The Hong Kong Polytechnic University

Department of Electronic and Information Engineering

An Improved Particle Swarm Optimization Algorithm and its Applications

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Philosophy

By

Yeung Chun Wan

August 2009

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degrees or diploma, except where due acknowledgement have been made in the text.

_____ (Signed)

Yeung Chun Wan_____(Name of student)

ABSTRACT

This thesis focuses on Particle Swarm Optimization and multi-kernel Support Vector Machine. Results in the following areas will be reported: (1) a real-coded Particle Swarm Optimization algorithm with a new mutation operation, (2) its application to half-tone image restoration, and (3) tuning a support vector machine with multi-kernel operations, which is applied to gene selection of DNA microarray data.

The Particle Swarm Optimization (PSO) is one of the evolutionary computation techniques, which is a population based stochastic optimization algorithm. It is inspired by the social behaviour of animals like fish schooling or bird flocking. In this thesis, PSO with a new mutation operation called multi-wavelet mutation (MWPSO) will be presented. By taking advantage of the wavelet theory, the mutation operation is enhanced such that the performance of the PSO is improved in terms of the fitness of cost function, solution stability and convergence rate. A suite of benchmark test functions is used to evaluate the performance of the proposed algorithm.

Application of the proposed MWPSO to half-tone image restoration is investigated. Half-toning is a reprographic technique that converts a continuous-tone image to a lower resolution, which is mainly used for printing. Error diffusion is one of the half-toning methods. It makes use of the fact that the human visual system is less sensitive to higher frequencies, and diffuses the quantization noise into high frequencies. In recent research, restoration of color-quantized images is rarely

addressed, especially when images are color-quantized with half-toning; and most existing restoration algorithms are mainly to deal with the noisy blurred images. Simulations have been carried out to evaluate the performance of the proposed MWPSO on realizing half-toned image restoration. Thanks to the multi-wavelet mutation, which seeks for a balance between the exploration and exploitation of the searching space for the swarm, it is found that the result can achieve a remarkable improvement in terms of convergence rate and signal-to-noise ratio.

Support vector machines (SVMs) are one of the supervised learning methods, which are used for doing classification and regression. By viewing input data as two sets of vectors and transforming the data in an *n*-dimensional space, an SVM can be constructed such that a separating hyperplane in the space that maximizes the margin between the two data sets is formed. In this thesis, an integrated approach of SVM with multiple kernels will be presented. The kernel of the SVM is realized as a linear combination of three commonly used kernels, and the weighting of each kernel are tuned by the proposed MWPSO. By using the integrated approach, the performance of the feature selection done by the SVM can be improved. An application example on gene signature selection of microarray data is used to show the performance of the proposed method. As DNA microarray studies produce a large amount of data, expression data are highly redundant and noisy such that most of the genes are believed to be uninformative with respect to the studied classes. Only a fraction of genes may present distinct profiles for different classes of samples. The proposed method is introduced to deal with these issues. We simultaneously optimize the selection of feature subset and the classifier through a common solution coding

mechanism. Thanks to the proposed MWPSO, the simulation results show improved performance over existing methods in terms of classification accuracy.

ACHNOWLEDGEMENT

I would like to express my sincere gratitude to my chief supervisor, Dr Frank H. Leung for his guidance, support and valuable advice. I would also like to thank my four colleagues, Dr H. K. Lam, Dr Steve S. H. Ling, Mr Johnny C. Y. Lai, and Ms Ginny Y. K. Wong for the helpful advice and discussions during the progress of the research. Thanks are extended to my family and my girlfriend, Ms Amy S.W. Lee, for their continual encouragement. I truly appreciate the help from the staff in the General Office of the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University.

The work described in this thesis was substantially supported by a grant from the Hong Kong Polytechnic University, the Hong Kong Special Administrative Region (Project account code #RGYY.)

STATEMENT OF ORIGNALITY

The following contributions reported in this thesis are claimed to be original.

1. *The multi-wavelet mutation for Particle Swarm Optimization (Chapter 3, Section 3.2.2.)* The multi-wavelet mutation applied the wavelet theory. By controlling the numbers of elements undergoing mutation, more freedom will be given to explore the searching space, and the convergence and solution stability are improved.

2. *The image restoration using MWPSO (Chapter 4, Section 4.3.)* The problem of recovering a color-quantized half-toned image back to the original image has been solved by using the proposed Particle Swarm Optimization with multi-wavelet mutation.

3. *Gene signature selection using MWPSO with multi-kernel Support Vector Machine (Chapter 5, Section 5.2.)* The problems of gene classification and tuning the parameters of the support vector machine have been solved by applying the proposed Particle Swarm Optimization with multi-wavelet mutation.

TABLE OF CONTENTS

AUTHOR'S PUBLICATIONS

INTERNATIONAL CONFERENCE PAPERS

[1] H.K. Lam, F.H.F. Leung, and C.W. Yeung, "Stability analysis and performance deign for fuzzy-model-based control system under imperfect premise matching" in *Proc. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2007)*, London, 23-26 July, 2007, pp. 1-6.

[2] H.K. Lam, F.H.F. Leung, and C.W. Yeung, "Stability conditions for fuzzy control systems with fuzzy feedback gains" in *Proc. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2007)*, London, 23-26 July, 2007, pp. 1-6.

[3] H.K. Lam, S.H. Ling, H.H.C. Lu, C.W. Yeung, and F.H.F. Leung, "Control of nonlinear systems with a linear state-feedback controller and a modified neural network tuned by genetic algorithm," in *Proc. IEEE Congress on Evolutionary Computation 2007 (CEC 2007)*, Singapore, 25-28 Sept, 2007, pp. 1614-1619.

[4] S.H. Ling, C.W. Yeung, K.Y. Chan, H.H.C. Lu, and F.H.F. Leung, "A new hybrid particle swarm optimization with wavelet theory based mutation operation," in *Proc. IEEE Congress on Evolutionary Computation 2007 (CEC 2007)*, Singapore, 25-28 Sept, 2007, pp. 1977-1984.

[5] C.W. Yeung, S.H. Ling, K.Y. Chan, and F.H.F. Leung, "Restoration of half-toned color-quantized images using particle swarm optimization with wavelet

mutation," in *Proc. IEEE Region 10 Conference(TENCON 2008)*, Hyderabad, 19-21 Nov, 2008, pp. 1-6.

[6] C.W. Yeung, F.H.F. Leung, K.Y. Chan, and S.H. Ling, "An integrated approach of particle swarm optimization and support vector machine for gene signature selection and cancer prediction," in *Proc. 2009 International Joint Conference On Neural Networks (IJCNN 2009)*, Atlanta, 14-19 June, 2009, pp. 3450- 3456.

[7] C.W. Yeung, C.Y. Lai, and F.H.F. Leung, "A particle swarm optimization with multi-mutation operation base on wavelet theory," *2009 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC 2009)*, Japan, 4-7 Oct, 2009, (Accepted).

[8] C.Y. Lai C.W. Yeung, and F.H.F. Leung, "A new hybrid differential evolution with wavelet based mutation and crossover," *2009 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC 2009)*, Japan, 4-7 Oct, 2009, (Accepted).

INTERNATIONAL JOURNAL PAPERS

[1] S.H. Ling, H.H.C. Iu, K.Y. Chan, H.K. Lam, C.W. Yeung, and F.H.F. Leung, "Hybrid particle swarm optimization with wavelet mutation and its industrial applications," *IEEE Trans. Syst., Man and Cybern., Part B: Cybernetics*, vol. 38, no. 3, pp. 743-763, Jun. 2008.

LIST OF SYMBOLS

$X(t)$	Swarm
w	Inertia weight factor for PSO
φ_1, φ_2	Acceleration constants for PSO
k	Constriction factor for PSO
$v(t),\ v_j^p(t)$	Velocity of a particle and its element
v_{max}	Maximum velocity of a particle
$\mathbf{x}^p(t)$	Particle of a swarm
$x_\kappa^p(t)$	the κ -th elements of a particle
γ	The number of particles in a swarm
t	Time
β	Step ratio of gradient descent rule
a	Dilation parameter for wavelet function
b	Translation parameter for wavelet function
$\psi(x)$	Wavelet function
∇f	Gradient of the fitness function
ε	Small constant for gradient descent rule
E^p	Standard basis vector for gradient descent rule
μ_m	Probability of mutation
N_m	Element probability
$para_{\max}^j,\ para_{\min}^j$	Maximum and minimum values of a particle element

ζ_{wm}	Shape parameter of a monotonic increasing function
$\vec{O}_{(i,j)}$	The original image
$\vec{U}_{(i,j)}$	State vector of the color quantization system
$\vec{E}_{(i,j)}$	Quantization error of a pixel at certain position
$H_{(i,j)c}$	Coefficient of the error diffusion filter
$Q_c[\cdot]$	Quantization function
C	Color palette
$\vec{Y}_{(i,j)}$	Color-quantized image
D	Gene microarray dataset
$\ell(\cdot)$	Gene select function
$\phi(\cdot)$	Non-linear mapping function
w	Separating hyperplane of SVM
K	Kernel function
ω_i	weight for each kernel
μ	Mean
Var	Variance

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

APSO	Ahmed's PSO
HGAPSO	Hybrid Genetic Algorithm PSO
HGPSO	Hybrid Gradient Descent PSO
HPSOM	Hybrid PSO with Mutation
LOOCV	leave-one-out cross-validation
MWPSO	PSO with Multi-Wavelet Mutation
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
SA	Simulated Annealing
SVM	Support Vector Machine
SNRI	Signal-to-Noise Ratio Improvement
Std Dev	Standard Deviation
WPSO	PSO with Wavelet Mutation

CHAPTER 1

INTRODUCTION

This thesis focuses on the problem of optimization using global search techniques. The evolutionary algorithm of Particle Swarm Optimization is to be investigated. The performance and results of a proposed Particle Swarm Optimization with multi-wavelet mutation will be reported. Apart from being tested by eighteen benchmark functions, the proposed algorithm is also applied in two application examples: the restoration of color-quantized half-toned images and the DNA microarray feature selection.

1.1 PARTICLE SWARM OPTIMIZATION

1.1.1 Introduction

Particle Swarm Optimization (PSO) [J. Kennedy, 1995] is a recently proposed population based stochastic optimization algorithm, which is one evolutionary algorithm technique for tackling complex optimization problems. It is inspired by the social behavior of animals like fish schooling and bird flocking. PSO involves a number of particles moving around the searching space and interacting with each others to look for the best solution. Comparing with other population based stochastic optimization methods, such as the evolutionary programming, PSO has comparable or

even superior search performance for many hard optimization problems, offering a faster and more stable convergence rate [J. Kennedy, 2001].

Basically, the standard PSO involves only one operation: updating the positions of the particles. Unlike other evolutionary algorithms like genetic algorithm and evolutionary programming, PSO does not have the selection operation. The position update is based on sharing the information between all the particles, and then the particles will move toward a new position accordingly. This is analogous to people discussing among each other on solving a problem before coming up with some behavioral change. In practice, the particles form a social communication network. Each particle will evaluate its fitness and remember its best position. The individual best solution is defined as the local best. This information will be shared to all the particles, and the global best is found as the best particle position among all. The motion of each particle is guided by the information of the local best and the global best. After a certain number of iteration, the particles will converge to the problem solution.

1.1.2 Limitations and other improvements to PSO

Observations reveal that PSO converges sharply in the earlier stage of the searching process, but it saturates or even terminates in the later stage. It behaves like the traditional local search methods that might get trapped in local optima. It is hard to obtain any significant improvement by examining neighboring solutions in the later stage of the search. Vaessens *et al* [Vaessens, 1992] and Reeves [C.R. Reeves, 1994] put PSO into the context of local search or neighborhood search.

Ahmed *et al.* proposed a hybrid PSO [A.A.E. Ahmed, 2005] that integrates GA's mutation into it with a constant mutating space. Under this approach, the particles can follow different directions, and the local positions of particles can be mutated. The solution space can still be explored in the later stage of the search thanks to the mutation operation. Pre-mature convergence is more likely to be avoided. However, under that approach, the mutating space is kept unchanged throughout the time of the search. It can be further improved by varying the mutating space along the search.

1.2 COLOR-QUANTIZED HALF-TONED IMAGE

1.2.1 Color quantization

Color quantization is a process of reducing the number of colors in a digital image by replacing them with some representative colors selected from a palette [M.T. Orchard, 1991]. When color quantization is performed, certain degradation of quality will be introduced owing to the limited number of colors used to produce the output image. The most common artifact is the false contour. False contours occur when the available palette colors are not enough to represent a gradually changing region. Another common artifact is the color shift. In general, the smaller the color palette size is used, the more severe the defects will be. For reducing these defects, digital half-toning would be helpful as the human eyes inherently exhibit some spatial low-pass filter characteristics.

1.2.2 Digital half-toning

Half-toning [X. Wu, 1992] [S.S. Dixit, 1991] [R.S. Gentile, 1990] [P. Heckbert, 1982] [R. Ulichney, 1987] is a reprographic technique that converts a continuous-toned image to a lower resolution mainly for printing, which was developed by W.F. Talbot in 1850s. Traditional half-toning reproduces the image using dots of varying size to simulate the different shades of color. These tiny halftone dots can be blended into smooth tones again by the eyes if we look at the halftone image from a certain distance. Digital half-toning is similar to traditional half-toning in which an image is decomposed into a grid of half-toned cells. At

present, error diffusion is one of the most popular digital half-toning methods. It takes advantage of the fact that human eyes exhibit spatial low-pass filter characteristics. During color quantization, the quantization error of a pixel is diffused to the neighboring pixels so as to hide the defects and to achieve a more faithful reproduction of colors.

1.3 SUPPORT VECTOR MACHINE

1.3.1 Introduction

Support Vector Machine (SVM) [B. Boser, 1992] is one of the tools for realizing supervised learning, which is very useful for doing classification and regression. By viewing input data as two sets of vectors and transforming the data into an *n*-dimensional space, an SVM can be constructed such that a separating hyperplane in the space that maximizes the margin between the two data sets is formed.

1.3.2 Kernel methods

Support Vector Machines can be regarded as algorithms that use the kernel method. For a given problem, the kernel method maps the data into a higher dimensional feature space by using a kernel function. Typical kernel functions like the linear, sigmoid, and radial basis functions are widely used in classification studies [V.N. Vapnik, 1998] [V. Vapnik, 2000]. In the feature space, each point corresponds to one feature of the data items.

1.3.3 Classification

Classification is a common task done by the Support Vector Machine. Suppose some given data belong to any one of two different classes, the classifier

should be able to identify the class the data belong to and decide to which class a new data should belong. If each data is treated as an *n*-dimensional vector, the classifier should construct an *n*–1 dimensional hyperplane to separate them. However, there might be many hyperplanes that can satisfy the problem requirement; in other words, there is no trivial solution. Finding out the maximum margin between two classes should therefore be achieved by the SVM such that the distance between the nearest data and the hyperplane is maximized. If such a hyperplane exists, this linear classification is known as a maximum margin classification.

1.4 MICROARRAY

1.4.1 Introduction to microarray

Living organisms often have trillions of cells, and each cell carries the same DNA pattern. However, only a fraction of the genes inside DNA are active in any given cell. That means different genes are expressed for some specific function of a cell. DNA microarrays are popular tools for showing how strongly a particular gene is expressed in a sample. The arrays are usually attached to glass, plastic or silicon chips, which consist of thousands of microscopic spots of single-stranded DNA called features. Each feature contains a specific DNA sequence. When the targets, which are short sections of genes or DNA fragments from an unknown sample, are deposited onto the array, they will only stick to the complement sequences. As a result, fluorescent spots will be shown on the array such that the intensity of the fluorescent emission at a given probe location is directly proportional to the amount of DNA fragments with the same identity present in the sample.

1.4.2 Cancer study using microarray

A microarray chip can simultaneously interrogate thousands of genes, which provides an extremely powerful tool for the genomic studies of cancers. A few key genes, when mutated, will cause deregulation of the transcription and translation of other genes through complicated signaling pathways to initiate oncogenesis, and ultimately leading to derangement of the cellular phenotype and the clinical

manifestation of cancer [M.D. Radmacher, 2002]. Analyzing microarray data could therefore help discover some significant cancer genes and their mutual interactions, which can be used to generate hypothesis for the identification and validation of genetic biomarkers for diagnostic and therapeutic purposes [M. Daly, 2002] [M.B. Eisen, 1998] [L.M. Fu, 2003] [T.R. Golub, 1999].

1.4.3 Limitations of microarray

As DNA microarray studies produce a large amount of data, the expression data become highly redundant and noisy, and most genes are believed to be uninformative with respect to the studied classes. Only a fraction of genes may present distinct profiles for different classes of samples. The data are noisy because some contaminants might be introduced to the DNA samples when they go through the processing steps. Some sequences cannot be expressed as the fluorescent marker may not successfully bind to all the sequences in the sample.

1.5 ACHIEVEMENTS

An improved hybrid Particle Swarm Optimization (PSO) that incorporates a mutation operation is proposed. The conventional PSO does not have any mutation operation; the swarms update their positions by sharing the position information among each others. The proposed mutation operation is called multi-wavelet mutation, which applies the wavelet theory to enhance the convergence rate, the solution quality and stability. Wavelet is a tool to model seismic signals. It can be modeled by combining translations and dilations of an oscillatory function of a finite duration. The proposed multi-wavelet mutation is modified from a published wavelet mutation operation in which only one element in each particle may undergo the mutation process. On applying the published wavelet mutation, the searching space, though allowed to be varying during the evolution, is sometimes unnecessarily restricted. By introducing the proposed multi-wavelet mutation into the PSO, the performance of it is significantly improved, which is reflected in a suite of 18 benchmark test functions. Applications of the proposed PSO with multi-wavelet mutation to two problems are also discussed in this thesis. They are the applications on image restoration of color-quantized half-toned images and microarray gene data selection. Improvements to the results are found in these two application examples thanks to the proposed PSO.

1.6 THESIS ORGANIZATION

This thesis is organized into six chapters. A literature review on Particle Swarm Optimization will be given in Chapter 2. A brief account of the applications of Support Vector Machine, image restoration, and micro-array will also be given in this chapter.

An improved hybrid Particle Swarm Optimization (PSO) that incorporates a wavelet-based mutation operation is proposed in Chapter 3. It applies the wavelet theory to enhance PSO for providing more freedom to particles in exploring the solution space more effectively for better solutions. A suite of benchmark test functions are employed to evaluate the performance of the proposed method.

Chapter 4 presents an application example of the proposed Particle Swarm Optimization on image restoration. Restoration of color-quantized images is rarely addressed in the literature, especially when the images are color-quantized with half-toning. The proposed PSO with multi-wavelet mutation will be used to restore half-toned color-quantized images. The signal-to-noise ratio improvement (SNRI) and the simulation results will be presented to illustrate the restoration ability of the proposed methods.

Chapter 5 presents another application example of the proposed Particle Swarm Optimization on micro-array data selection. DNA microarray studies produce a large amount of data, which are highly redundant and noisy, and only a fraction of genes may present distinct profiles for different classes of samples. An integrated

approach of Support Vector Machine (SVM) and Particle Swarm Optimization (PSO) is proposed in this chapter for this issue**.** Training of the parameters for a Support Vector Machine using the proposed PSO with multi-wavelet mutation will be discussed, and the trained SVM will be used to classify the gene features for the diagnosis of cancers in order to illustrate the ability of the proposed method.

Chapter 6 gives a conclusion to the materials covered in this thesis. The directions for further research will also be discussed.

CHAPTER 2

LITERATURE REVIEW

A review on Particle Swarm Optimization with some recent improvements to the method will be given in this chapter. A review on image restoration and gene expression will also be provided. The advantage and disadvantage of them will be discussed.

2.1 INTRODUCTION TO PARTICLE SWARM OPTIMIZATION

2.1.1 General Description

Particle Swarm Optimization (PSO) is a recently proposed population-based stochastic optimization algorithm [J. Kennedy, 1995]. It models the processes of the sociological behavior associated with bird flocking, and is one of the evolutionary computation techniques. Comparing with other population based stochastic optimization methods, PSO has comparable or even superior search performance for many hard optimization problems with a faster and more stable convergence rate [J. Kennedy, 2001]. PSO has been used in different industrial areas, such as power systems, parameter learning of neural networks, control, prediction, modeling, etc. It

is particularly useful for complex systems with a large number of parameters and the analytical solutions are difficult to find.

```
Procedure standard PSO
Begin
        t→0             //iteration number
        initialize X(t) //X(t) be the Swarm for iteration t
        evaluate f(X(t)) //f(·) be the fitness function
while (not termination condition) do
        begin
                t→t+1
                // Process of standard PSO//
                record the best current position among all particles as gbest
                record the best current position for each particle as pbest
                update velocity v(t) and position of each particle based on gbest and pbest
                if v(t) > v_max
                        v(t) = v_max
                end
                if v(t) < -v_max
                        v(t) = -v_max
                end
                // End of the process of standard PSO//
                reproduce a new X(t)
                evaluate f(X(t))
        end
end
```

Fig. 2.1 Pseudo code for standard PSO

The process of the standard PSO is shown in Fig. 2.1. Each particle of a swarm $X(t)$ is first initialized.

$$X(t) = \left[\mathbf{x}^1(t), \quad \mathbf{x}^2(t), \quad \ldots \quad , \mathbf{x}^\gamma(t)\right] \tag{1.1}$$

$$\mathbf{x}^p(t) = \left[x_1^p(t), \quad x_2^p(t), \quad \ldots \quad , x_\kappa^p(t)\right] \tag{1.2}$$

where p = 1, 2, ..., γ and γ denotes the number of particles in the swarm. Each particle **x** contains κ elements and occupies a position in the searching space.

The swarm evolves from generation *t* to *t*+1 by repeating the following procedures.

(1) Since the fitness value of each particle is calculated in each generation, the position of the particle with the best fitness value among all the particles will be recorded as *gbest*; also the best position of each particle is recorded as *pbest*.

(2) Let the velocity $v_j^p(t)$ be the flight speed of the *j*-th element of the *p*-th particle in a search space, the value of this velocity will be updated based on *gbest* and *pbest*, such that each particle will move to a new position.

