pp. 2426-2430, September 2003.

[263] B. Hu, L.-L. Yang, H. Wei, and L. Hanzo. "Subspace-Based Blind and Group-Blind Space-Time Multiuser Detection for the Generalized Multicarrier DS-CDMA Uplink," *Proceedings of Vehicular Technology Conference*, pp.1-5, 2006.

[264] L. -L. Yang. and L. Hanzo, "Performance of multicarrier DS-CDMA using space time spreading-assisted transmit diversity." IEEE *Transactions* on *Wireless Communications*, vol. 4, no.3, pp. 885-894, May 2005.

[265] L. Vandendorpe. "Multitone direct sequence CDMA system in an indoor wireless environment," Proceedings IEEE *First Symposium* of *Communications and Vehicular* Technology, pp. 4.1.1-4.1.8, October 1993.

[266] L. Vandendorpe, "Multitone spread spectrum multiple access communications systems in a multipath Rician fading channel." IEEE *Transactions on Vehicular Technology*, vol.44, no.?. pp.327-337, May 1995.

[267] L. Vandendorpe. "Overview of the results about multitone CDMA detection." *Proceedings First International* Workshop *on Multi-Carrier Spread-Spectrum*, Kluwer Academic Publishers. pp. 13-21. 1997.

[268] Q. M. Rahman and A. B. Sesay, "Performance Evaluation of Non-coherent Multi-tone CDMA system." Proceedings of *IEEE* CCECE. pp. 245-249. May 2000.

[269] K.-W Yip, X. Zhang, T.-S Ng. J. Wang, "On the multiple access capacity of multi-tone CDMA communications." *IEEE Communications Letters*, vol.4, No.2, pp. 40-42, February 2000.

[270] Q. Me. Rahman and Abu B. Sesay. Noncoherent MT-CDMA system with Diversity combining, *Proceedings* of *IEEE* CCECE. pp. 1351-1356. May. 2001.

[271] D. W. Matolak, V. Deepak, F. *A.* Alder, "Performance of Multitone & Multicarrier DS-SS in the Presence of Imperfect Phase Synchronization." Proceedings of *MILCOM 2002,* pp.1002-1006, October 7-10.2002.

[272] H. Li, D. W. Matolak, "Performance of MT-DS-SS in the Presence of phase noise." Proceedings *of MILCOM 2002*, pp.674-678. October. 2002.

[281] A. G. Bors, G: Gabbouj, "Minimal topology for a radial basis function neural network for pattern classification." *Digital Signal Processing: a review journal*, vol. 4. no. 3. pp. 173-188, 1994.

[282] M. W. Mak and K. W. Cho. "Genetic evolution of radial basis function centers for pattern classification." *Proceedings* of *International Joint Conference on Neural Networks*, vol. 1, pp. 669-673, 1998.

[283] A. M. S. Zalzala and N. Chaiyaratana, "Myoelectric signal classification using evolutionary hybrid RBF-MLP networks." Proceedings *of* Congress on *Evolutionary Computation*, vol. 1,2000, pp. 691698, 2000.

[284] T. Poggio. F. Girosi. "Networks for approximation and learning." Proceedings *of IEEE*, vol.78, no.9. pp. 1481-1497, 1990.

[285] J. González, I. Rojas, J. Ortega, H. Pomares. F. Fernandez. and A. F. Diaz. "Multiobjective evolutionary optimization of the size. shape, and position parameters of radial basis function networks for function approximation," *IEEE Transactions* on *Neural* Networks, vol. 14, no. 6, pp. 1478-1495, December 2003.

[286] I. Cha, S. A. Kassam, "RBFN restoration of non-linearly degraded images." IEEE *Transactions* on Image Processing, vol.5, no. 6, pp.964-975. 1996.

[287] M. Casdagli. "Nonlinear prediction of chaotic time series." Physica D, vol. 35, pp. 335-356, May 1989.

[288] S. Aiguo and L. Jiren. "Evolving Gaussian RBF network for nonlinear time series modeling and prediction." *Electronic Letters*, vol. 34. no. 12. pp. 1241–1243, 1998.

[289] Q. Zhang, X. He, and J. Liu, "RBF network based on genetic algorithm optimization for nonlinear time series prediction." *Proceedings of International Symposium on Circuits and Systems*, vol. 5, pp. V–693–V–696, 7003.

[290] I. Cha and S. A. Kassam. "Interference cancellation using radial basis function network." *Signal Processing*, vol. 47, pp. 247-268, 1995.