(3) The particles will be replaced by the new ones, and form a new set of swarm if the fitness value of the old particle is poorer than that of the new one.

2.1.2 Drawbacks of the standard PSO

There are some drawbacks for the standard PSO (SPSO). SPSO converges sharply in the early stage of the searching process; however, it is easily trapped in local optima in the later stage. It behaves like the traditional local searching methods that trap in the local optima. As a result, it is hard to obtain any significant improvement by examining neighboring solutions in the later stage of the search. The behavior of SPSO is affected by some important aspects related to the velocity update. If a particle's current position coincides with the global best position, the particle will only move away from this point if its inertia weight and velocity are different from zero. If their velocities are very close to zero, all the particles will stop moving once they catch up with the global best particle, which may lead to a premature

convergence and no further improvement can be obtained. This phenomenon is known as stagnation.

2.2 RECENT HYBRID PSO AND ITS LIMITATIONS

Much research effort has been spent for improving the performance of PSO. Recently, different hybrid PSOs have been proposed to overcome the drawback of trapping in the local optima. A hybrid PSO was first proposed in 1998, in which a standard selection mechanism is integrated with the PSO [P. Angeline, 1998]. A new hybrid gradient descent PSO (HGPSO), which integrates with some gradient information to achieve faster convergence without getting trapped in local minima, was proposed by Noel and Jannett [M.M. Noel, 2004]. Juang proposed a hybrid PSO algorithm named HGAPSO, which incorporates the Genetic Algorithm's evolutionary operations of crossover, mutations, and reproduction into the standard PSO [C.F. Juang, 2004]. Ahmed *et al* proposed a hybrid PSO named HPSOM, in which a constant mutating space is used for mutations [A.A.E. Ahmed, 2005].

More details about the aforementioned hybrid PSOs are given as follows.

2.2.1 PSO with a selection mechanism

To improve the performance of the standard PSO, Angeline developed a selection operation for the PSO. This hybrid PSO is similar to the original PSO except it incorporates a tournament selection method into it.

In each iteration process, the fitness value of each particle will be calculated. The fitness of each particle is compared with the fitness of the other particles, such that all the particles are sorted based on the fitness value, and the particle with the highest fitness value appears at the head of the population. The position and the

velocity of the worst half of the population will be replaced by the position and the velocity of the best half of the population; however, the individual information *pbest* will remain unchanged.

By using this approach, half of the particles will be removed from the searching space, meaning that half of the position information will be lost permanently. This might not be a good approach for complex functions, such as those functions with many local minima. If a particle with a high fitness value has already trapped in a local minimum, this approach might not be able to drive it to escape from the local minimum. Even all the particles may move toward this local minimum.

2.2.2 PSO with gradient descent

The classical gradient descent rule can be described by the following equation:

$$\mathbf{x}(t+1) = \mathbf{x}(t) - \gamma \nabla f(\mathbf{x}(t)) \tag{1.3}$$

where $f(\cdot)$ is the fitness function and γ is the step ratio or the learning rate. The gradient descent algorithm works well for the functions with no local minimum; however, if multimodal functions are involved, the gradient descent rule might lead to the particles converging to local minima only. In order to help the searching to escape from local minima, a hybrid PSO with integrated gradient descent had been developed by Noel and Jannett named HGPSO. The following update equation is used in HGPSO:

$$v_j^p(t) = v_j^p(t-1) + \begin{pmatrix} rand() \cdot \left(gbest_j - x_j^p(t-1)\right) + \\ \left(rand()/\varepsilon\right) \cdot \left(f\left(x_j^p(t-1) + \varepsilon \cdot E^p\right) - f\left(x_j^p(t-1)\right)\right) \end{pmatrix} \quad (1.4)$$

where ε is a small constant and E^p is the standard basis vector for the *p*-th particle. HGPSO is similar to the standard PSO except that it discards the local information *pbest* and replaces it by the gradient. Although this approach can achieve fast convergence without getting trapped in the local minima, it discards the local information which might be helpful for the searching. Also, the computational demand of HGPSO is increased by the process of the gradient descent.

2.2.3 PSO with genetic operation

Ahmed *et al* proposed to integrate GAs' mutation operation into PSO, which aids to overcome *stagnation*. In this thesis, we refer to this hybrid PSO as APSO. The mutation operation starts with a randomly chosen particle in the swarm and moves to different positions inside the search area. The following mutation operation is used in APSO:

$$mut(x_j) = x_j - \omega \quad (1.5)$$

where x_j is the randomly chosen particle element from the swarm, and ω is a number randomly generated within the range $\left[0, \quad 0.1 \times \left(para_{\max}^j - para_{\min}^j\right)\right]$ representing 10% of the length of the search space. $para_{\max}^j$ and $para_{\min}^j$ are the upper and lower

boundaries of each particle element respectively. The mutation on particles will be performed after updating the velocities and positions of the particles. It can be seen that the PSO and APSO are identical except the mutation operation has been integrated in the second method.

However, it can be noticed from above that the mutating space in APSO is limited by ω, which is 10% of the range of the parameter x_j (i.e. $0.1\times\left(para^{j}_{\max} - para^{j}_{\min}\right)$). It may not be a good approach to fix the mutating space at all the time of the searching. The approach can be further improved by employing a dynamic mutation operation in which the mutating space varies during the search.

2.3 A REVIEW ON IMAGE RESTORATION

Color quantization [M.T. Orchard, 1991] is a process that reduces the number of colors required to represent a digital image. The main purpose of color quantization is to reduce the file size for limited bandwidth transmission and the data storage. However, despite the data loss after the quantization, it is often necessary to recover the quantized image back to the original one in greater detail. Image restoration is another topic in image processing. In particular, the restoration of color-quantized images is rarely addressed in the literature, especially when the images are color-quantized with half-toning [R. Ulichney, 1987]. Most of the work concerns mainly on the restoration of noisy and blurred images [M. Barni, 2000] [G. Angelopoulosnd, 1994] [N.P. Galatsanos, 1991] [B.R. Hunt, 1984] [H. Altunbasak, 2001] [K.J. Boo, 1997] [N.P. Galatsanos, 1989] [N.P. Galatsanos, 1991].

Fung proposed an algorithm that made use of Simulated Annealing for half-toned image restoration [Y.H. Fung, 2006], and compared it with 6 other image restoration methods, namely Galatsanos's algorithm [N.P. Galatsanos, 1991a], Hunt's algorithm [B.R. Hunt, 1984], Altunbasak-IND, Altunbasak-KL [H. Altunbasak, 2001], Fung's algorithm [Y.H. Fung, 2004] and Mese's algorithm [M. Mese, 2001]. It is reported that Simulated Annealing provides a remarkable result in terms of signal to noise ratio improvement (SNRI). However, those methods for comparison, except the Mese's algorithm, are mainly for handling noisy and blurred color images.

2.4 A REVIEW ON GENE EXPRESSION CLASSIFICATION

Classification of tumor types based on genomic information is important, and plays a role for improving future cancer diagnosis and drug development. The main objective of the cancer classification is to target specific therapies to distinct tumor types, such that it can maximize the treatment efficiency and minimize the drug toxicity [T. R. Golub, 1999]. DNA microarray studies, which provide a powerful tool for the genomic studies of cancer, produce a large amount of data. However, the expression data might be highly redundant and noisy. An effective analytical method should been developed to sort out whether specific cancer sample have distinctive feature of gene expression over normal samples.

Many classification methods have been employed to solve the above problem. For example, the compound covariate predictor has been discussed in [M.D. Radmacher, 2002]. The class discovery procedure for classifying the cancer was presented in [T.R. Golub, 1999]. Gene feature selection and prediction by employing neural networks were discussed in [F.Z. Brill, 1998] [J. Khan, 2001]. In [L. Li, 2001], the *k*-nearest neighbour concept with Genetic Algorithm was used to develop a classifier. Support Vector Machine as a tool for classifying the gene expression data is presented in [T.S. Furer, 2000].

CHAPTER 3

PARTICLE SWARM OPTIMIZATION WITH MULTI- WAVELET MUTATION

3.1 INTRODUCTION

Particle Swarm Optimization (PSO) is one of the evolutionary computation algorithms for tackling complex optimization problems [R. Eberhart, 1995]. Observations reveal that the traditional PSO converges sharply in the early stage of the searching process [J. Kennedy, 2001]; however, it easily traps in local optima in the later stage. Much research effort has been spent for improving the performance of PSO [R. Eberhart, 1995] [R.C. Eberhart, 2000] [B. Zhao, 2005] [S.H. Ling, 2007]. In general, every iteration step of the PSO algorithm involves one operation: updating the positions of the particles. Unlike other evolutionary algorithm, such as Genetic Algorithm and Evolutionary Programming, PSO does not have the selection operation and the mutation operation. The position update is used to share the information among all the particles, which is similar to the crossover operation of the Genetic Algorithm, and generates a new set of swarm.

In this chapter, an improved PSO with a proposed wavelet mutation will be introduced. The proposed mutation operation is called multi-wavelet mutation, which applies the wavelet theory to further improve a recently published wavelet mutation

operation [I. Daubechies, 1992]. Wavelet is a tool to model seismic signals. It can be described by combining translations and dilations of an oscillatory function within a finite duration. A continuous-time function can be called a wavelet function if it satisfies two properties: 1) the time integration of this function is equal to 0, or the total positive momentum is equal to the total negative momentum; and 2) the energy associated with this function is confined to a finite duration and is bounded. In a recently published PSO with wavelet mutation (WPSO), we notice that only one element in each particle may undergo the mutation process in an iteration step. This may pre-maturely restrict the searching space, although the searching space has been varied during the searching process. A multi-wavelet mutation is proposed in this thesis such that more than one element in each particle can mutate in each iteration step. The performance of the PSO is found to be improving in terms of the solution quality and stability. Also, a faster rate of convergence can be reached. Experimental studies of applying the proposed PSO with multi-wavelet mutation to a suite of 18 benchmark test functions are to be conducted. The searching ability of the proposed PSO will be evaluated and compared with the conventional methods.

This chapter is organized as follows. Section 3.2 presents the proposed hybrid PSO with multi-wavelet mutation. Details about the experimental studies for the 18 benchmark test functions and the results of the performance evaluation will be given in Section 3.3. A conclusion will be drawn in Section 3.4.

3.2 PSO WITH MULTI-WAVELET MUTATION

3.2.1 Standard Particle Swarm Optimization (SPSO)

PSO is a novel optimization method developed by Eberhart *et al.* [R. Eberhart, 1995]. It models the processes of the sociological behavior of bird flocking, and is one of the important evolutionary computation techniques. Within a number of particles that constitute a swarm, each particle traverses the search space looking for the global optimum. The standard PSO (SPSO) process is shown in Fig. 2.1. First, a swarm of particles is created randomly, and each particle **x** contains κ elements representing a position in the searching space.

$$\mathbf{x}^p(t) = \left[x_1^p(t), \quad x_2^p(t), \quad \ldots \quad , x_\kappa^p(t)\right] \tag{3.1}$$

where p = 1, 2,... , γ and γ denotes the number of particles in the swarm. Second, each particle is evaluated by a defined fitness function. The objective of SPSO is to minimize the fitness value of a particle through iteration steps; hence, the best particle should return the smallest fitness value. Third, the best ever position of each particle is recorded as *pbest*, and the position of the best particle among all the particles is recorded as *gbest*. All the particles will update their positions based on the values of *pbest* and *gbest*. The velocity $v_j^p(t)$ (being the flight speed in the search space) and the position $x_j^p(t)$ of the j-th element of the p-th particle at the t-th generation are related by the following equations:

$$v_j^p(t) = \begin{pmatrix} w \cdot v_j^p(t-1) \\ +\varphi_1 \cdot rand() \cdot \left(pbest_j^p - x_j^p(t-1)\right) \\ +\varphi_2 \cdot rand() \cdot \left(gbest_j - x_j^p(t-1)\right) \end{pmatrix} \tag{3.2}$$

$$x_j^p(t) = x_j^p(t-1) + k \cdot v_j^p(t) \tag{3.3}$$

where

$pbest^p = \left[pbest_1^p \quad pbest_2^p \quad ,... \quad pbest_\kappa^p\right]$, $gbest = \left[gbest_1 \quad gbest_2 \quad ,... \quad gbest_\kappa\right]$, j = 1,2, …, κ; w is an inertia weight factor, φ_1 and φ_2 are acceleration constants, *rand*() returns a uniformly random number in the range of [0,1], k is a constriction factor derived from the stability analysis of (3.2) to ensure the system to converge but not prematurely [R. C. Eberhart, 2000]. Mathematically, k is a function of φ_1 and φ_2 as reflected in the following equation:

$$k = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|} \quad \text{where } \varphi = \varphi_1 + \varphi_2 \text{ and } \varphi > 4. \tag{3.4}$$

SPSO utilizes *pbest* and *gbest* to modify the current search point to avoid the particles moving in the same direction, but to converge gradually toward *pbest* and *gbest.* A suitable choice of the inertia weight w provides a balance between global and local explorations. Generally, w can be dynamically set with the following equation [J. Kennedy, 2001]:

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{T} \times t \tag{3.5}$$

where t is the current iteration number, T is the total number of iteration, $w_{\max}$ and $w_{\min}$ are the upper and lower limits of the inertia weight, and are set to 1.2 and 0.1 respectively in this thesis. In (3.2), the particle velocity is limited by a maximum value $v_{\max}$. The parameter $v_{\max}$ determines the resolution of regions between the present position and the target position to be searched. This limit enhances the local exploration of the problem space, and it realistically depicts the incremental changes of human learning. If $v_{\max}$ is too large, particles might fly past good solutions. If $v_{\max}$ is too small, particles may not explore sufficiently beyond local solutions. From many experiments of PSO, $v_{\max}$ was often set at 10% - 20% of the dynamic range of the variables on each dimension.

After estimating a new position, each particle will evaluate the fitness value again based on the defined fitness function. If the fitness value of a particle is better than its previous one, the particle's position will be updated; otherwise, its position will be kept unchanged. Finally a new population is formed. The new population will repeat the same process until the termination condition is met.

```
Procedure of hybrid PSO with mutation
Begin
        t→0                 //iteration number
        initialize X(t)     //X(t) be the Swarm for iteration t
        evaluate f(X(t))    //f(·) be the fitness function
while (not termination condition) do
        begin
                t→t+1
                perform standard PSO process (shown in Fig 2.1)
                perform mutation operation with μ_m
                reproduce a new X(t)
                evaluate f(X(t))
        end
end
```

Fig 3.1. Pseudo code for PSO with mutation.

3.2.2 Multi-Wavelet Mutation

3.2.2.1 Wavelet

Some seismic signals can be modeled by combining translations and dilations of an oscillatory function with a finite duration called a "wavelet". A continuous function $\psi(x)$ is called a "mother wavelet" or "wavelet" if it satisfies the following properties:

Property 1:

$$\int_{-\infty}^{+\infty}\psi(x)dx = 0 \qquad (3.6)$$

In other words, the total positive momentum of $\psi(x)$ is equal to the total negative

momentum of $\psi(x)$.

Property 2:

$$\int_{-\infty}^{+\infty}|\psi(x)|^2 dx < \infty \tag{3.7}$$

where most of the energy in $\psi(x)$ is confined to a finite duration and bounded. The Morlet wavelet [2] as shown in Fig. 3.2 is an example mother wavelet:

$$\psi(x) = e^{-x^2/2}\cos(5x) \tag{3.8}$$

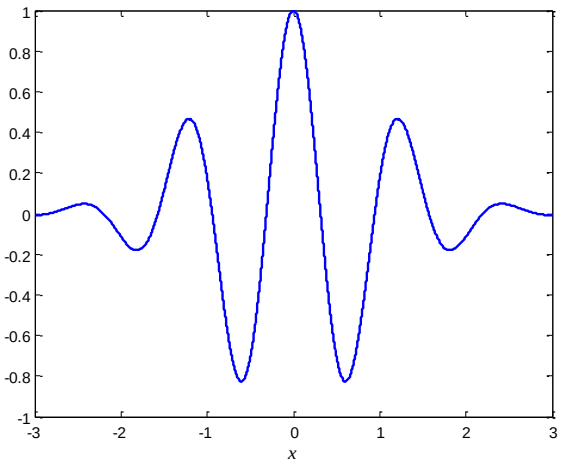


Fig. 3.2. Morlet wavelet.

The Morlet wavelet integrates to zero (*Property 1*). Over 99% of the total energy of the function is contained in the interval of $-2.5 \le x \le 2.5$ (*Property 2*).

In order to control the magnitude and the position of $\psi(x)$, a function $\psi_{a,b}(x)$ is defined as follows.

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right) \tag{3.9}$$

where a is the dilation parameter and b is the translation parameter. Notice that $\psi_{1,0}(x) = \psi(x)$. As

$$\psi_{a,0}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x}{a}\right), \tag{3.10}$$

it follows that $\psi_{a,0}(x)$ is an amplitude-scaled version of $\psi(x)$. Fig. 3.3 shows different dilations of the Morlet wavelet. The amplitude of $\psi_{a,0}(x)$ will be scaled down as the value of the dilation parameter a increases. This property is used to do the mutation operation in order to enhance the searching performance.

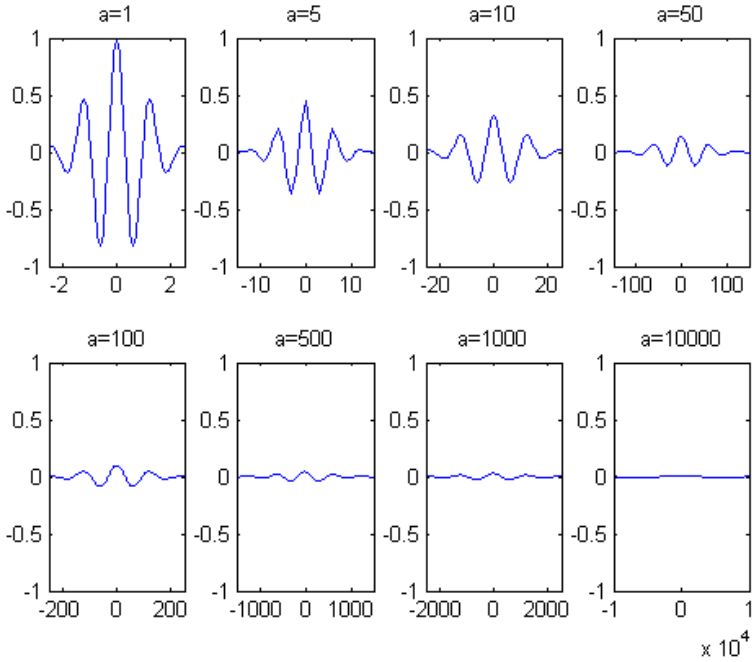


Fig. 3.3. Morlet wavelet dilated by different values of the parameter a (x-axis: x, y-axis: $\psi_{a,0}(x)$.)

3.2.2.2 Operation of multi-wavelet mutation

We propose the algorithm of Particle Swarm Optimization with multi-wavelet mutation (MWPSO) that varies the mutating space based on the wavelet theory. Under this algorithm, the number of elements in each particle that undergo the mutation process can be controlled (Fig. 3.4).

3.2.2.2.1. MWPSO and its operation

```
Procedure of hybrid PSO with proposed wavelet
Begin
        t→0              //iteration number
        initialize X(t)  //X(t) be the swarm for iteration t
        evaluate f(X(t)) //f(·) be the fitness function
while (not termination condition) do
        begin
                t→t+1
                Perform standard PSO process (shown in Fig 2.1)
                Perform mutation operation with μ_m and N_m
                reproduce a new X(t)
                evaluate f(X(t))
        end
end
```

Fig 3.4. Pseudo code for PSO with mutation.

The mutation operation mutates the elements of particles. In general, various methods like uniform mutation or non-uniform mutation [Z. Michalewicz, 1994] [A. Neubauer, 1997] can be employed to realize the mutation operation. It is found that if only one element in each particle can undergo the mutation process in each iteration step (Fig. 3.1), the searching space may become unnecessarily restricted. We propose the multi-wavelet mutation (WM) operation that exhibits a fine-tuning ability. The details of the operation are as follows.

Every particles of the swarm will have a chance to mutate governed by a probability of mutation $\mu_m \in [0 \quad 1]$, which is defined by the user. For each particle, a random number between 0 and 1 will be generated; if μ_m is larger than the random number, this particle will be selected for the mutation operation.

Another parameter called the element probability $N_m \in [0 \quad 1]$, which is also defined by the user, is introduced to control the number of elements in the particle that

will mutate in each iteration step. For instance, if $\mathbf{x}^p(t)=\left[x_1^p(t),\quad x_2^p(t),\quad \ldots\quad ,x_\kappa^p(t)\right]$ is the selected *p*-th particle, the expected number of elements that undergo mutation is controlled by this equation:

$$\text{Expected number of mutated elements} = N_m \times \kappa \tag{3.11}$$

The exact elements for doing mutation in a particle are randomly selected. The resulting particle is given by $\bar{\mathbf{x}}^p(t)=\left[\bar{x}_1^p(t+1),\quad \bar{x}_2^p(t+1),\quad \ldots\quad ,\bar{x}_\kappa^p(t+1)\right]$. If the *j*-th element is selected for mutation, the operation is given by

$$\bar{x}_j^p(t)=\begin{cases} x_j^p(t)+\sigma\times\left(para_{\max}^j - x_j^p(t)\right) \text{ if } \sigma > 0 \\ x_j^p(t)+\sigma\times\left(x_j^p(t) - para_{\min}^j\right) \text{ if } \sigma \le 0 \end{cases}, \tag{3.12}$$

$$\sigma = \psi_{a,0}(\varphi) \;= \frac{1}{\sqrt{a}}\psi\left(\frac{\varphi}{a}\right) \tag{3.13}$$

By using the Morlet wavelet in (3.8) as the mother wavelet,

$$\sigma = \frac{1}{\sqrt{a}} e^{-\left(\frac{\varphi}{a}\right)^2/2} \cos\left(5\left(\frac{\varphi}{a}\right)\right) \tag{3.14}$$

If σ is positive ($\sigma > 0$) approaching 1, the mutated element's value will tend to the maximum value of $x_j^p(t)$. Conversely, when σ is negative ($\sigma \le 0$) approaching –1,

the mutated element's value will tend to the minimum value of $x_j^p(t)$. A larger value of $|\sigma|$ gives a larger searching space for $x_j^p(t)$. When $|\sigma|$ is small, it gives a smaller searching space for fine-tuning. Referring to *Property 1* of the wavelet, the total positive momentum of the mother wavelet is equal to the total negative momentum of the mother wavelet. Then, the sum of the positive σ is equal to the sum of the negative σ when the number of samples is large and φ is randomly generated:

$$\frac{1}{N}\sum_{N}\sigma = 0 \text{ for } N \to \infty, \tag{3.15}$$

where N is the number of samples. Hence, the overall positive mutation and the overall negative mutation throughout the evolution are nearly the same. This property gives better solution stability (smaller standard deviation of the solution values upon many trials). As over 99% of the total energy of the mother wavelet function is contained in the interval [–2.5, 2.5], φ can be generated from [–2.5, 2.5] randomly. The value of the dilation parameter a is set to vary with the value of t/T in order to meet the fine-tuning purpose, where T is the total number of iteration and t is the current number of iteration. In order to perform a local search when t is large, the value of a should increase as t/T increases so as to reduce the significance of the mutation. Hence, a monotonic increasing function governing a and t/T is proposed as follows.

$$a = e^{-\ln(g)\times\left(1-\frac{t}{T}\right)^{\zeta_{wm}}+\ln(g)} \tag{3.16}$$

where ζ_{wm} is the shape parameter of the monotonic increasing function, g is the upper limit of the parameter a. The effects of the various values of the shape parameter ζ_{wm} to a with respect to t/T are shown in Fig. 3.5. In this figure, g is set as 10000. Thus, the value of a is between 1 and 10000. Referring to (3.14), the maximum value of σ is 1 when the random number of φ =0 and a =1 (t/T = 0). Then referring to (3.12), the offspring element $\bar{x}_j^p(t) = x_j^p(t)+1\times\left(para_{\max}^j - x_j^p(t)\right) = para_{\max}^j$. It ensures that a large search space for the mutated element of particle is given. When the value t/T is near to 1, the value of a is so large that the maximum value of σ will become very small. For example, at t/T = 0.9 and ζ_{wm} =1, the dilation parameter a = 400; if the random value of φ is zero, the value of σ will be equal to 0.0158. With $\bar{x}_j^p(t) = x_j^p(t)+0.0158\times\left(para_{\max}^j - x_j^p(t)\right)$, a smaller searching space for the mutated element of particle is given for fine-tuning.

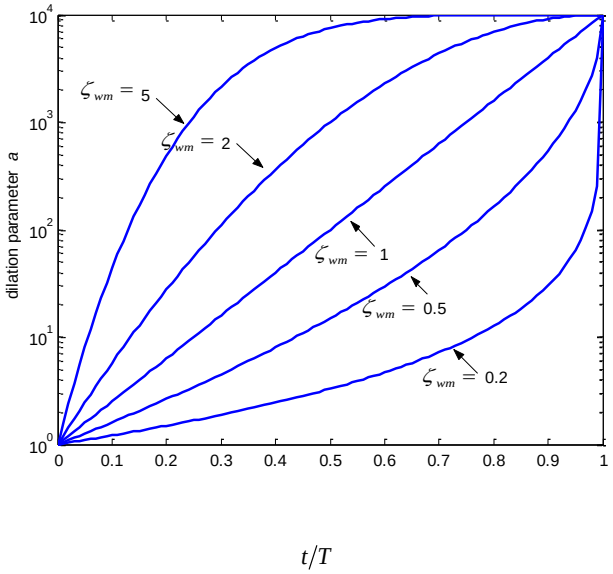


Fig. 3.5. Effect of the shape parameter ζ_{wm} to a with respect to t/T.

After the operation of wavelet mutation, a new swarm is generated. This new swarm will repeat the same process. Such an iterative process will be terminated when a defined number of iteration is met.

3.2.3 How Wavelet Improve Performance

All the wavelet functions exhibit two properties. 1) The total momentum of the function is equal to zero, or the positive momentum is equal to the negative momentum. This property can enhance the mutation operation such that the direction of mutation is not in one direction; yet the performance of the mutation is quite uniform on repeated experiments. 2) The amplitude of the wavelet function is not fixed, but is varying in a domain interval. All the selected particles will mutate in different directions to different degrees, which can increase the chance for the particles to reach the global optimal point. By making use of these 2 properties, the stability of the mutation operation is improved. The solution stability is reflected by the standard deviation of the solution.

Apart from the above, the amplitude of the wavelet function can be adjusted by controlling the dilation parameter. The fine-tuning effect to the mutation operation can be realized by decreasing the amplitude of the wavelet function so as to reduce the searching space when the number of iteration increases. Therefore, the solution quality can be improved with a higher convergence rate.

3.3 SIMULATION OF BENCHMARK TEST FUNCTIONS

3.3.1 Parameters Used

To do the simulation for evaluating the performance of the proposed method, some control parameters' values have to be determined: the probability of mutation (μ_m), the swarm size, the shape parameter of the wavelet function (ζ_{wm}), and the mutation probability of the particle elements. They affect the performance of the PSO, and suitably selecting their values is necessary. Some remarks about the choices of the parameter values in the simulations are given below.

i) Probability of mutation μ_m

Every particle of the swarm will have a chance to mutate governed by a probability of mutation $\mu_m \in [0 \quad 1]$, which is defined by the user. If μ_m is larger than a randomly generated number between 0 and 1, the corresponding particle will be selected for the mutation process. A larger value of μ_m will lead to more particles to be selected for doing the mutation operation.

ii) Swarm size

The swarm size is defined at the beginning of the simulation. By increasing the swarm size, the diversity of the searching space is enhanced, which can increase the chance of the PSO to reach the global optimum point. However, a longer computation time has to be traded off.

iii) Shape parameter of the wavelet function ζ_{wm}

The shape parameter of the wavelet function is used to change the characteristics of the monotonic increasing function governing the extent of mutation. It is chosen by trial and error, depending on the kind of the optimization problem. The value of the dilation parameter *a* is set to vary with the iteration number and ζ_{wm} in order to meet the fine-tuning purpose. If the value of ζ_{wm} increases, the extent of mutation will diminish more quickly at the early stage of evolution.

iv) Element probability N_m

This parameter controls the number of elements in a particle that will mutate in each iteration step. When more elements in each particle mutate, more freedom will be given to the particle to explore the searching space. It should be noted that the PSO will become a random search if the value of N_m and μ_m are both equal to one.

3.3.2 The benchmark test functions

A suite of eighteen benchmark test functions are used to test the performance of the proposed MWPSO. Many different kinds of optimization problems are covered by these benchmark test functions so that the proposed method is not biased to some selected problems, and the searching ability of the proposed method can be tested comprehensively. The benchmark test functions can be divided into three types. The first type covers unimodal functions, which has a single minimum. Functions f_1 to f_7

are unimodal functions. The second type covers multimodal functions with a few local minima. Functions f_8 to f_{13} belong to this type. The last type covers multimodal functions with many local minima. Functions f_{14} to f_{18} belong to this type. The details of these eighteen test functions are given in Appendix.

3.3.3 Experimental Setup

In addition to the proposed MWPSO, three other optimization algorithms are employed for comparison experimentally. These algorithms are the standard PSO (SPSO), APSO and PSO with wavelet mutation (WPSO). SPSO is the conventional PSO. APSO [A.A.E. Ahmed, 2005] is the PSO that integrates with genetic algorithm mutation operation, which aids to overcome stagnation. WPSO employs wavelet mutation, but only one element will be mutated in each mutation process. To do the simulations, the following conditions are used:

- The shape parameter of wavelet mutation (ζ_{wm}): For comparison, it is set to 0.2 for all functions
- The acceleration constant φ_1: 2.05
- The acceleration constant φ_2: 2.05
- Maximum velocity $v_{\max}$: 0.2
- Swarm size: 40
- Number of runs: 50
- Probability of mutation (μ_m): 0.1
- Element probability (N_m): 0.3
- Initial population: Generated uniformly at random

3.3.4. Result and Analysis

In this section, the simulation results for the 18 benchmark test functions are given to show the performance of the MWPSO. The experimental results in terms of mean fitness value, best fitness value, standard deviation and convergence rate will be given in section 3.3.4.1.

In addition to evaluating the number of iteration for each algorithm to reach the optimal point, an extra simulation for time consumption have been taken, and the simulation results will be given in section 3.3.4.2.

3.3.4.1 Simulation results

3.3.4.1.1 Unimodal function

Functions f_1 to f_7 are unimodal functions. Fig. 3.6 shows the convergence under different PSOs. The experimental results in terms of mean fitness value, standard deviation and best fitness value for f_1 to f_7 are shown in Table. 3.1.

Function f_1 is a sphere model. In view of the characteristic of f_1, which is smooth and symmetric, the main propose of employing this function is to measure the convergence rate of searching. It is probably the most widely used test function for this purpose. From the results in terms of mean fitness value and best fitness value, the MWPSO performs better than APSO and SPSO. Also, the standard deviation is better, which means the searched solutions are more stable. As shown in Fig. 3.6a, the convergence rate of MWPSO is higher than that of WPSO, APSO and SPSO.

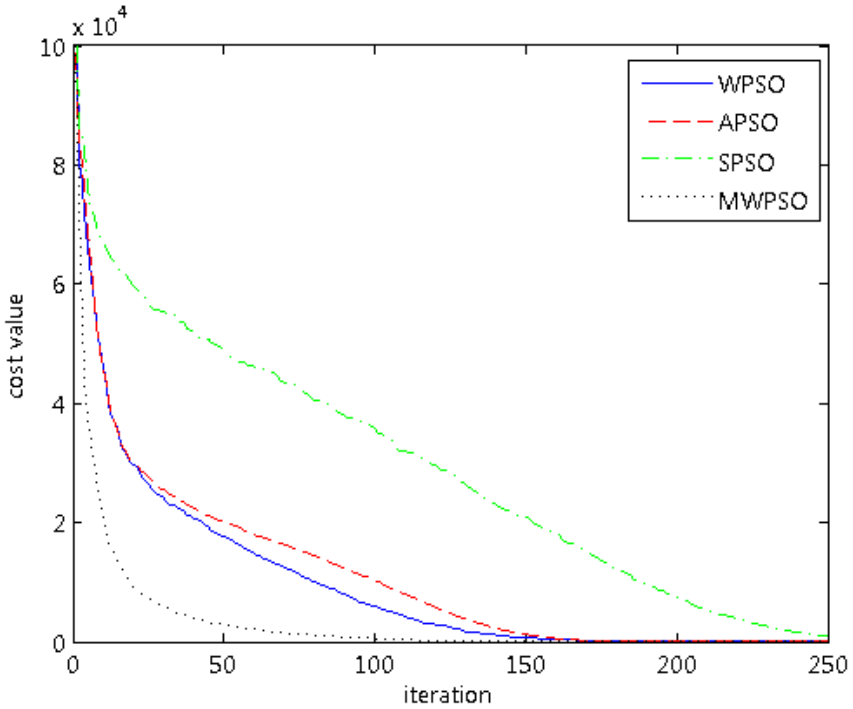
Function f_2 is the generalized Rosenbrock's function. The optimum point of it is located in a very narrow ridge. From Fig. 3.6a, we can see that all the algorithms are able to discover the optimal point with different convergence rate. Still, MWPSO can provide the fastest convergence than the other algorithms.

Function f_3 is a step function, which is a representative of flat surfaces. Flat surfaces are obstacles for optimization algorithms because they do not give any information about the search direction. This problem might be overcome if the algorithm has a variable step size. From Fig. 3.6b, it is clearly shown that MWPSO offers the best convergence rate over SPSO, WPSO and APSO. This indicates that by increasing the number of elements for mutations, we can enhance the searching space.

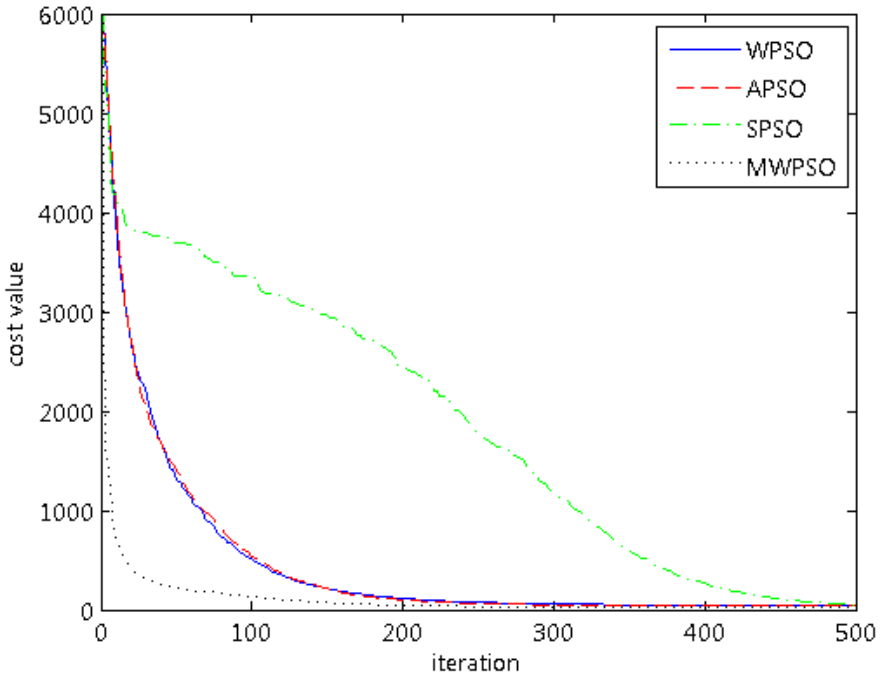
Function f_4 is a quartic function with Gaussian noise. Owing to the noise, the function never returns the same value for the same input parameter values. This increases the difficulty for searching the optimal point for all the algorithms. From Fig. 3.6b, we can see that the proposed algorithm offers the best convergence rate over the other algorithms at the first 30 times of iteration. The proposed method, WPSO and APSO all converge to the optimal point before 180 times of iteration.

Function f_5 is the Schwefel's problem 2.21. From Fig. 3.6c, the proposed algorithm provides the best convergence rate. Although Table 3.1 shows that the standard derivation offered by MWPSO is a little bit worse than that of WPSO, the mean fitness value and the best fitness value of MWPSO are the best. Thus, MWPSO gives a better solution quality.

Function f_6 is the Schwefel's problem 2.22 and function f_7 is the Easom's function. In these problems, MWPSO performs the best in terms of mean fitness value, standard deviation and best fitness value.

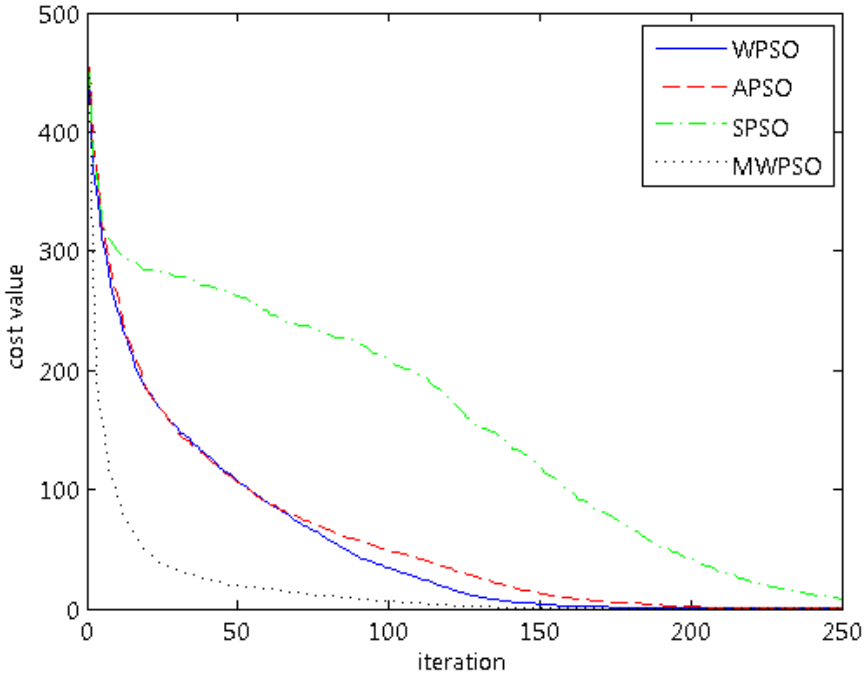


f_1

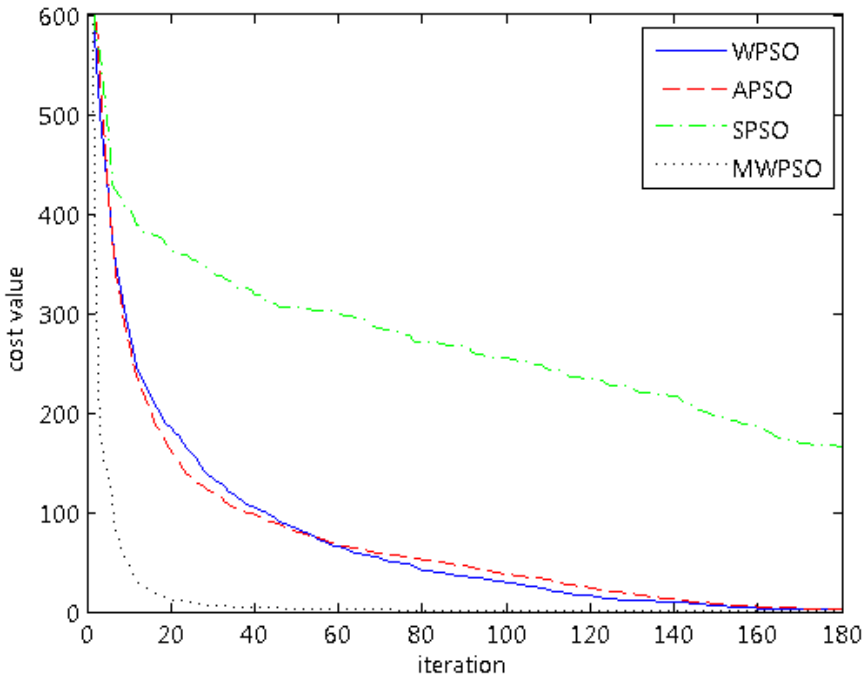


f_2

Fig. 3.6a. Comparisons between different hybrid PSOs for f_1 and f_2.

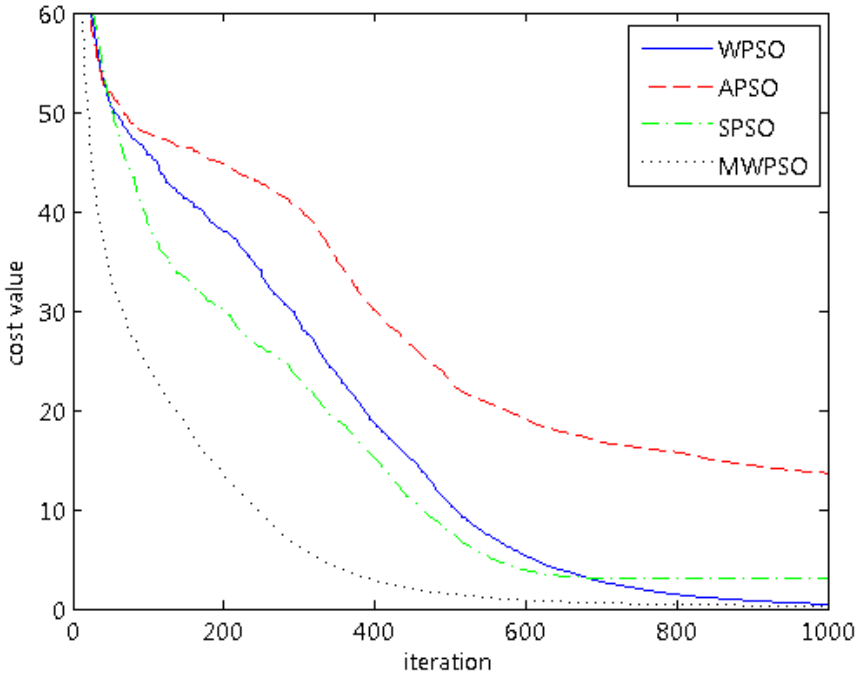


f_3

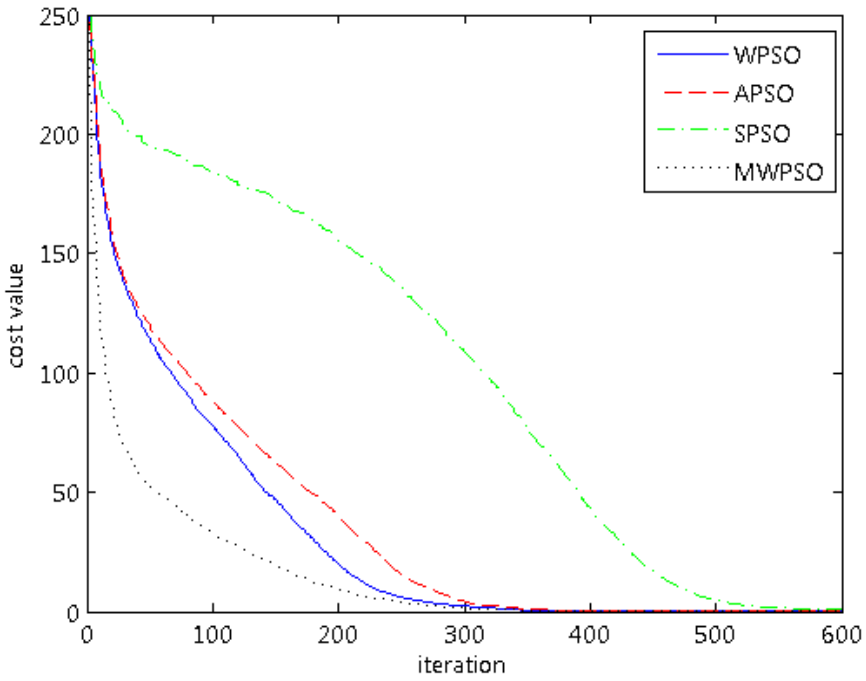


f_4

Fig. 3.6b. Comparisons between different hybrid PSOs for f_3 and f_4.