[291] S. Chen. C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Transactions on Neural Networks*, vol. 2. no. 2, pp. 302-309, 1991.

[292] W. Zhao, D.-S. Huang, and L. Guo, "Optimizing radial basis probabilistic neural networks using recursive orthogonal least squares algorithms combined with micro-genetic algorithms." *Proceedings of International Joint Conference Neural Networks*, vol. 3. pp. 2277–2282, 2003.

[293] J. E. Moody and C. J. Darken. "Fast learning in networks of locally tuned processing Units." *Neural Computation*, vol.1. pp. 281-294, 1989.

[294] B. Aazhang, B. P. Pans, and G. Orsak, "Neural networks for multiuser detection in CDMA communication," *IEEE Transactions on Communications*, vol. 40. no. 7, pp. 1212-1222, July 1992.

[295] Mitra and H. V. Poor. "Neural Networks Techniques for Adaptive Multiuser Demodulation." *IEEE Journal on Selected Areas in Communications*, vol. 12. no. 9. pp. 1460-1470. December 1994.

[296] D.G. M. Cruicshank. "Radial Basis Function Receivers for DS-CDMA." *Electronics Letters*, vol. 32. pp.188-190, February 1996.

[297] R. Tanner and D. G. M. Cruickshank, S.Z. W. H. Sweatman and B. Mulgrew, ."Receivers for Nonlinearly Separable Scenarios in DS-CDMA." *IEEE Electronics Letters*, vol. 33. pp.2103-2105, December 1997.

[298] R. Tanner and D. G. M. Cruickshank, "RBF based Receivers for DS-CDMA with Reduced Complexity." *Proceedings of ISSSTA*, pp.647-651, September1998.

[299] G. M. A. Sessler, E. Witzel and F. K. Jondral. "Performance Evaluation of an RBF Based Receiver as Multiuser detector for TDD-CDMA." ***Proceedings** of European Wireless*, pp. 231-235, September 2000.

[300] H. Wei and L..Hanzo, "Reduced-complexity genetic algorithm aided and radial basis function assisted multiuser detection for synchronous CDMA." *Proceedings of EURASIPCO*, pp.157-160. September 2004

[301] S. X. Ng, M. S. Yee and L. Hanzo, "Radial Basis Function Aided Turbo Equalizations of TCM. TTCM, BICM, BICM-ID Assisted Wireless Transceivers". *Proceedings of International Symposium on Turbo Codes*, pp. 67-74, September 1-5, 2003

[302] M. S. Yee. S. X. Ng and L. Hanzo. "Radial Basis Function Assisted Reduced Complexity In-phase/Quadrature-phase Turbo Equalization of Coded Modulation

Schemes". Proceedings of IEEE *Vehicular Technology Conference*, Vol. 4. pp. 2367-2371, April 22-25.

[303] M. S. Yee. S. X. Ng and L. Hanzo. "Iterative radial basis function assisted turbo equalisation of various coded modulation schemes", *Proceedings of IEEE Vehicular Technology Conference*, vol. 4, pp. 1705-1709. May 2002.

[304] S. X. Ng, M. S. Yee and L. Hanzo, "Coded Modulation Assisted Radial Basis Function Aided Turbo Equalisation for Dispersive Rayleigh Fading Channels", IEEE Transactions on Wireless Communications, vol. 3, no. 6. pp. 2198-2206, November 2004.

[305] S. Chen, S. Mclaughin and B. Mulgrew, "Complex-valued Radial Basis Function Network Part-I: Network Architecture and Learning Algorithms Signal Processing," *EURASIP Signal Processing Journal*, vol.35, pp.19-31, January 1994.

[306] S. Chen, S. Mclaughin and B. Mulgrew, "Complex-valued Radial Basis Function Network Part-II: Application to Digital Communications Channel Equalisation," *EURASIP* Signal Processing *Journal*, vol.35, pp.19-31, January 1994.

[307] M. S. Yee, "Radial Basis Network Based Multilevel Channel Equalisation Techniques for Wireless Communications," PhD thesis. University of Southampton, 2001.

[308] S. U. H. Qureshi. "Adaptive Equalization." *Proceedings of the IEEE*, Vol. 73. pp. 1349-1387, September 1985.

[309] L. Tong and S. Perreau, "Multichannel blind identification: From subspace to maximum likelihood methods." ***Proceedings of IEEE***, vol.86, no.10, pp.1951–1968. 1998.

[310] Y. Sato, "A Method of Self-Recovering Equalization for Multi-level Amplitude Modulation," *IEEE Transactions on Communications*, vol. COM-29, pp. 679-683. June 1975.