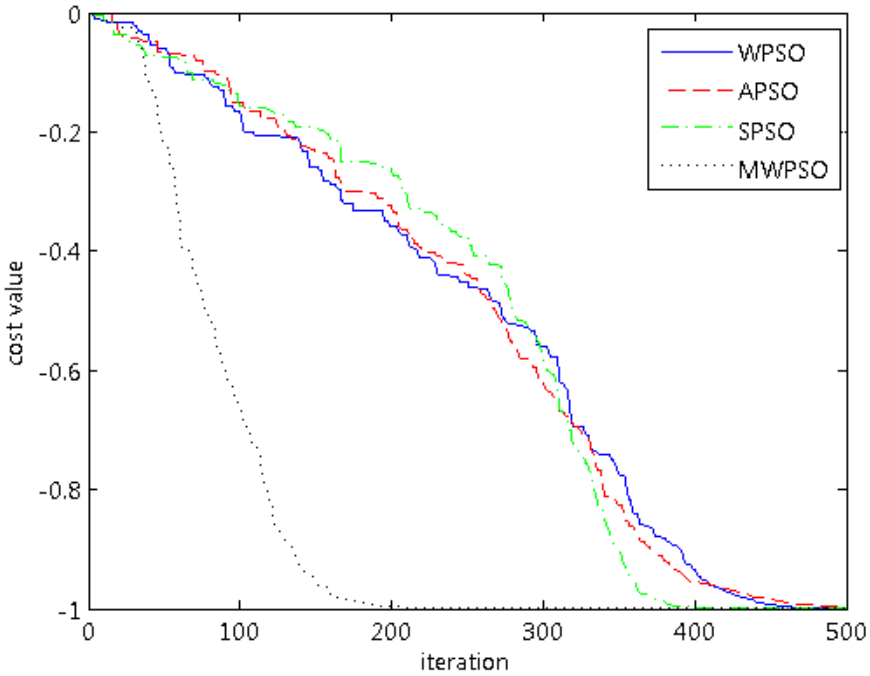


f_5



f_6

Fig. 3.6c. Comparisons between different hybrid PSOs for f_5 and f_6.



f_7

Fig. 3.6d. Comparisons between different hybrid PSOs for f_7.

f_1, number of iteration: 500				
	MWPSO	WPSO	APSO	SPSO
Mean	**0**	**0**	0.0002	3.1008
Best	**0**	**0**	0	0.0197
Std Dev	**0**	**0**	0.0001	4.4962
f_2, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**24.6688**	36.1977	31.9909	31.1844
Best	1.9512	**0.0350**	3.4805	18.9374
Std Dev	**7.3941**	23.9398	20.8112	16.1509
f_3, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0**	0	0	1.6000
Best	**0**	0	0	0
Std Dev	**0**	0	0	1.1606
f_4, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0.0121**	0.0174	0.0141	0.0415
Best	0.0050	0.0049	**0.0048**	0.0078
Std Dev	**0.0049**	0.0074	0.0069	0.0190
f_5, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0.2654**	0.4515	13.6634	3.0885
Best	**0.0535**	0.2589	0.1372	1.3772
Std Dev	0.2832	**0.0944**	2.11974	1.1097
f_6, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0**	**0**	**0**	0.4837
Best	**0**	**0**	**0**	0.0101
Std Dev	**0**	**0**	**0**	0.4927
f_7, number of iteration:1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**-1**	-1	-1	-1
Best	**-1**	-1	-1	-1
Std Dev	**0**	0	1.5294e-12	0

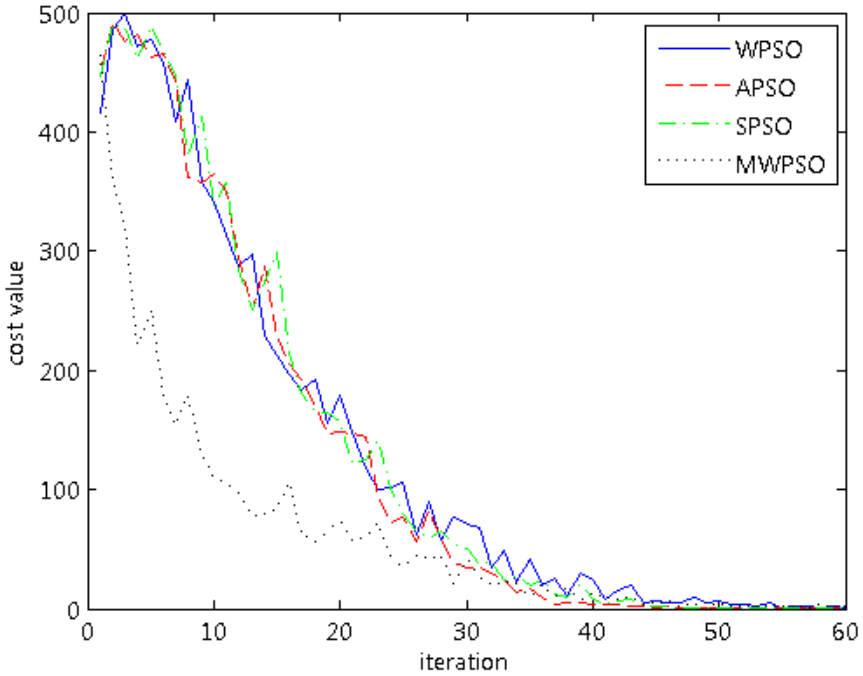
Table 3.1. Comparisons between different hybrid PSOs for f_1 to f_7.

3.3.4.1.2. Multimodal function with a few local minima

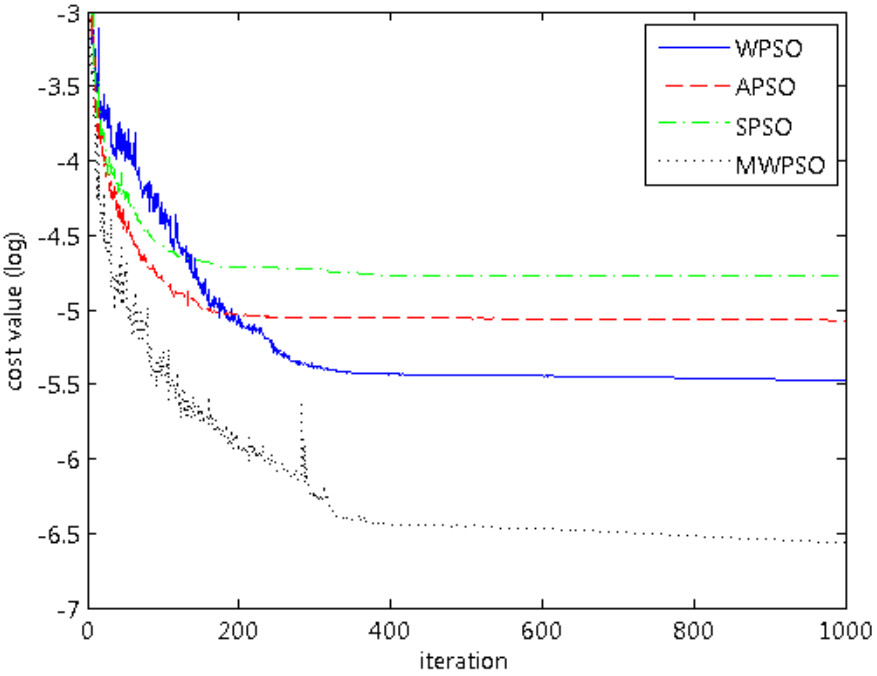
Functions f_8 to f_{13} are multimodal functions with a few local minima. The experimental results for all these functions are shown in Table 3.2. Fig. 3.7 shows the convergence rates for all the functions. Experimental results reveal that all the functions, except function f_9, do not bring any significant differences to the performance of the optimization algorithms; they all basically can reach the optimal points.

Function f_9 is a Kowalik's function. From Fig. 3.7a, on applying SPSO, APSO and WPSO, the optimizations are trapped in different local minimum. While MWPSO can successfully converge to the global optimum, the convergence rate of it is also higher than that of others. MWPSO can offer the best results in terms of fitness value and standard deviation.

Among all the functions in general, MWPSO provides the best values of standard deviation, meaning that the algorithm can offer the best solution stability.

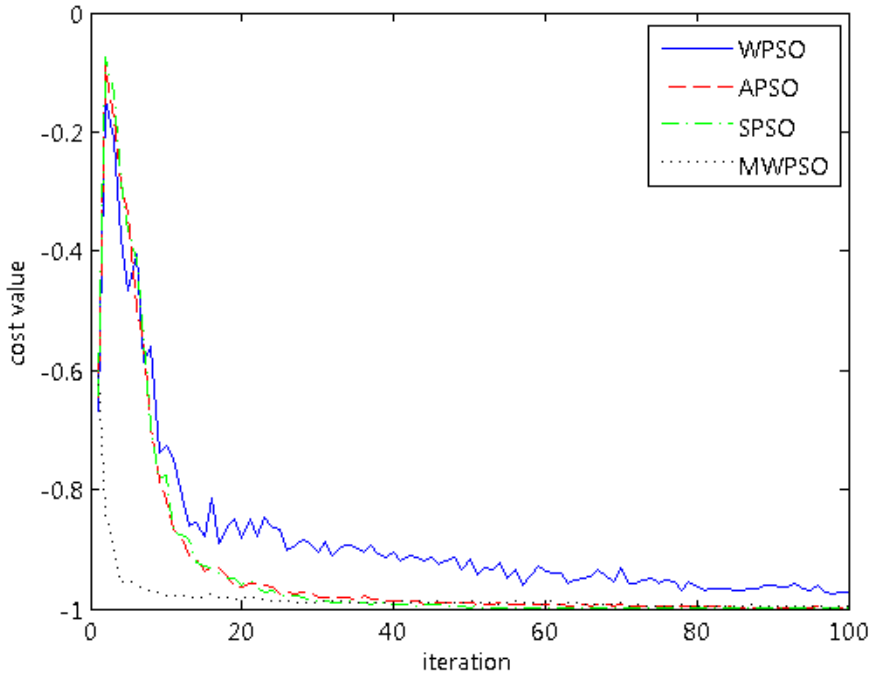


f_8

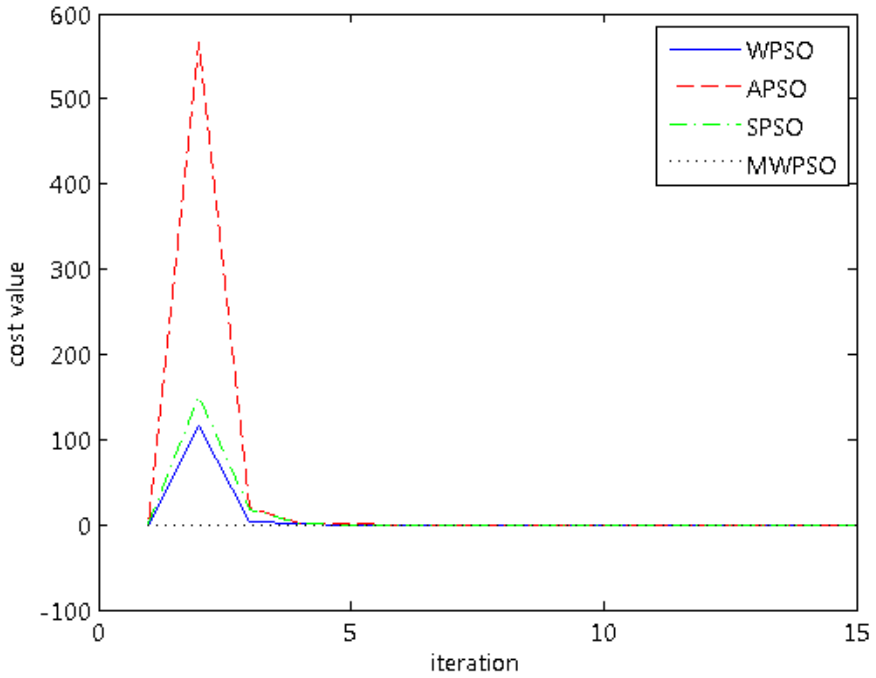


f_9

Fig. 3.7a. Comparisons between different hybrid PSOs for f_8 and f_9.

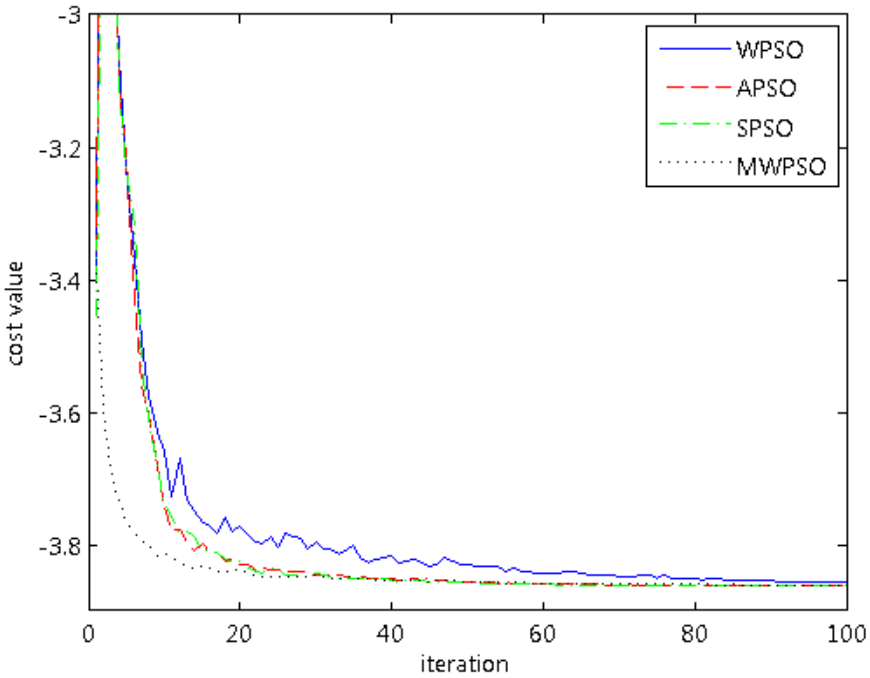


f_{10}

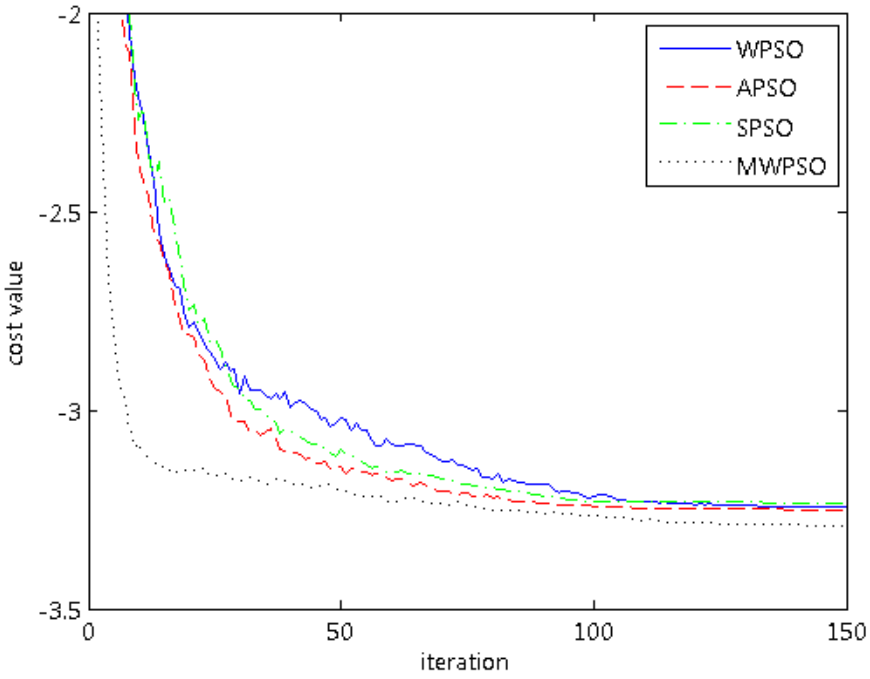


f_{11}

Fig. 3.7b. Comparisons between different hybrid PSOs for f_{10} and f_{11}.



f_{12}



f_{13}

Fig. 3.7c. Comparisons between different hybrid PSOs for f_{12} and f_{13}.

f_8, number of iteration: 500				
	MWPSO	WPSO	APSO	SPSO
Mean	**0.998**	0.998	0.998	0.998
Best	**0.998**	0.998	0.998	0.998
Std Dev	**0**	0	0	0
f_9 , number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0.0014**	0.0042	0.0063	0.0085
Best	**0.0003**	0.0004	0.0005	0.0003
Std Dev	**0.0039**	0.0077	0.0089	0.0093
f_{10}, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**-1**	-1	-1	-1
Best	**-1**	-1	-1	-1
Std Dev	**0**	1.5391×10^{-11}	5.0967×10^{-15}	0
f_{11}, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**-1.0316**	-1.0316	-1.0316	-1.0316
Best	**-1.0316**	-1.0316	-1.0316	-1.0316
Std Dev	**0**	0	0	0
f_{12}, number of iteration: 500				
	MWPSO	WPSO	APSO	SPSO
Mean	**-3.8628**	-3.8628	-3.8628	-3.8625
Best	**-3.8628**	-3.8628	-3.8628	-3.8628
Std Dev	**2.7683e-15**	3.5092e-11	2.7849e-14	0.0016
f_{13}, number of iteration: 500				
	MWPSO	WPSO	APSO	SPSO
Mean	**-3.2911**	-3.2459	-3.2507	-3.235
Best	**-3.322**	-3.322	-3.322	-3.322
Std Dev	**0.0527**	0.0576	0.0588	0.0903

Table 3.2. Comparisons between different hybrid PSOs for f_8 to f_{13}.

3.3.4.1.3. Multimodal function with many local minima

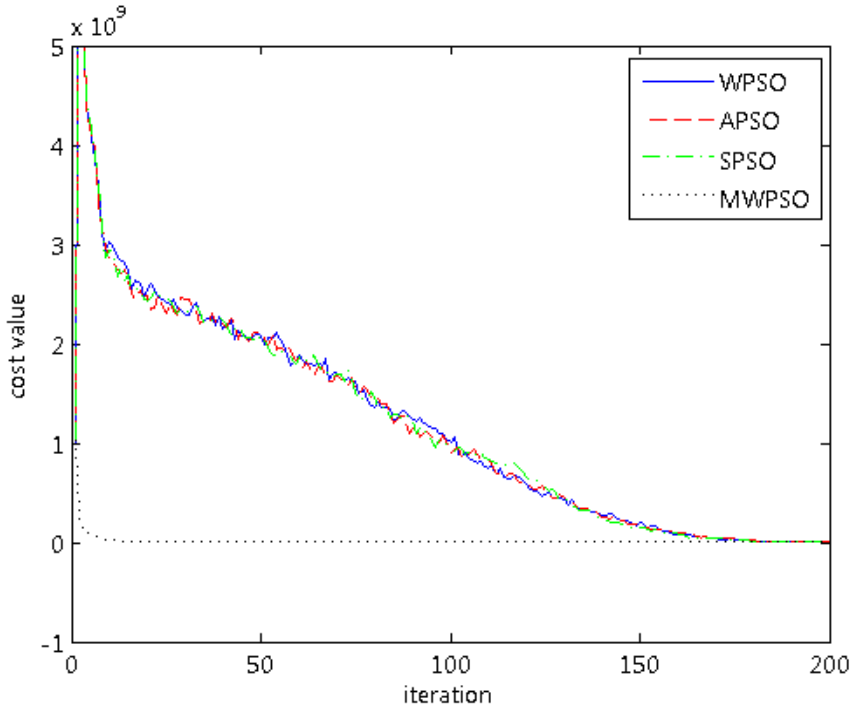
Functions f_{14} to f_{18} are multimodal function with many local minima. The experimental results for all these functions are shown in Table 3.3. The convergence rates of the algorithms for each function are shown in Fig. 3.8. Function f_{14} is a generalized penalized function. From the figure, only several times of iteration are necessary for the proposed MWPSO to reach the optimal point, while the other methods almost require 200 times of iteration. It clearly shows that MWPSO offers a good searching ability. Thanks to the multi-wavelet mutation, MWPSO is superior to the other three methods in terms of the mean fitness value and standard deviation.

Function f_{15} is the generalized Rastrigin's function. Although the proposed MWPSO cannot provide the best solution, the results in term of mean fitness value and standard deviation are much better than those of the other methods. The proposed method can provide more stable results than the others.

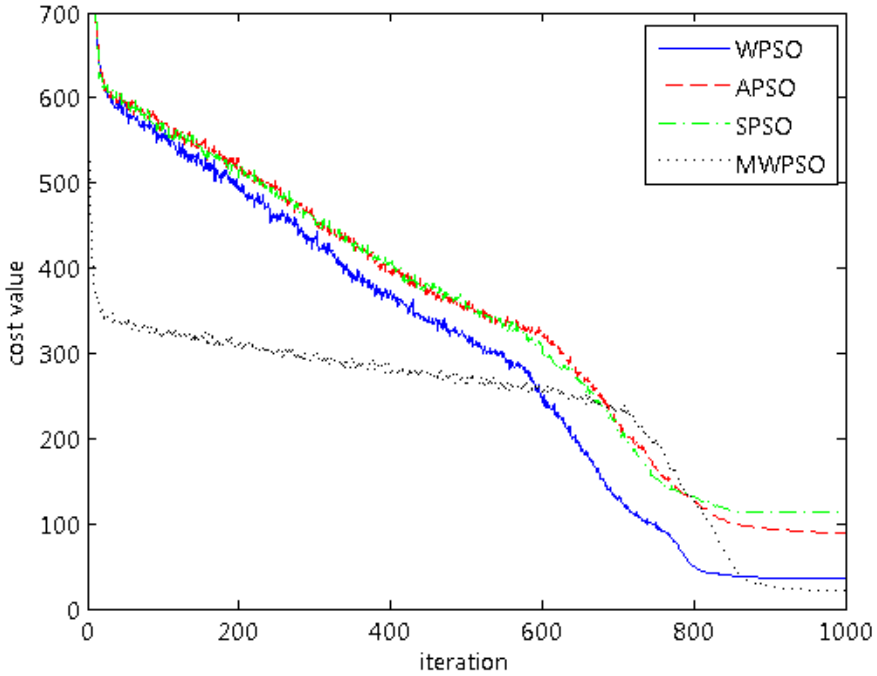
Function f_{16} is the generalized Griewank's function. Similar to function f_{14}, MWPSO is the fastest algorithm to converge to the optimal point. From the result obtained, MWPSO, WPSO and APSO all can reach the best fitness value. Yet, the value of standard deviation of MWPSO is the smallest among them. Again, MWPSO provides a more stable and high-quality result.

Function f_{17} is the Ackley's function. From Fig. 3.8b, all the algorithms provide a similar rate of convergence; however, the result shows that the proposed MWPSO provides the best result in terms of mean fitness value, standard deviation and best fitness value.

Function f_{18} is the Schwefel's function. From Fig. 3.8c, it is clearly shown that the searching ability of the proposed MWPSO is quite different from that of the other methods. All algorithms, except MWPSO, exhibit similar behavior at the first 400 times of iteration and then trapped in different local minima. The fitness value offered by MWPSO decreases gradually, and converges to the best result as compared with the other methods.

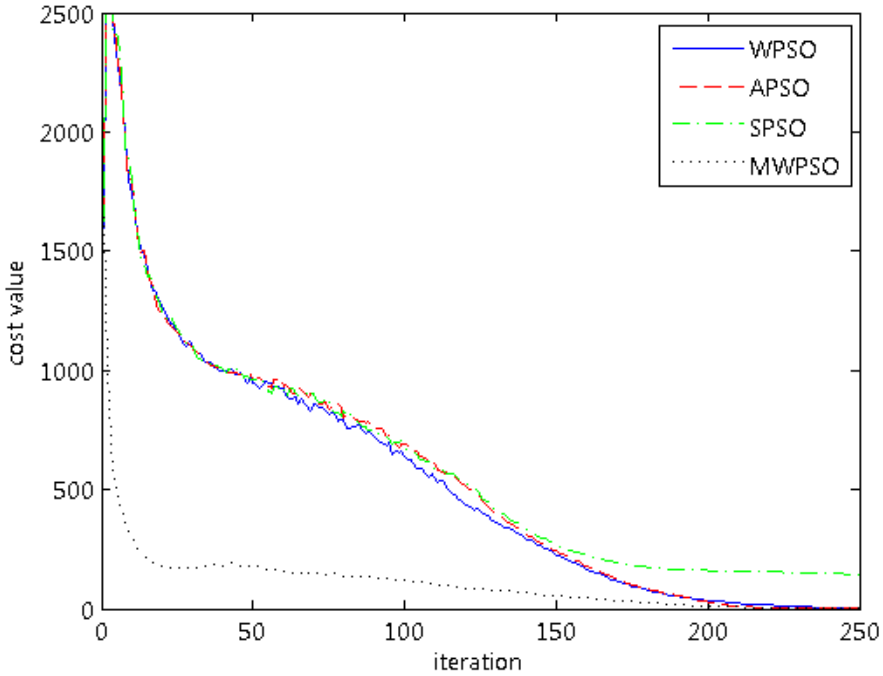


f_{14}

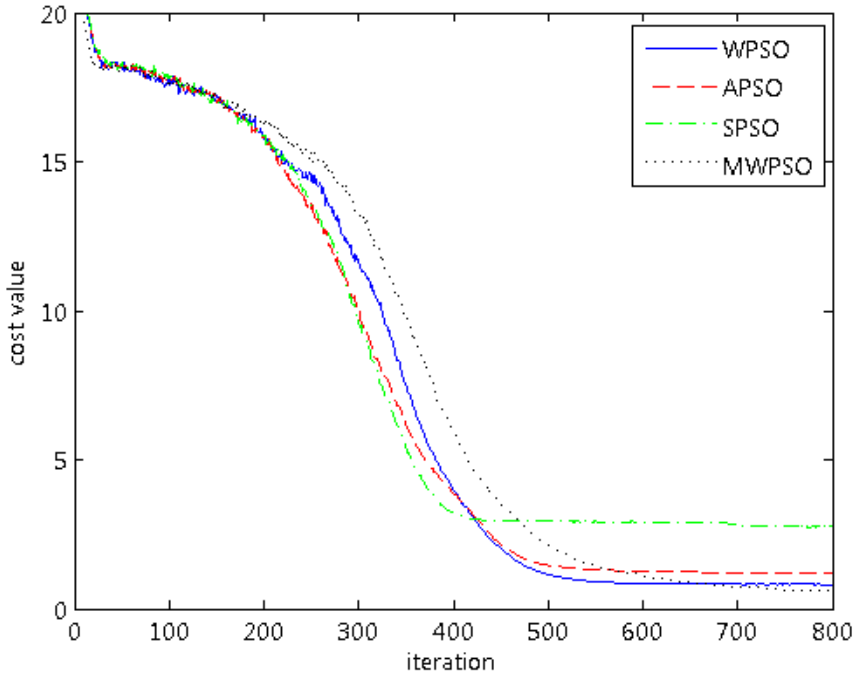


f_{15}

Fig. 3.8a. Comparisons between different hybrid PSOs for f_{14} and f_{15}.

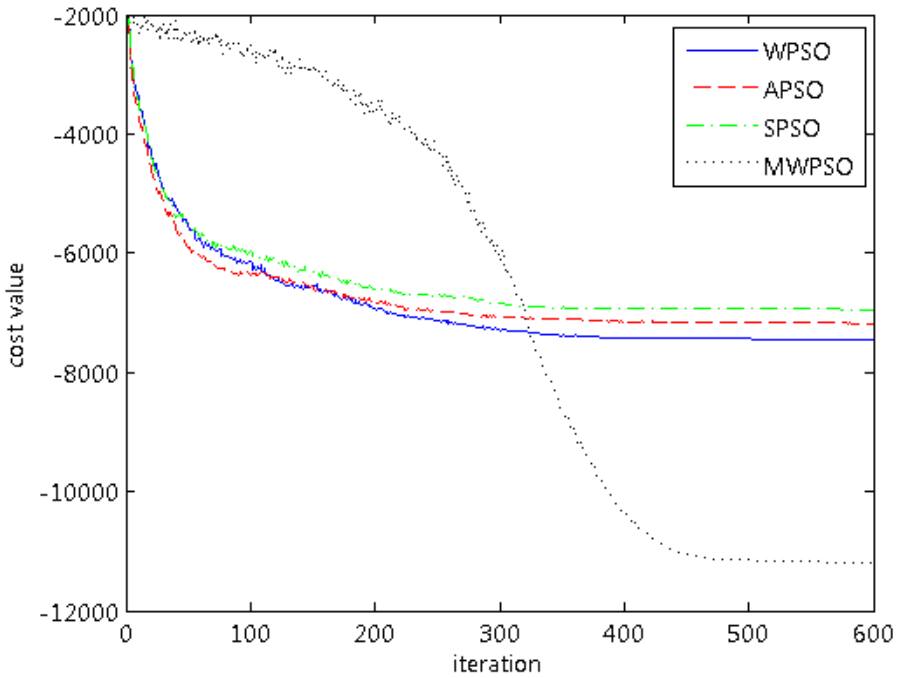


f_{16}



f_{17}

Fig. 3.8b. Comparisons between different hybrid PSOs for f_{16} and f_{17}.



f_{18}

Fig. 3.8c. Comparisons between different hybrid PSOs for f_{18}.

f_{14}, number of iteration: 500				
	MWPSO	WPSO	APSO	SPSO
Mean	**0.0044**	509.6625	0.0179	3278.0132
Best	**0**	0	0	3.8481
Std Dev	**0.0069**	1744.8548	0.112	3581.2285
f_{15}, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**21.8358**	36.5153	88.9324	112.6748
Best	10.9485	13.9301	**9.9711**	33.8994
Std Dev	**5.5178**	17.0411	46.6626	57.1291
f_{16}, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0**	0.1925	**0**	138.1759
Best	**0**	0	**0**	0.0709
Std Dev	**0**	0.2864	**0**	128.0549
f_{17} ($\times 10^{-3}$), number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**0.4561**	0.8049	1.2009	2.7416
Best	**0**	0	0.0006	0.0437
Std Dev	**2.8286**	3.9831	4.7975	4.9783
f_{18}, number of iteration: 1000				
	MWPSO	WPSO	APSO	SPSO
Mean	**-11210.9656**	-7441.1954	-7180.3602	-6951.7609
Best	**-12352.3469**	-8161.4782	-8159.9612	-8278.3995
Std Dev	578.664	**438.2333**	450.2045	656.8472

Table 3.3. Comparisons between different hybrid PSOs for f_{14} to f_{18}.

3.3.4.2. Time consumption test

In this test, the time required for different methods to reach the optimal point is recorded. The simulation for each function will be stopped when the fitness value has reached a threshold value.

Since the proposed method has one more parameter N_m introduced as compared to WPSO, it might increase the simulation time for each iteration step on finding the optimal solution. However, from the results shown in Table 3.4, we can see that MWPSO provides the best result among all the benchmark test functions. The time for each iteration step might increase for MWPSO; however, by introducing the multi-wavelet mutation into the PSO, the number of iteration to reach the optimal point can be reduced, meaning that the overall time consumption is reduced.

	Solution	Threshold	MWPSO	WPSO	APSO	PSO
F1	0	0.1	**1.6172**	2.3948	2.6217	3.4139
F2	0	0.1	**7.3587**	7.7805	7.5972	7.4092
F3	0	0.1	**1.228**	2.0284	2.0612	2.437
F4	0	0.1	**2.1407**	3.7846	3.7893	4.0521
F5	0		Trapped in different local minima			
F6	0	0.1	**3.3355**	5.049	5.27	4.9172
F7	-1	-0.95	**1.3967**	1.4186	1.6846	1.8979
F8	≈1	1.5	**0.0472**	0.0517	0.049	0.0475
F9	0.0003		Trapped in different local minima			
F10	-1	-0.9	**0.0264**	0.0289	0.0277	0.0284
F11	-1.0316	-0.95	**0.0375**	0.0564	0.0375	0.0545
F12	-3.8628	-3.8	**0.0535**	0.0546	0.0554	0.0584
F13	-3.32	-3	**0.062**	0.0657	0.0627	0.0623
F14	0	1	**1.969**	2.3724	2.578	3.2147
F15	0		Trapped in different local minima			
F16	0	1	**0.1414**	1.1492	1.1415	Trapped
F17	0		Trapped in different local minima			
F18	-12569.5		Trapped in different local minima			

Table 3.4. Simulation time in second (average over 50 times of running)

3.4 SUMMARY

In this chapter, the wavelet theory is applied to a proposed hybrid PSO with multi-wavelet mutation. Our objective is to increase the searching area by increasing the number of elements in a particle that undergo mutation so as to enhance the exploration and exploitation of the original WPSO. Simulation results have shown that the proposed algorithm is useful as a method to solve various optimization problems. On testing with a suite of benchmark functions, MWPSO in general shows better results in terms of solution quality and stability than the algorithms of WPSO, APSO and SPSO. Also, a faster convergence rate can be achieved by MWPSO.

CHAPTER 4

RESTORATION OF HALF-TONED COLOR-QUANTIZED IMAGES USING MWPSO

4.1 INTRODUCTION

Color quantization is a process of reducing the number of colors in a digital image by replacing them with some representative colors selected from a palette [M.T. Orchard, 1991]. It is widely used because it can save transmission bandwidth and data storage requirement in many multimedia applications. When color quantization is performed, certain degradation of quality will be introduced owing to the limited number of colors used to produce the output image. The most common artifact is the false contour. False contours occur when the available palette colors are not enough to represent a gradually changing region. Another common artifact is the color shift. In general, the smaller color palette size is used, the more severe the defects will be.

Half-toning is a reprographic technique that converts a continuous-tone image to a lower resolution, and it is mainly for printing [P. Heckbert, 1982] [R. Ulichney, 1987] [R.S. Gentile, 1990] [S.S. Dixit, 1991] [X. Wu, 1992]. Error diffusion is one of the most popular half-toning methods. It makes use of the fact that the human visual system is less sensitive to higher frequencies, and during diffusion, the quantization

error of a pixel is diffused to the neighboring pixels so as to hide the defects and to achieve a more faithful reproduction of colors.

Recovering a color-quantized image back to the original image is often necessary. However, most of recent restoration algorithms mainly concern the restoration of noisy and blurred color images [M. Barni, 2000] [G. Angelopoulosnd, 1994] [N.P. Galatsanos, 1991a] [N.P. Galatsanos, 1991b] [B.R. Hunt, 1984] [H. Altunbasak, 2001] [K.J. Boo, 1997] [N.P. Galatsanos, 1989]; literature for restoring half-toned color quantized images is seldom found. Since error diffusion is a nonlinear process, conventional gradient-oriented optimization algorithms might not be suitable to solve the addressed problem.

In this chapter, we make use of the Particle Swarm Optimization with multi-wavelet mutation (MWPSO), which is an adaptive search algorithm that works very well on optimization problems, to restore half-toned color-quantized images. By taking advantage of the wavelet theory, which increases the searching space for the PSO, the chance of converging to local minima is lowered. We will formulate the process of color quantization with error diffusion, and apply the proposed MWPSO to restore the degraded images. The results demonstrate that the WMPSO can achieve a remarkable improvement in terms of convergence rate and signal-to-noise ratio.

4.2 COLOR QUANTIZATION WITH HALF-TONING

A color image generally consists of three color planes, namely, O_r, O_g and O_b, which represents the red, green and blue color planes of the image respectively. Accordingly, the (i, j)-th color pixel of a 24-bit full-color image of size $N \times N$ pixels consists of three color components. The intensity values of these three components form a 3D vector $\vec{O}_{(i,j)} = (O_{(i,j)r}, O_{(i,j)g}, O_{(i,j)b})$, where $O_{(i,j)c} \in [0 \quad 1]$, is the intensity value of the c color component of the (i, j)-th pixel. Here, we assume that the maximum and the minimum intensity values of a pixel are 1 and 0 respectively.

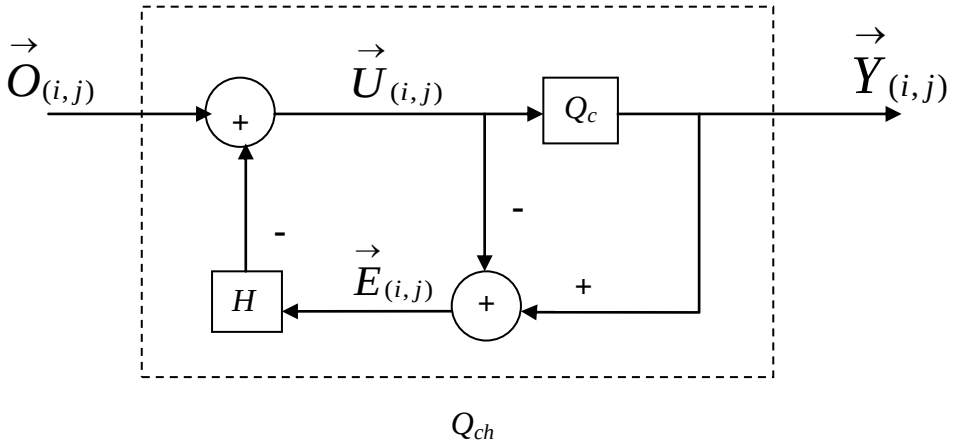


Fig. 4.1. Color quantization with half-toning.

Fig. 4.1 shows the system that performs color quantization with error diffusion. The input image is scanned in a row-by-row fashion from top to bottom and from left

to right. The relationship between the original image $\vec{O}_{(i,j)}$ and the encoded image $\vec{Y}_{(i,j)}$ is described by

$$U_{(i,j)c} = O_{(i,j)c} - \sum_{(k,l)\in\Omega} H_{(i,j)c}\, E_{(i-k,j-l)c} \tag{4.1}$$

$$\vec{E}_{(i,j)} = \vec{Y}_{(i,j)} - \vec{U}_{(i,j)} \tag{4.2}$$

$$\vec{Y}_{(i,j)} = Q_c[\vec{U}_{(i,j)}] \tag{4.3}$$

where $\vec{U}_{(i,j)} = (U_{(i,j)r}, U_{(i,j)g}, U_{(i,j)b})$ is a state vector of the system, $\vec{E}_{(i,j)}$ is the quantization error of the pixel at position (i, j) and $H_{(i,j)c}$ is a coefficient of the error diffusion filter for the c color component. Ω is the corresponding causal support region of $H_{(i,j)c}$.

The operator $Q_c[\cdot]$ performs a 3D vector quantization. Specifically, the 3D vector $\vec{U}_{(i,j)}$ is compared with a set of representative color vectors stored in a previously generated color palette $C = \{\hat{v}_i : i = 1,2,...N_c\}$. The best-matched vector in the palette is selected based on the minimum Euclidean distance criterion. In other words, a state vector $\vec{U}_{(i,j)}$ is represented by the color $\hat{v}_k$ if and only if $\| \vec{U}_{(i,j)} - \hat{v}_k \| \leq \| \vec{U}_{(i,j)} - \hat{v}_l \|$ for all $l = 1, 2, \ldots, N_c$; $l \neq k$. Once the best-matched vector is selected from the color palette, its index is recorded and the quantization error $\vec{E}_{(i,j)} = \hat{v}_k - \vec{U}_{(i,j)}$ is diffused to pixel (i, j)'s neighborhood as described in (4.1). Note that in order to handle the boundary pixels, $\vec{E}_{(i,j)}$ is defined to be zero when (i, j) falls

outside the image. Without loss of generality, in this thesis, we use a typical Floyd–Steinberg error diffusion kernel as $H_{(i,j)c}$ to perform half-toning. The recorded indices of the color palette will be used in the future to reconstruct the color quantized image with the same color palette.

4.3 FORMULATION OF RESTORATION ALGORITHM

4.3.1 Description for the fitness function

WMPSO is proposed to be used to restore the half-toned color-quantized image. Let X be the output image of the restoration. Obviously, when the restored image X is color-quantized with error diffusion, the output should be close to the original half-toned image Y. Suppose $Q_{ch}[\cdot]$ denotes the operator that performs the color quantization with half-toning, then we should have

$$Y \approx Q_{ch}[X] \tag{4.4}$$

Based on above criterion, the cost function for the restoration problem can be defined as

$$fitness = \sum [Y - Q_{ch}[X]] \tag{4.5}$$

The fitness is equal to the total absolute error between the original half-toned color-quantized image and the half-toned color-quantized restored image. If *fitness* = 0, it implies the restored image X and the original image O provide the same color-quantization results. In other words, the cost function provides a good measure to judge if X is a good estimate of O.

4.3.2 Restoration with MWPSO

Let $X_{cur}(t-1)$ be the group of the current estimates of the restored image (the swarm) at a particular iteration $t-1$, $X^{p}_{cur}(t)$ be the p-th particle of the swarm, and E^{P}_{cur} be its corresponding fitness value. According to (3.2) and (3.3), the new estimate of the restored image is given by

$$X^{p}_{new}(t) = X^{p}_{cur}(t-1) + k \cdot v^{p}(t) \tag{4.6}$$

where k is a constriction factor. Before evaluating the fitness of each particle, the wavelet mutation is performed. The mutation operation realizes a fine tuning. E^{p}_{new}, which is the fitness for $X^{p}_{new}(t)$, is then evaluated. If $E^{p}_{new} < E^{p}_{cur}$, $X^{p}_{cur}(t)$ is updated to $X^{p}_{new}(t)$. Furthermore, if $E^{p}_{new} < E_{best}$ happens, where E_{best} is the fitness value of the best estimate X_{best} so far, then X_{best} will be updated by $X^{p}_{new}(t)$. In summary, we have,

$$X^{p}_{cur}(t) = \begin{cases} X^{p}_{new}(t) & \text{if} \quad E^{p}{}_{new} < E^{p}{}_{cur} \\ X^{p}_{cur}(t-1) & \text{otherwise} \end{cases} \tag{4.7}$$

and

$$X_{best} = \begin{cases} X^{p}_{new}(t) & \text{if} \quad E^{p}{}_{new} < E_{best} \\ X_{best} & \text{otherwise} \end{cases} \tag{4.8}$$

4.3.3 Experimental Setup

On realizing the MWPSO restoration, the following simulation conditions are used:

- Shape parameter of the wavelet mutation (ζ_{wm}): 0.2

- Probability of mutation (μ_m): 0.2
- Element probability (N_m): 0.3
- Acceleration constant φ_1 : 2.05
- Acceleration constant φ_2 : 2.05
- Constriction factor *k*: 0.005
- Parameter *g*: 10000
- Swarm size: 8
- Number of iteration: 500
- Initial population *X*(0): All the particles are initialized to be the Gaussian filtered output of the original half-toned image *Y*.
- Initial global best *gbest*: The original half-toned image *Y*.
- Initial previous best $pbest^p$: a zero 3-D matrix

4.4 RESULT AND ANALYSIS

Simulations have been carried out to evaluate the performance of the proposed algorithm. Three *de facto* standard 24-bit full-color images of size 256×256 pixels each are used, and all the test images are shown in Fig. 4.1. These test images are color-quantized to produce the corresponding images *Y*. The color-quantized images with 256-color palette size, 128-color palette size, 64-color palette size and 32-color palette size are shown in Fig. 4.2 - Fig. 4.5 respectively.