[311] F.-C. Zheng, S. McLaughlin, and B. Mulgrew. "Blind equalization of non-minimum phase channels: Higher order cumulant based algorithm." ***IEEE Transactions on Signal Processing***, vol. 41, pp. 681–691, February 1993.

[312] A.P. Petropulu and C.L. Nikias, "Blind Deconvolution Using Signal Reconstruction. from Partial Higher Order Cepstral Information." ***IEEE Transactions* on *Signal Processing*** vol. 41, pp.2088-2095, June 1993.

[313] J.F. Cnrdoso, "Source separation using higher ordered moments." *Proceedings of ICASSP-1989*, vol.4, pp.2109–2112. 1989.

[314] P. Comon. "Independent component analysis: A new concept?" ***Signal Processing. Elsevier***, vol.36, no.3, pp.287–314, 1994.

[315] C.-Y. Chi, C.-Y. Chen, C.-H. Chen and C.-C. Feng, "Batch processing algorithms for blind equalization using higher-order statistics." ***IEEE Signal Processing Magazine***, vol.20, no.1, pp.25–49, 2003.

[316] D. Godard. "Self-recovering equalization and carrier tracking in two dimensional data communication systems." ***IEEE Transactions on Communications***, vol. COMM-78. pp. 1867–1875. 1980.

[324] K. Abed-Menim. Y. Xiang. J.H. Manron and Y-B. Hua. '-A new approach to blind separation of cyclostationary sources." *Proceedings of 2nd IEEE Workshop Signal Processing Advances in Wireless Communications*. pp. 114–117. May 1999.

[325] A. Hussain, A. Naveed, I. M. Qureshi, "New Hybrid HOS-SOS Approach for Blind Equalization of Communication Channels," ***IEE Electronics Letters***, vol. 41, no.6, pp.376-377.2005.

[326] M. Ghosh and C. L. Weber, "Maximum-likelihood blind equalization." *Proceedings of SPIE. San Diego*, CA. vol. 1563. pp. 188–195. 1991.

[327] N. Seshadri. "Joint Data and Channel Estimation Using Blind Trellis Search Techniques," *IEEE Transactions on Communications*. vol. 42, pp. 1000-1011, February/March/April 1994.

[328] M. J. Omidi. P. G. Gulak, and S. Pasupathy, "Parallel structures for joint channel estimation and data detection over fading channels," ***Proceedings of IEEE VLSI Signal Processing IX***, pp. 325–336, 1996.

[329] M. J. Omidi, S. Pasupathy, and P. G. Gulak, "Joint data and Kalman estimation of fading channel using a generalized Viterbi algorithm," *Proceedings of International Conference on Communications (ICC'96)*. pp. 1198–1203, 1996.

[330] M. J. Omidi. S. Pasupathy. P. G. Gulak. "Joint data and Kalman estimation of fading channel using a generalized Viterbi algorithm." *International Conference on Communications* (ICC'96), pp. 1198-1203. 1996.

[331] M. J. Omidi, P. G. Gulak, and S. Pasupathy, "Joint data and channel estimation using the per-branch processing method," Proceedings of *IEEE Signal* Processing Workshop *Wireless Communications*, pp.389–392, April 1997.

3 H. Zhu, B. Farhang-Boroujeny, C. Schlegel, "Pilot embedding for joint channel estimation and data detection in MIMO communication systems," IEEE *Communications Letters*, vol.7, no.1, pp.30-2. January 2003.

[333] P. Stoica, H. Vikalo, and B. Hassibi, "Joint maximurn-likelihood channel estimation and signal detection for SIMO channels," *Proceedings* of *IEEE* ICASSP. vol. 4, pp. 13–16, April 2003.

[334] S. Buzzi, M. Lops and S. Sardellitti, "Performance of iterative data detection and channel estimation for single-antenna and multiple-antennas wireless communications," *IEEE Transactions on Vehicular Technology*, vol.53, no.4, pp. 1085–1104, 2004.

[335] S. Chen, X.C. Yang and L. Hanzo. "Blind joint maximum likelihood channel estimation and data detection for single-input multiple-output systems," *Proceedings of 6th IEE International Conference on 3G & Beyond*, pp. 5 pages. November 2005..

[336] M. Luise, R. Reggiannini, and G. M. Vitetta, "Blind equalizationldetection for OFDM signals over frequency-selective channels," IEEE *Journal* on *Selected Areas in Communications*, vol. 16, no. 8, pp. 1568–1578, October 1998.