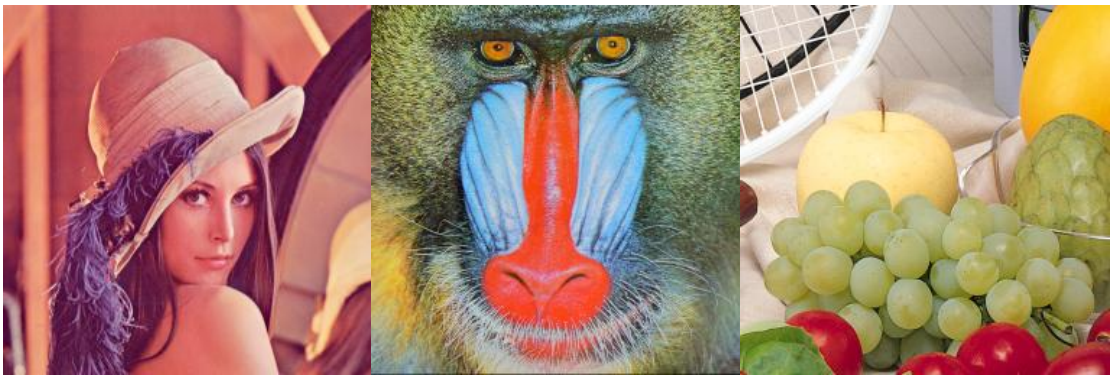
Color palettes of different sizes are generated using the median-cut algorithm [P. Heckbert, 1982]. On doing color quantization, half-toning is performed with error diffusion, and the Floyd–Steinberg diffusion filter [R. Ulichney, 1987] is used. The proposed restoration algorithm is used to restore the half-toned color-quantized images.

For comparison, Simulated Annealing (SA) is also applied to do the restoration. The performance of SA and the proposed MWPSO are reflected in terms of the signal-to-noise ratio improvement (SNRI) that is achieved when the algorithms are used to restore the color-quantized images “Lena”, “Baboon” and “Fruit”. The simulation condition for SA is basically the same as that of the proposed MWPSO. The control parameter of SA, *gamma*, is set to be equal to the parameter *k* of the MWPSO. The number of iteration is set at 500. The experimental results in terms of SNRI and the convergence rate are summarized in Table 4.1 - Table 4.3 and Fig. 4.14 - Fig. 4.19 respectively. The signal-to-noise ratio improvement (SNRI) is defined as:

$$\text{SNRI} = 10 \log \frac{\sum_{(i,j)} \left\| \vec{O}_{(i,j)} - \vec{Y}_{(i,j)} \right\|^2}{\sum_{(i,j)} \left\| \vec{O}_{(i,j)} - \vec{X}_{best(i,j)} \right\|^2}$$

where $\vec{O}_{(i,j)}$, $\vec{Y}_{(i,j)}$ and $\vec{X}_{best(i,j)}$ are the (*i*, *j*)-th pixels of the original, the half-toned color-quantized and the optimally restored images respectively.

From Figs. 4.6 - Fig. 4.13, we can see that the final results achieved by SA and MWPSO are similar: both of them converge to some similar values. It means that both algorithms can reach the same optimal point. However, we can see that the proposed MWPSO exhibits a higher convergence rate. Also, from Table 4.2 - Table 4.3, MWPSO converges to smaller fitness values in all experiments and offers higher SNRI in most of the experiments.



Lena Baboon Fruit

Fig. 4.1. Original images used for testing the proposed restoration algorithm.

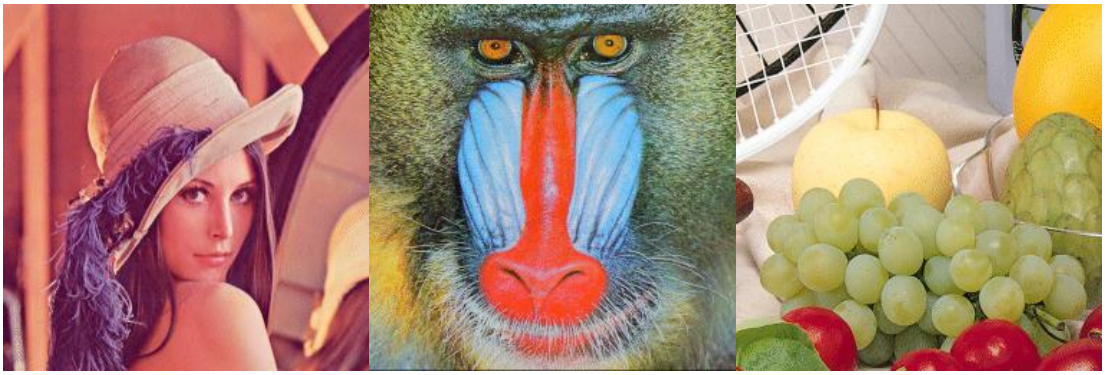


Fig 4.2. Half-toned images with 256 palette size.

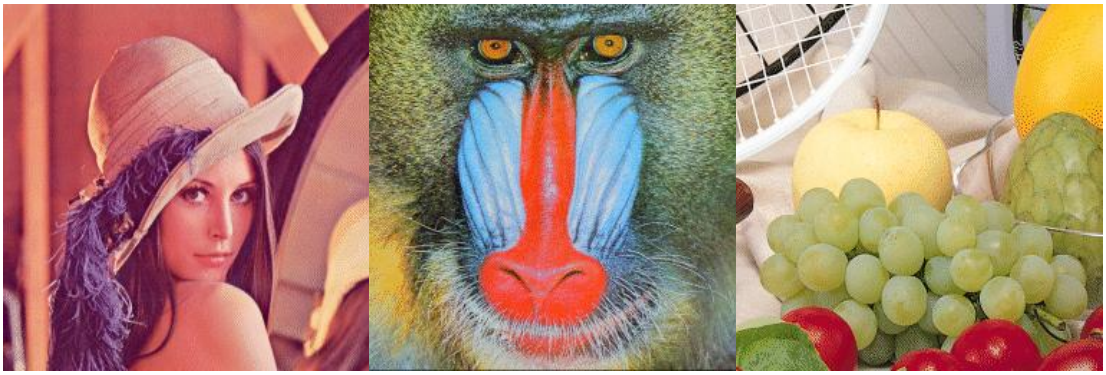


Fig. 4.3. Half-toned images with 128 palette size.

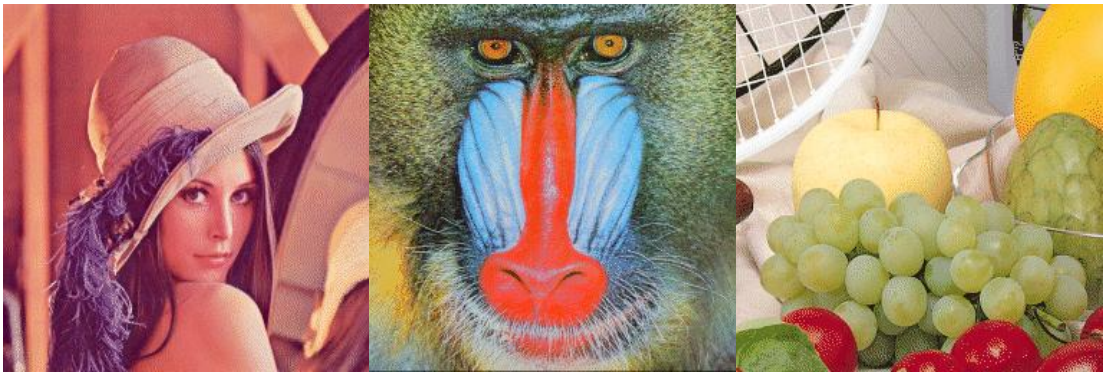


Fig. 4.4. Half-toned images with 64 palette size.

Fig. 4.5. Half-toned images with 32 palette size.

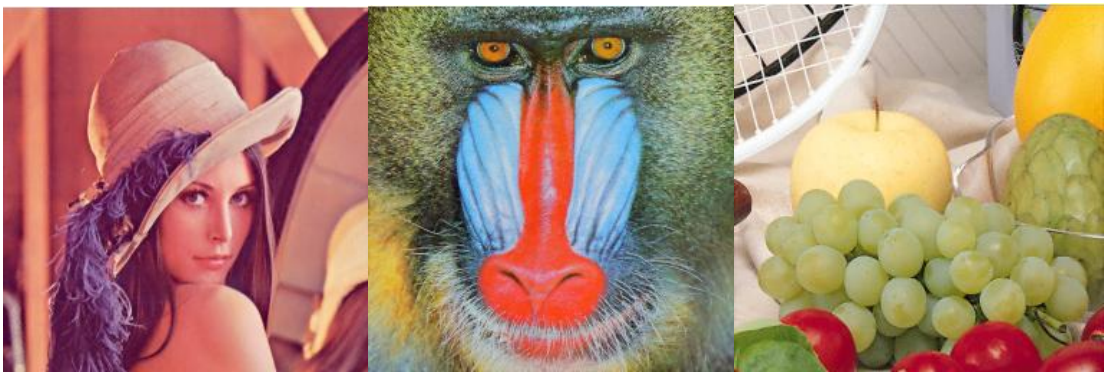


Fig. 4.6. MWPSO restored image with 256 palette size.

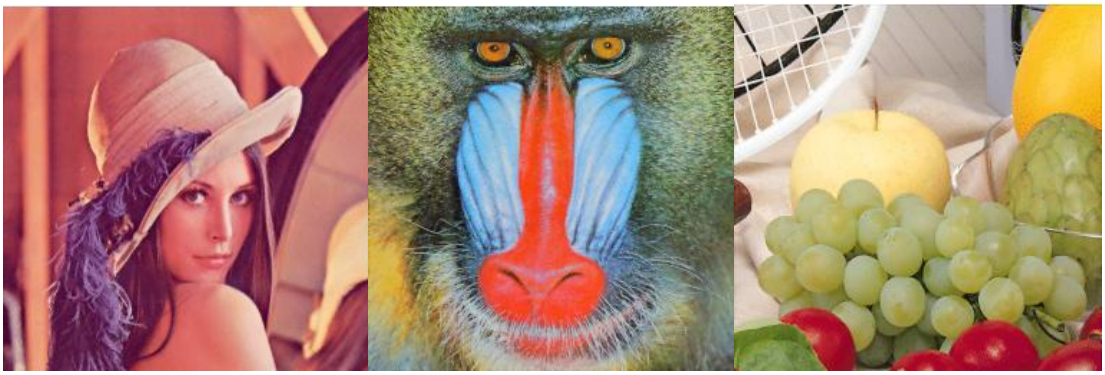


Fig. 4.7. MWPSO restored image with 128 palette size.

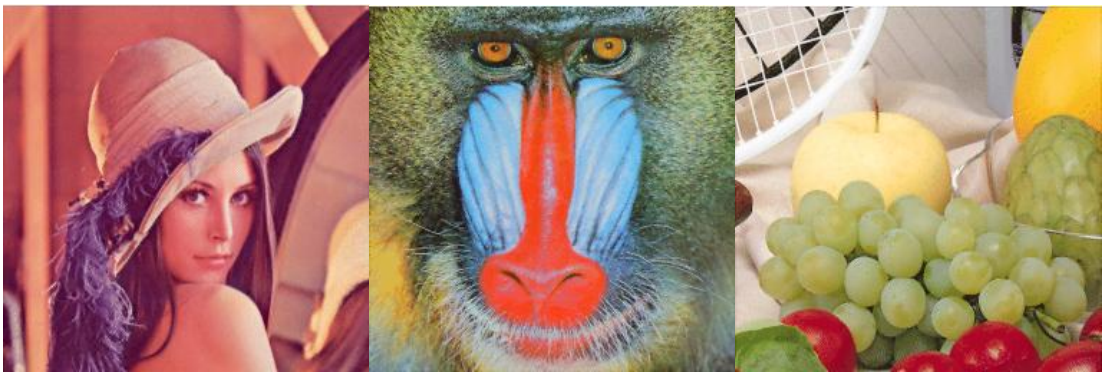


Fig. 4.8. MWPSO restored image with 64 palette size.

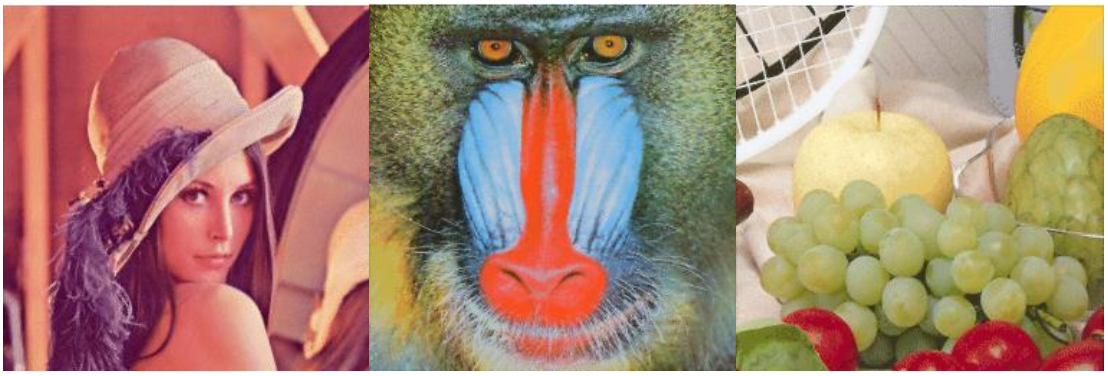


Fig. 4.9. MWPSO restored image with 32 palette size.

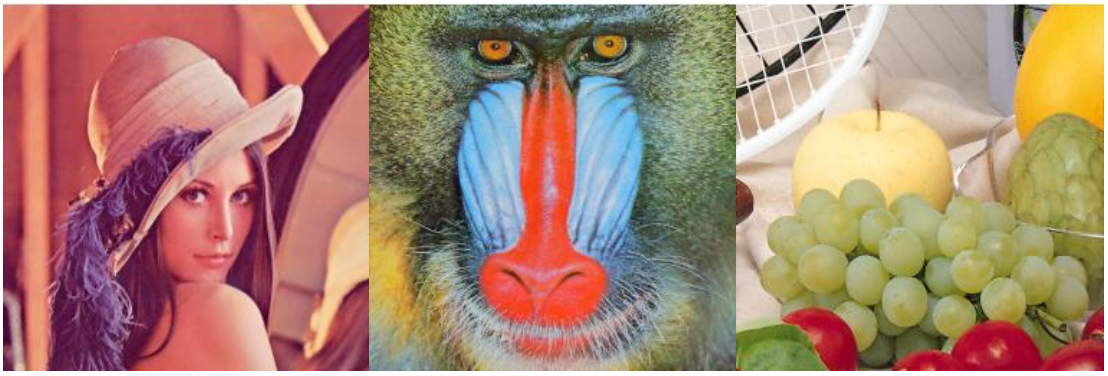


Fig. 4.10. SA restored image with 256 palette size.

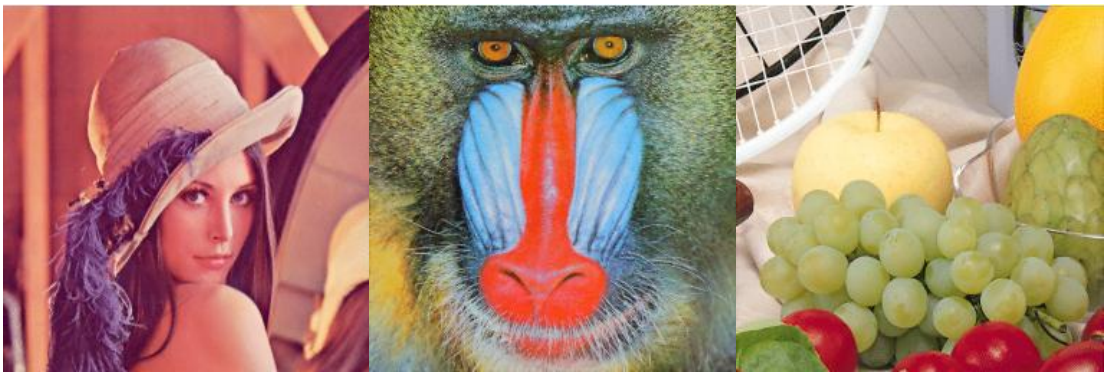


Fig. 4.11. SA restored image with 128 palette size.

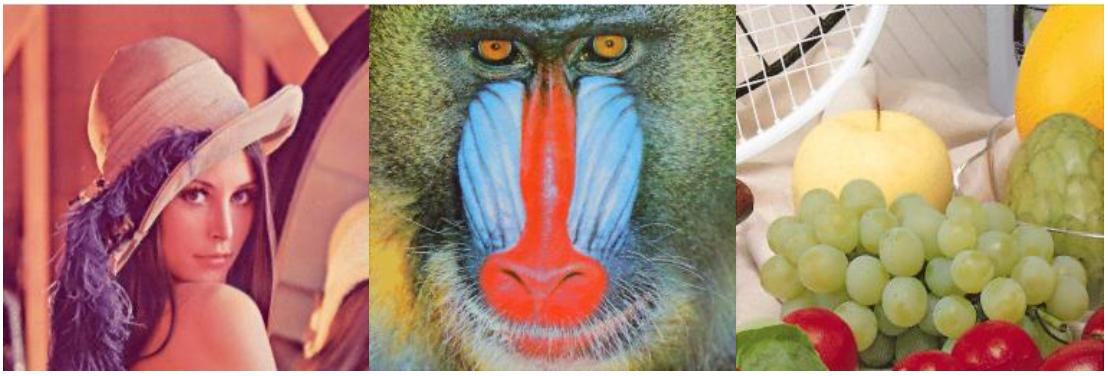


Fig. 4.12. SA restored image with 64 palette size.

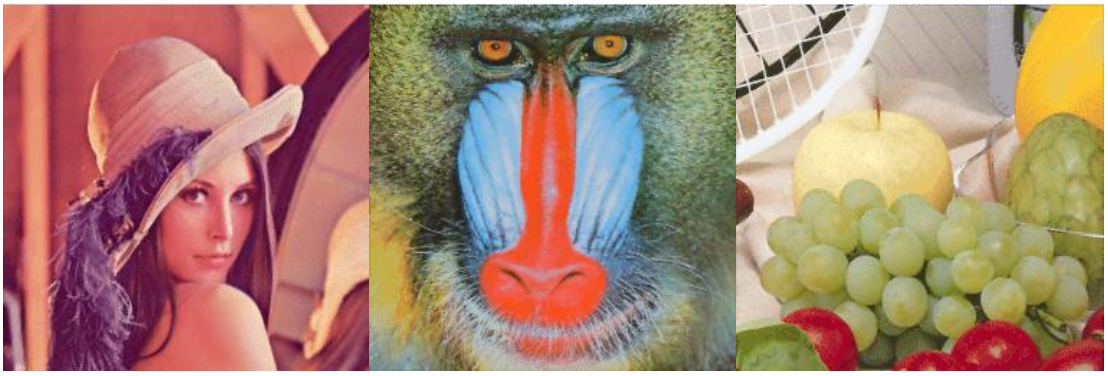


Fig. 4.13. SA restored image with 32 palette size.

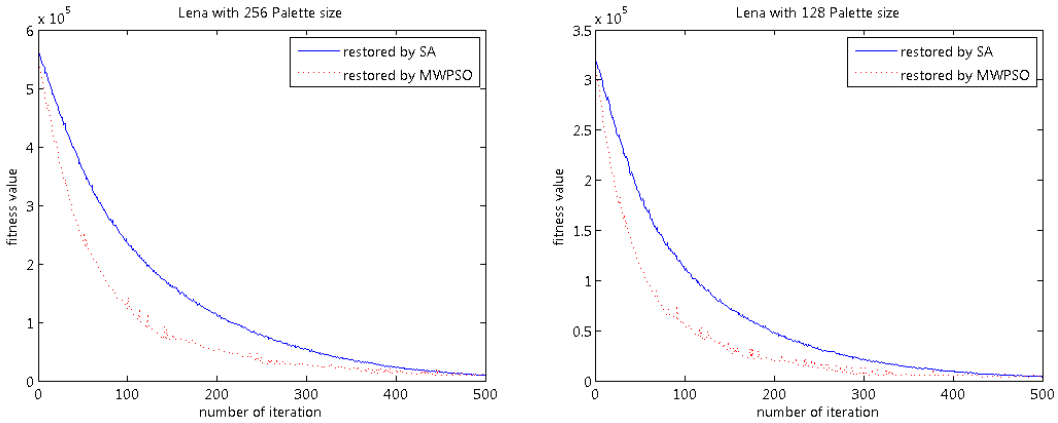


Fig. 4.14. Comparisons between SA and MWPSO for Lena with 256 and 128 palette size.

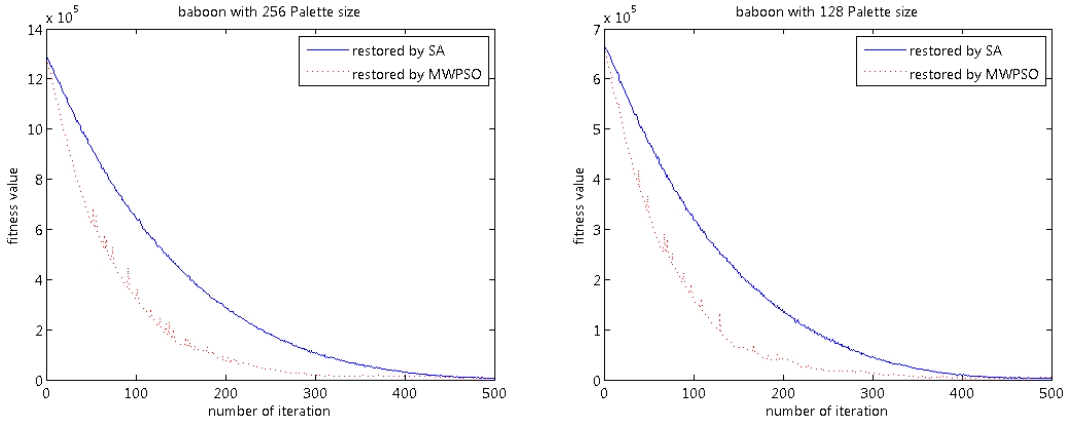


Fig. 4.15. Comparisons between SA and MWPSO for Baboon with 256 and 128 palette size.

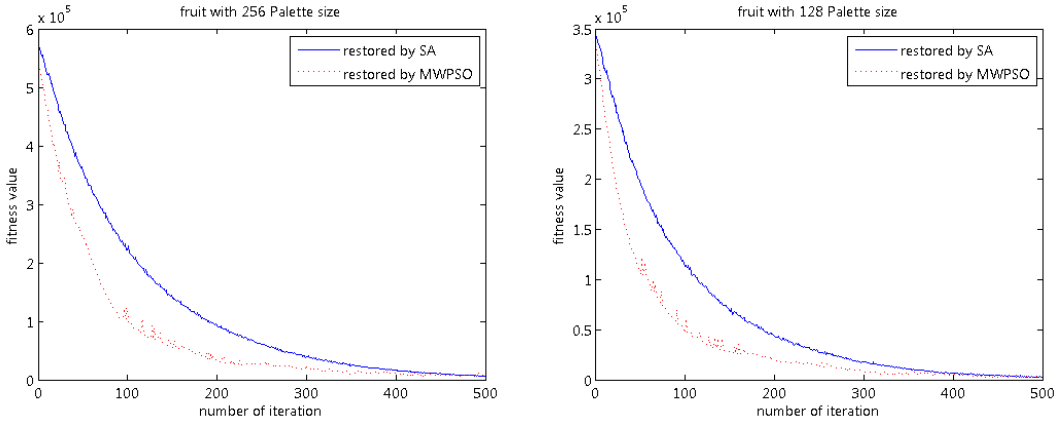


Fig. 4.16. Comparisons between SA and MWPSO for Fruit with 256 and 128 palette size.

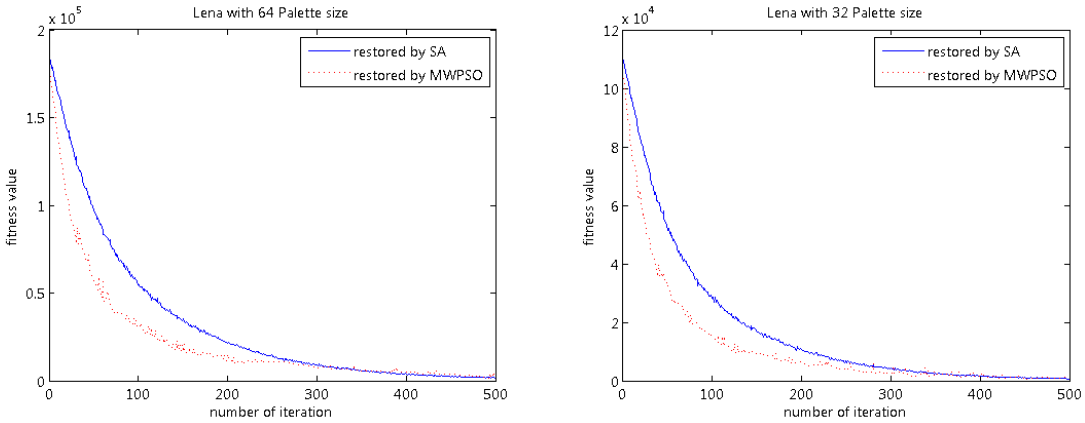


Fig. 4.17. Comparisons between SA and MWPSO for Lena with 64 and 32 palette size.

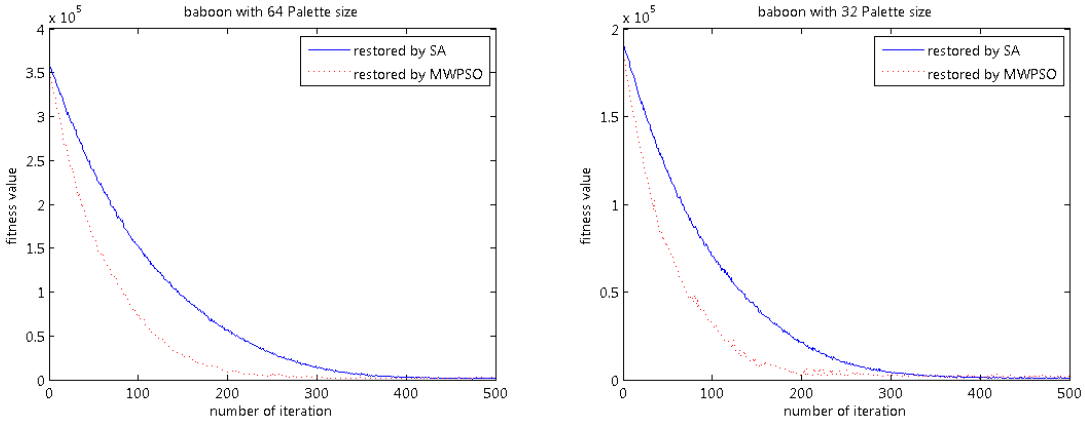


Fig. 4.18. Comparisons between SA and MWPSO for Baboon with 64 and 32 palette size

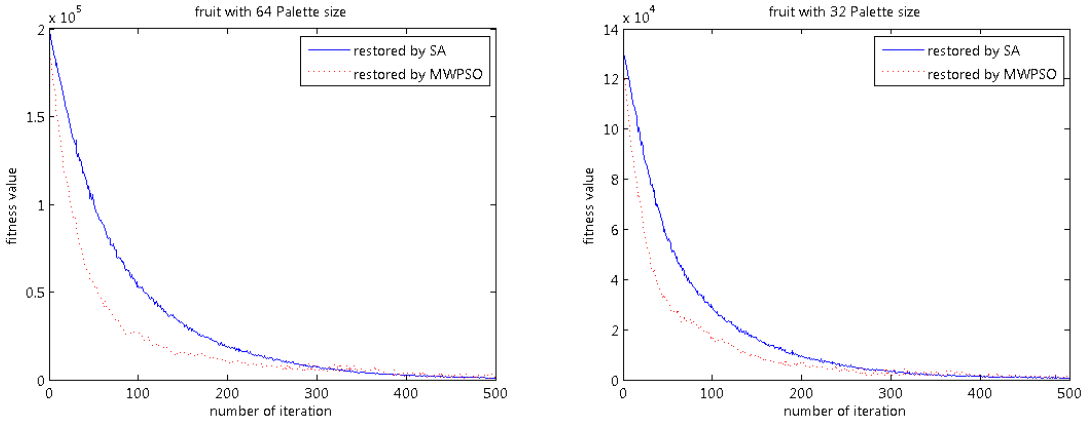


Fig. 4.19. Comparisons between SA and MWPSO for fruit with 64 and 32 palette size.

Lena with 256 Palette size		
	MWPSO	SA
SNRI	**6.7736**	6.7123
Best fitness	**8945**	8957
Lena with 128 Palette size		
	MWPSO	SA
SNRI	**8.1969**	8.1944
Best fitness	**3849**	3851
Lena with 64 Palette size		
	MWPSO	SA
SNRI	9.4404	**9.566**
Best fitness	**1604**	1652
Lena with 32 Palette size		
	MWPSO	SA
SNRI	**11.522**	11.4903
Best fitness	**649**	677

Table. 4.1. Fitness and SNRI of two algorithms for restoring the half-toned image Lena.

Baboon with 256 Palette size		
	MWPSO	SA
SNRI	**2.6854**	2.6706
Best fitness	**5410**	5445
Baboon with 128 Palette size		
	MWPSO	SA
SNRI	**4.3301**	4.3293
Best fitness	**2189**	2277
Baboon with 64 Palette size		
	MWPSO	SA
SNRI	5.5836	**5.616**
Best fitness	**1045**	1166
Baboon with 32 Palette size		
	MWPSO	SA
SNRI	**5.8208**	5.8186
Best fitness	**750**	760

Table. 4.2. Fitness and SNRI of two algorithms for restoring the half-toned image Baboon.

Fruit with 256 Palette size		
	MWPSO	SA
SNRI	**7.5598**	7.5155
Best fitness	**6372**	6394
Fruit with 128 Palette size		
	MWPSO	SA
SNRI	**9.9556**	9.9186
Best fitness	**2400**	2417
Fruit with 64 Palette size		
	MWPSO	SA
SNRI	**10.8616**	10.8607
Best fitness	**982**	1012
Fruit with 32 Palette size		
	MWPSO	SA
SNRI	**11.7873**	11.7815
Best fitness	**478**	574

Table. 4.3 Fitness and SNRI of two algorithms for restoring the half-toned image Fruit.

4.5 Summary

In this chapter, we use the Particle Swarm Optimization with multi-wavelet mutation (MWPSO) as the restoration algorithm for half-toned color-quantized images. The half-toned color-quantization and the application of the proposed MWPSO have been described. Simulation results demonstrate that the proposed algorithm can achieve an improvement in terms of convergence rate over the conventional Simulated Annealing (SA) algorithm. By using MWPSO, the number of particles used for searching is increased, and more freedom has been given to search the optimal result thanks to the multi-wavelet mutation. This explains the improved searching ability and the convergence rate. The chance for the particles being trapped in some local optimal point is also reduced.

CHAPTER 5

Gene Signature Selection using MWPSO with Multi-Kernel Support Vector Machine

5.1 INTRODUCTION

Cancer is one of the leading causes of death in developed countries [A. Vander, 2001]. Studies reveal that it is difficult to treat if cancer patients are diagnosed at a late stage. Epithelial cells (carcinomas), connecting/muscle tissue (sarcomas), and bone marrow cells (leukemia and lymphomas) are the main parts for cancer development. Successive mutations in the normal cells might lead to DNA damages and impair the replication mechanism, ultimately causing significant gene features that contribute to a cancerous state.

A microarray chip can simultaneously examine thousands of genes, which provides an extremely powerful tool for genomic studies of cancer. A few key genes (typically involving oncogenes and tumor suppressor genes), when mutated, will cause deregulation of the transcription and translation of other genes through complicated signaling pathways to initiate oncogenesis, and ultimately lead to derangement of the cellular phenotype and the clinical manifestation of cancer [M.D. Radmacher, 2002]. Analyzing microarray data could help discover some significant cancer genes and their mutual interactions, which can be used to generate hypothesis

for the identification and validation of genetic biomarkers for diagnostic and therapeutic purposes [M. Daly, 2002] [M.B. Eisen, 1998] [L.M. Fu, 2003] [T.R. Golub, 1999].

Significance based methods such as T-test or confidence intervals [S. Dudoit, 2003], which aim at finding statistically significant genes in differentiating various patient groups, have been extensively used. However, the common characteristic of these methods is to evaluate each individual gene one by one, thus neglecting the intrinsic interactions among genes. Multivariate methods, instead, are highly desirable to identify a group of genes that can cooperatively regulate tumor patterns. One of the commonly used approaches is to iteratively combine different genes as predictors for tumor classification. Gene predictors that account for specific clinical phenotypes should classify tumor samples into different groups in a robust manner. In other words, the performance of classification of a predictor can implicate the significance of genes. Classification problems have been addressed by different machine learning tools like neural network [F.Z. Brill, 1998] [J.H. Hong, 2006] [J. Khan, 2001], *k*-nearest neighborhood [L. Li, 2001], decision tree, self-organizing maps [T.R. Golub, 1999], graph theory [E. Hartuv, 2000], and Support Vector Machine (SVM) [B. Boser, 1992] [M. Daly, 2002] [S. Peng, 2003] [V.N. Vapnik, 1998] [Y.D. Zhao, 2003], etc. However, the choice and implementation of these tools are rather an empirical and experimental work. In fact, the data-driven machine learning techniques suffer from the "curse of dimensionality", which is a typical problem when analyzing microarray data. On the other hand, the recent methods [S. Peng, 2003] often solve the problems of feature selection and classifier optimization separately in tandem. Gene features are selected first according to prior knowledge or

their statistical significance. Then the classifier is optimized based on the selected feature set. There is no guarantee that the selected gene features can achieve optimal classification.

In this chapter, an approach that integrates Support Vector Machine with Particle Swarm Optimization (PSO) [J. Kennedy, 1995] is used to build a more reliable classifier. This approach incorporates gene features and SVM parameters into one common solution code. The problem is solved by simultaneously optimizing the selection of gene features and the parameters of the SVM classifier. The optimal solution achieved by this procedure may reach the best gene features and the corresponding optimal classifier.

5.1.1 Microarray data used in this experiment

We shall focus on osteosarcoma, which is the most common malignant bone tumor in children and accounts for 60% of malignant bone tumors diagnosed in the first two decades of life [M.P. Link, 2002]. The current standard care procedure is to give neo-adjuvant chemotherapy after initial diagnosis is made, followed by definitive surgery and postoperative chemotherapy. After surgery, according to its response to pre-operative chemotherapy, the tumor will be categorized to be a good responder (>90% necrosis) or a poor responders (<90% necrosis), and based on which a proper postoperative chemotherapy can be determined. Unfortunately the overall survival rate of the poor responders did not significantly improve [M.P. Link, 2002] [P.A. Meyers,1992]. It is possible that resistant tumor cells have additional time to either metastasize to the lungs or evolve further during the period when ineffective

preoperative chemotherapy is given. Therefore it is necessary to identify at the time of initial diagnosis the patients who are likely to have a poor response to standard preoperative therapy and therefore a poor outcome eventually. The DNA microarray data were collected from the tumor samples of patients enrolled in a clinical trial at the Texas Children's Hospital. Expression profiling by DNA microarray was done on tumor tissues before and after chemotherapy treatments. The classifier was initially developed with the integrated approach to distinguish good responders and poor responders based on the sampled data after chemotherapy. The results were compared with the existing methods tested on the same dataset [T.K. Man, 2005] to verify its performance and robustness. The derived classifier was further validated through predicting the potential chemo-response of biopsy samples acquired before chemotherapy.

5.2 CLASSIFICATION AND FEATURE SELECTION WITH SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) is one of supervised learning methods, which is used for doing classification and regression. By viewing input data as two sets of vectors and transforming the data into an *n*-dimensional space, an SVM can be constructed such that a separating hyperplane in the space that maximizes the margin between the two data sets is formed. In this section, we are going to describe the feature selection using a multi-kernel Support Vector Machine approach.

5.2.1 Support Vector Machine

Support Vector Machine basically is a binary classifier. Let a gene microarray dataset *D* be denoted by $\{(X_i, y_i)\}_{i=1}^{l}$, where $X_i \in R^m$ is the gene expression level of the *i*-th patient, $y_i \in \{-1,1\}$ is the condition label for a binary classification problem, *l* is the number of patients, and *m* is the number of gene features. The dataset after performing the gene selection is defined as $\ell(D) = \{(\ell(X_i), y_i)\}_{i=1}^{l}$ with $\ell \in R^{m'}$, where the function $\ell(\cdot)$ selects *m*' (<*m*) gene features among all the *m* gene features from the gene expression data set *D*. To construct the SVM, all training samples are first mapped to a feature space by a non-linear function $\phi(\ell(X_i))$. A separating hyperplane in the feature space can be expressed as $w = \sum_{i=1}^{l} \alpha_i y_i \phi(\ell(X_i)), \ \alpha_i \geq 0$. The

optimal separating hyperplane is defined as a linear classifier which can separate the two classes of training samples with the largest marginal width, and the optimal parameters $\alpha = [\alpha_i]$ are obtained by maximizing the following function:

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \left[\alpha_i \alpha_j y_i y_j \phi(\ell(X_i)) \cdot \phi(\ell(X_j)) \right] \qquad (5.1)$$

with $0 \le \alpha_i \le C_r, \quad i = 1,...,l$ in which C_r is the regularization parameter controlling the tradeoff between model complexity and empirical risk, and C_r is equal to 15 in this thesis. In the hyperplane expression, only those items with $\alpha_i > 0$ can remain. The samples that lie along the margins of the decision boundary (by Kuhn-Tucker theorem) are called the support vectors, which satisfy the equation:

$$w = \sum \alpha_i y_i \phi(\ell(X_i)), \quad \alpha_i \ge 0 \qquad (5.2)$$

To avoid the computation of the inner product $\langle \phi(\ell(X_i)), \phi(\ell(X_j)) \rangle$ in the high dimensional space during the optimization of (5.1), the kernel function that can satisfy the Mercer's condition is introduced:

$$K(\ell(X_i), \ell(X_j)) = \langle \phi(\ell(X_i)), \phi(\ell(X_j)) \rangle \qquad (5.3)$$

For a new test sample *X*, a decision function of the SVM classifier with the kernel *K* can then be defined based on the selected gene subset:

$$f(X, D, \ell) = \text{sgn}\left(\sum_{\substack{SUPPORT \\ VECTOR}} \alpha_i y_i K(\ell(X_i), \ell(X)) \right) \quad (5.4)$$

To develop a robust SVM model based on the training set in a practical way, the leave-one-out cross-validation (LOOCV) can be applied. Under this approach, one sample is left out as a testing sample, and the remaining $l-1$ samples are employed as training samples. Let $\overline{D_k} = \{(X_i, y_i)\}_{i=1, i\neq k}^{l}$ represent the training set, the overall accuracy is reflected by:

$$J(D, \ell) = \frac{1}{2l}\left(\sum_{k=1}^{l} \left| f(X_k, \overline{D_k}, \ell) - y_k \right| \right) \quad (5.5)$$

where $J(D, \ell)$ is the total error of the classifier. Now the integrated gene feature selection and the SVM classification problem are transformed to optimizing the above single objective function (5.5) by searching the feature selection ℓ (and the multi-kernel SVM parameters as discussed below.)

5.2.2 Multi-Kernel approach for SVM

Typical kernel functions used in SVM design include the linear, sigmoid, and radial basis function (RBF). RBF is widely used as the kernel function in gait classification studies [V.N. Vapnik, 1998] [V. Vapnik, 2000]. As there is no analytical study for choosing the optimal kernel function, it is usually chosen by trial and error, which depends on the problem nature. As a result, the selected kernel to

the problem may not be a good one. We propose a multi-kernel approach for Support Vector Machines to combine heterogeneous sources of information for learning the decision function. The parameters in the integrated kernel could be found by some tuning algorithm. The proposed multi-kernel approach defines a new kernel that combines different kernels linearly:

$$K = \sum_{i=1}^{3} \omega_i K_i \qquad (5.6)$$

$$\sum_{i=1}^{3} \omega_i = 1 \qquad (5.7)$$

where K_1, K_2, K_3 are the linear, sigmoid and radial basis functions respectively, and ω_i is the weight for each kernel.

5.3 DATA DESCRIPTION

We use a set of osteosarcoma microarray data [T.K. Man, 2005], which were generated through institutional reviewed board-approved protocols at four centers (Texas Children's Hospital/Baylor College of Medicine, Cook Children's Medical Center, Pediatric Branch of the National Cancer Institute, and University of Oklahoma Health Science Center). A total of 34 samples (14 initial biopsies and 20 definitive surgery specimens) were included, which were obtained from 28 individual patients with 18 males and 10 females. Six patients contributed two samples each, both initial biopsies and definitive surgery specimens; whereas the remaining 22 patients contributed one sample each, either an initial biopsy or a definitive surgery specimen. The initial biopsy samples were obtained at the time of diagnosis before the initiation of preoperative chemotherapy. The definitive surgery samples were collected after the completion of preoperative chemotherapy. The drug responses are centrally reviewed by one pathologist after definitive surgery. Good response is defined as more than 90 percent necrosis in tumor, and poor response with less than 90 percent necrosis.

It is clinically useful to monitor drug response after chemotherapy so that further therapeutic measures could be adjusted to optimize the chance of survival of the poor responders as revealed at the time of definitive surgery. Another more significant approach is to predict the drug response before chemotherapy is initiated, which provides clinicians the option to customize therapy based on the predicted response. Therefore the gene arrays of the 20 definitive surgery specimens can be

used to train the classifier, and the gene arrays of the 14 initial biopsies can be used to verify the trained classifier.

To process the gene data, the raw quantification outputs of all array experiments were pre-processed and filtered by removing spots with low signal intensity and low sample variance ($P > 0.01$), as well as those that were missing in the experiments. A total of 1,934 genes remained after pre-processing and filtering. The intensities were then normalized by an intensity dependent local weighted regression method. After normalization, the intensity ratios were log-transformed before further analysis.

5.4 FORMULATION OF GENE SELECTION

The number of features in microarrays is significantly larger than the number of samples. In order to improve the performance of the classification, reducing the dimension of the data set is necessary. The filtering technique has been adopted for this purpose.

As shown in Fig. 5.1, the two-sample *t*-test is performed to pre-screen the noisy genes, and the remaining unfiltered genes are the more significant genes. The *t*-value of the gene feature can be determined by the following equation:

$$t = \frac{\mu_{-1} - \mu_1}{\sqrt{\frac{Var_1}{n_1} + \frac{Var_{-1}}{n_{-1}}}} \tag{5.8}$$

where n_1 and n_{-1} are the numbers of patients with positive and negative conditions, μ_1 and Var_1 are the mean value and variance of gene feature in the positive group, and μ_{-1} and Var_{-1} are the mean value and variance of gene feature in the negative group, respectively. The gene features with low *t*-values are filtered out, and the ones with high *t*-values are passed to the MWPSO optimization unit, which can search for the most significant gene features and the SVM parameters for classification. Practically, the gene features that have *t*-values higher than 2.5 are passed to the MWPSO optimization unit and the rest of the gene features are not considered. Fig 5.1 illustrates the procedures of the proposed algorithm, and the details are discussed below.

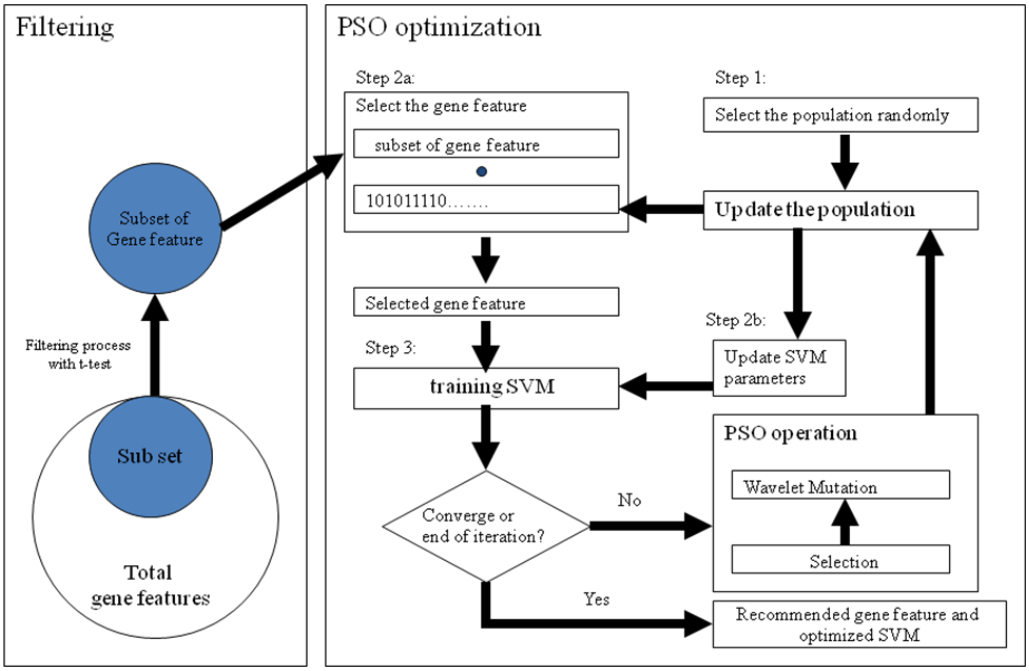


Fig. 5.1. Illustration for the proposed algorithm.

Step 1: Generate the population randomly

A set of solutions is generated randomly which represents the particles of the PSO. Each solution consists of two parts, the binary-coded representation and the real-coded representation. The binary-coded representation is used for gene feature selection and the real-coded representation is for determining the weighting of each kernel of the SVM.

Binary-coded representation is composed of a fixed-length binary string. It is used to determine the usage of the gene feature by their corresponding gene. If a bit value is 1, that gene is kept in the subset; and a bit value of 0 indicates that the gene is not included in the subset. Therefore, the string length is equal to the number of genes. The real-coded representation governs the weights of the three different kernels of the SVM.

Step 2: Fitness evaluation

As illustrated in Fig 5.1, the fitness value's evolution is divided into 2 steps. In step 2a, the binary coded representation part is separated from the solution string. Based on the selected gene features from the binary-coded representation, the SVM classifier is then trained in Step 2b with real-coded parameters through leave-one-out-cross-validation (LOOCV), during which the following cost function is used to evaluate the fitness of a solution

$$fitness = error = J(D, \ell, \omega) \qquad (5.9)$$

The objective of the MWPSO is to minimize the fitness function by searching the optimal gene feature ℓ and the weight vector ω of the SVM kernel. It can be seen that the best fitness value is zero if there is not any error.

Step 3: Termination condition

There are many possible termination conditions for the PSO. In this experiment, the termination condition is set as when the number of iteration is equal to a pre-defined number or the fitness is equal to zero, meaning that the best solution is found; otherwise, the algorithm goes to Step 4 for the next evolutionary iteration.

Step 4: PSO operation

In this step, all the particles will update their position according to the *pbest* and *gbest* values. After that some of the particles will be selected to undergo wavelet mutation. Every particle of the swarm will have a chance to mutate governed by the

probability of mutation μ_m. The number of elements in a particle that mutate is controlled by the element probability N_m, such that more than one element in each particle can mutate in each mutation process.

5.5 EXPERIMENTAL SETUP

On realizing the MWPSO, the following simulation conditions are used:

- Shape parameter of the wavelet mutation (ζ_{wm}): 0.2
- Probability of mutation (μ_m): 0.2
- Element probability (N_m): 0.3
- Acceleration constant φ_1 : 2.05
- Acceleration constant φ_2 : 2.05
- Constriction factor *k*: 0.005
- Parameter *g*: 1000
- Swarm size: 20
- Number of iteration: 1000
- Initial population: All the particles are initialized randomly
- Initial global best *gbest*: It is generated uniformly at random
- Initial local best *pbest*: It is generated uniformly at random

5.6 RESULT AND ANALYSIS

The SVM classifier was first trained by the sample data training set to classify the tumor to be a good responder or a poor responder. The best gene combination can be derived according to its cross-validation accuracy. Then the gene subset and the trained SVM classifier will be verified by using the test data set. The results show the relevance of using the selected gene combination as the signature of chemoresistance, and the ability of this gene signature to predict at the time of initial diagnosis the histologic response as determined at the time of definitive surgery.

The result of the proposed algorithm was used to compare with the results of various supervised classification algorithms (compound covariate predictor CCP, linear discriminant analysis LDA, 1-nearest neighbor 1-NN, 3-nearest neighbor 3-NN, nearest centroid NC and SVM) collected from [T.K. Man, 2005]. For all algorithms, LOOCV was used to fairly evaluate the performance of each classifier, and all of them can only achieve 60-70% classification accuracies. Besides, different hybrid PSOs incorporating different kernels for the SVM were also tested for comparisons of performance. In the training, the proposed method and all other methods using PSO can achieve 100% classification accuracy, as shown in Table 5.1. That means out of the 20 samples of training sets, all samples can be classified successfully. Comparing with other classifiers, the proposed classifier is more effective for the gene signature selection and classification. In general, the classifier with parameters optimized by PSO can provide a better result.

For the proposed method, the weights of kernels ω_1, ω_2 and ω_3 were found to be 0.533, 0.1663 and 0.3007 respectively. In this thesis, there are 937 genes selected

as the predictor genes; however, there is not enough information to explain the cooperative relationship among genes in the subsets.

The optimal SVM classifier with the selected gene subset was then respectively applied to predict the chemo-response of the 14 initial biopsy samples for further validation purpose. The test data are completely independent of the training data and process.

In the testing, the proposed method misclassified one sample among the 14 samples, with a correct classification rate of 92.9% as shown in Table 5.2. The results in [T.K. Man, 2005] also misclassified Tumor ID 410 of a clinically good responder as a poor responder. This finding is consistent with the study in [T.K. Man, 2005], where it was pointed out that this patient initially presented with localized disease but eventually developed recurrence in the lungs 25 months after the completion of therapy. It was suggested that there were resistant cells present in the initial biopsy, which might have metastasized to the lungs before the definitive surgery, and subsequently gave rise to the recurrent tumor. This result further indicates that the gene expression signature of the resistant cells in the definitive surgery samples was already present in the initial biopsy samples at the time of diagnosis. Comparing with the other methods, the proposed method can provide the best result among all the methods. In particular, among all the methods incorporating PSO for tuning the SVM, the proposed multi-kernel SVM can offer the most accuracy result as summarized in Table 5.2.

tumor ID of the training sets	Histologic response	concordance of classification with histologic response									
		CCP	LDA	1-NN	3-NN	NC	SPSO with multi-kernel SVM	APSO with multi-kernel SVM	MWPSO with sigmoid kernel SVM	MWPSO with RBF kernel SVM	Proposed method
300	Good	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
308	Good	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
386	Good	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
394	Good	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
452	Good	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
759	Good	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
771	Good	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
221	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
236	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
241	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
252	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
311	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
342	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
392	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
483	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
591	poor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
680	poor	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
691	poor	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
760	poor	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
761	poor	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
% correctly classified		65%	70%	70%	70%	65%	100%	100%	100%	100%	100%

Table 5.1. LOOCV of 20 definitive surgery osteosarcoma samples

Tumor ID	Histologic response	SPSO with multi-kernel SVM	APSO with multi-kernel SVM	MWPSO with sigmoid kernel SVM	MWPSO with RBF kernel SVM	Proposed algorithm with multi-kernel SVM
410	Good	No	No	No	No	No
197	Poor	Yes	Yes	Yes	Yes	Yes
207	Poor	Yes	Yes	Yes	Yes	Yes
278	Good	Yes	Yes	Yes	Yes	Yes
289	Poor	Yes	Yes	Yes	Yes	Yes
345	Good	Yes	Yes	Yes	Yes	Yes
204	Poor	Yes	Yes	Yes	Yes	Yes
274	Poor	Yes	Yes	Yes	Yes	Yes
299	Good	Yes	Yes	Yes	Yes	Yes
464	Poor	Yes	Yes	Yes	Yes	Yes
479	Poor	No	Yes	No	No	Yes
481	Poor	Yes	Yes	Yes	Yes	Yes
545	Good	Yes	Yes	No	No	Yes
654	Good	Yes	Yes	Yes	Yes	Yes

Table 5.2. Classification results of the proposed algorithm.

5.7 SUMMARY

In this chapter, we have proposed an integrated approach for applying MWPSO and Support Vector Machine to select a compact gene subset and optimize SVM parameters simultaneously. Prediction of chemo-response using an independent dataset of 14 initial biopsy samples achieves 92.9% of accuracy, reflecting that the proposed algorithm is promising in selecting gene signatures that are consistently expressed in initial biopsy samples. It indicates a high predicting value for drug response. The results suggest that the proposed algorithm can be used to generate hypothesis for the identification and validation of genetic biomarkers for diagnostic and therapeutic purposes. In the future, we will further validate the proposed algorithm in solving other similar classification problems with large data sets of nasopharyngeal carcinoma and lung cancer.

CHAPTER 6

CONCLUSION

6.1 ACHIEVEMENTS

In this thesis, we have proposed an improved Particle Swarm Optimization, and provide some example applications to illustrate the ability of the proposed method. For the Particle Swarm Optimization, a new mutation operation has been introduced. By taking advantage of the multi-wavelet mutation operation, the solution quality and the solution stability are enhanced. Application examples on half-toned color-quantized image restoration and micro-array data selection have been provided to show the merits of the proposed method.

In Chapter 3, we have presented the details about the Particle Swarm Optimization with multi-wavelet mutation. The standard PSO does not have any mutation operation, and it is easily trapped in some local optima. By using the multi-wavelet mutation operation, the solution quality is improved because more freedom is provided for all particles to explore the searching area. The probability of reproducing good particles during the iteration process is increased. The property of the wavelet theory also enhances the stability of the solution statistically. A suite of 18 benchmark test functions has been used to test and illustrated the searching ability

of the proposed method. The merit of the proposed PSO with multi-wavelet mutation over other methods has been clearly indicated.

The application of the proposed MWPSO on the restoration of half-toned color-quantized images has been discussed in Chapter 4. Error diffusion is one of the most popular half-toning methods; however, most of the recent restoration algorithms mainly concern on the restoration of noisy and blurred color images. We use the Particle Swarm Optimization with multi-wavelet mutation as the restoration algorithm for half-toned color-quantized images. Simulation results have demonstrated that the MWPSO can achieve a remarkable improvement in terms of convergence rate over the conventional simulated annealing (SA) algorithm. By using MWPSO, the number of particles used for searching is increased. The searching area is also increased in the early stage of iteration thanks to the multi-wavelet mutation, and each particle can share the information in each iteration step. This explains the improved searching ability and convergence rate.

In Chapter 5, we have discussed another application example of the proposed MWPSO on tuning a support vector machine with multiple kernels for gene signature selection and cancer prediction. As DNA microarray studies produce a large amount of data, expression data are highly redundant and noisy and most genes are believed to be uninformative with respect to the studied classes. Only a fraction of genes may present distinct profiles for different classes of samples. An integrated approach of support vector machine (SVM) and particle swarm optimization (PSO) has been adopted as a classification tool for selecting the useful information. The results of simulations have shown that the proposed algorithm can be used to generate

hypothesis for the identification and validation of genetic biomarkers for diagnostic and therapeutic purposes.

6.2 FURTHER DEVELOPMENT

In this thesis, the mutation operation for the Particle Swarm Optimization is investigated. The operation provides more freedom for the particles to explore the searching space in each iteration step. However, the selection operation of particles and the termination condition for the iteration process have not been seriously considered in this thesis.

The selection operation is an important step for optimization; the improvement of the selection operation should be considered in the further study. For the standard PSO, during each iteration step, all the particles will be used to reproduce a new set of particles. Individual solutions are selected through a fitness based process, and a new particle will replace an old one if the fitness of the new particle is higher than that of the old one. A fitness based selection process might not be a good one for the particle selection; some information might be loss due to the replacement process. Studies of the selection process might improve the performance of the PSO, particularly when the functions to be optimized are highly nonlinear multi-modal ones.

How to determine if the algorithm has reached the best solution is another important concern for the optimization. In many popular optimization algorithms including the Genetic Algorithm and Differential Evolution, a fixed number of iteration is usually pre-defined as the terminate condition. It might unnecessarily waste too much computational power if the solution has been reached very early. On the other hand, the pre-defined number of iteration might not be big enough to allow the algorithm to reach the global solution. The final choice of the number of iteration

is usually determined by trial and error or from experience. A study on the termination condition might further improve the performance of the PSO.

APPENDIX

A.1 BENCHMARK TEST FUNCTIONS

The 18 benchmark test functions for testing the MWPSO performance are listed below. In these functions, $\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_n]$, where *n* is the number of variables.

A.1.1 Unimodal Functions

Function 1: Sphere Model

$$f_1(\mathbf{x}) = \sum_{i=1}^{30} x_i^2, \qquad -50 \le x_i \le 150, \qquad \min(f_1) = f_1(\mathbf{0}) = 0$$

Function 2: Generalized Rosenbrock's Function

$$f_2(\mathbf{x}) = \sum_{i=1}^{29} \left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2 \right], \quad -2.048 \le x_i \le 2.048, \quad \min(f_2) = f_2(\mathbf{1}) = 0$$

Function 3: Step Function

$$f_3(\mathbf{x}) = \sum_{i=1}^{30} \left(\lfloor x_i + 0.5 \rfloor \right)^2, \quad -5 \le x_i \le 10, \quad \min(f_3) = f_3(\mathbf{0}) = 0,$$

Function 4: Quartic Function (with noise)

$$f_4(\mathbf{x}) = \sum_{i=1}^{30} i x_i^4 + random[0,1), \quad -1.28 \le x_i \le 2.56, \qquad \min(f_4) = f_4(\mathbf{0}) = 0$$

where $random(\cdot)$ generates randomly a floating-point number between 0 and 1

Function 5: Schwefel's Problem 2.21

$$f_5(\mathbf{x}) = \max_i \{ |x_i|, 1 \le i \le 30 \}, \qquad -150 \le x_i \le 50, \qquad \min(f_5) = f_5(\mathbf{0}) = 0$$

Function 6: Schwefel's Problem 2.22

$$f_6(\mathbf{x}) = \sum_{i=1}^{30} |x_i| + \prod_{i=1}^{30} |x_i|, \qquad -5 \le x_i \le 15, \quad \min(f_6) = f_6(\mathbf{0}) = 0$$

Function 7: Easom's Function

$$f_7(\mathbf{x}) = -\cos(x_1) \cdot \cos(x_2) \cdot \exp\left(-\left((x_1 - \pi)^2 + (x_2 - \pi)^2 \right) \right), \qquad -300 \le x_1, x_2 \le 300,$$

$$\min(f_7) = f_7([\pi, \pi]) = -1$$

A.1.2. Multimodal Functions with Only a Few Local Minima

Function 8: Shekel's Foxholes Function

$$f_8(\mathbf{x}) = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} \left(x_i - a_{ij}\right)^6} \right]^{-1}, \qquad -65.536 \le x_i \le 65.536,$$

$$\min(f_8) = f_8([-32, -32]) \approx 1,$$

where $[a_{ij}] = \begin{bmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \cdots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \cdots & 32 & 32 & 32 \end{bmatrix}$

Function 9: Kowalik's Function

$$f_9(\mathbf{x}) = \sum_{i=1}^{11} \left[a_i - \frac{x_1\left(b_i^2 + b_i x_2\right)}{b_i^2 + b_i x_3 + x_4} \right]^2, \qquad -5 \le x_i \le 5,$$

$$\min(f_9) = f_9([0.1928 \quad 0.1908 \quad 0.1231 \quad 0.1358]) \approx 0.0003075,$$

where

i	a_i	b_i^{-1}
1	0.1957	0.25
2	0.197	0.5
3	0.1735	1
4	0.1600	2
5	0.0844	4
6	0.0627	6
7	0.0456	8
8	0.0342	10

9	0.0323	12
10	0.0235	14
11	0.0246	16

Function 10: Maxican hat Function

$$f_{10}(\mathbf{x}) = -\frac{\sin(x_1)\sin(x_2)}{x_1 x_2}, \qquad -5 \le x_1, x_2 \le 15, \qquad \min(f_{10}) = \lim_{x \to [0,0]} f_{10}(x) = -1$$

Function 11: Six-Hump Camel Back Function

$$f_{11}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4, \qquad -5 \le x_1, x_2 \le 5,$$

$$\min(f_{11}) = f_{11}([-0.08983, 0.7126]) \approx -1.0316$$

Function 12: Hartman's Family I

$$f_{12}(\mathbf{x}) = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right], \qquad 0 \le x_j \le 1,$$

$$\min(f_{12}) = f_{12}([0.114, \quad 0.556, \quad 0.852]) \approx -3.8628,$$

where

	a_{ij}			p_{ij}		
i \ j	1	2	3	1	2	3
1	3	10	30	0.3689	0.1170	0.2673
2	0.1	10	35	0.4699	0.4387	0.7470
3	3	10	30	0.1091	0.8732	0.5547
4	0.1	10	35	0.03815	0.5743	0.8828

i	c_i
1	1
2	1.2
3	3
4	3.2

Function 13: Hartman's Family II

$$f_{13}(\mathbf{x}) = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{6} a_{ij}\left(x_j - p_{ij}\right)^2 \right], \qquad 0 \le x_j \le 1,$$

$$\min(f_{13}) = f_{13}([0.201 \quad 0.15 \quad 0.477 \quad 0.275 \quad 0.311 \quad 0.627]) \approx -3.32,$$

	a_{ij}						p_{ij}					
j \ i	1	2	3	4	5	6	1	2	3	4	5	6
1	10	3	17	3.5	1.7	8	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.05	10	17	0.1	8	14	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	3	3.5	1.7	10	17	8	0.2348	0.1415	0.3522	0.2883	0.3047	0.6650
4	17	8	0.05	10	0.1	14	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

i	c_i
1	1
2	1.2
3	3
4	3.2

A.1.3. Multimodal Functions with Many Local Minima

Function 14: Generalized Penalized Functions

$$f_{14}(\mathbf{x}) = 0.1\left\{ \sin^2(\pi 23 x_1) + \sum_{i=1}^{29} (x_i - 1)^2 \cdot \left[1 + \sin^2(3\pi x_{i+1})\right] + (x_{30} - 1)^2 \left[1 + \sin^2(2\pi x_{30})\right] \right\}$$
$$+ \sum_{i=1}^{30} u(x_i, 5, 100, 4),$$

$$-50 \le x_i \le 50, \qquad \min(f_{14}) = f(\mathbf{1}) = 0$$

where

$$u(x_i,a,k,m)=\begin{cases} k(x_i-a)^m, & x_i>a, \\ 0, & -a\le x_1\le a, \\ k(-x_i-a)^m, & x_i<a \end{cases}$$

Function 15: Generalized Rastrigin's Function

$$f_{15}(\mathbf{x})=\sum_{i=1}^{30}\left[x_i^2-10\cos(2\pi x_i)+10\right], \qquad -5.12\le x_i\le 10.24, \qquad \min(f_{15})=f_{15}(\mathbf{0})=0$$

Function 16: Generalized Griewank's Function

$$f_{16}(\mathbf{x})=\frac{1}{4000}\sum_{i=1}^{30}x_i^2-\prod_{i=1}^{30}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1, \qquad -1200\le x_i\le 600, \qquad \min(f_{16})=f_{16}(\mathbf{0})=0$$

Function 17: Ackley's Function

$$f_{17}(\mathrm{x})=-20\exp\left(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{30}x_i^2}\right)-\exp\left(\frac{1}{30}\sum_{i=1}^{30}\cos 2\pi x_i\right)+20+e, \qquad -64\le x_i\le 32,$$

$$\min(f_{17})=f_{17}(\mathbf{0})=0$$

Function 18: Schwefel's Function

$$f_{18}(x)=\sum_{i=1}^{30}(x_i\sin(\sqrt{|x_i|})), \qquad -500\le x_i\le 500,$$

$$\min(f_{18})=f_{18}([420.9687, \quad \cdots \quad 420.9687,])=-12569.5$$

REFERENCES

[A.A.E. Ahmed, 2005]

A.A.E. Ahmed, L.T. Germano, and Z.C. Antonio, "A hybrid particle swarm optimization applied to loss power minimization," *IEEE Trans. Power Systems*, vol. 20, no. 2, pp. 859-866, May 2005.

[A. Neubauer, 1997]

A. Neubauer, "A theoretical analysis of the non-uniform mutation operator for the modified genetic algorithm," in *Proc. IEEE Int. Conf. Evolutionary Computation*, 1997, Indianapolis, pp. 93-96.

[A. Vander, 2001]

A. Vander, J. Sherman, and D. Luciano, *Human Physiology*. McGraw-Hill, New York, 2001.

[B. Boser, 1992]

B. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annual Workshop on Computational Learning Theory*, 1992, pp. 144-152.

[B.R. Hunt, 1984]

B.R. Hunt and O. Kubler, "Karhunen-Loeve multispectral image restoration, part 1: theory," *IEEE Trans. Acoust. Speech Signal Process*, 1984, vol. 32, no. 3, pp. 592-600.

[B. Zhao, 2005]

B. Zhao, C.X. Guo, and Y.J. Cao, "A multiagent-based particle swarm optimization approach for optimal reactive power dispatch," *IEEE Trans. Power Systems.*, vol. 20, no. 2, May 2005.

[C.F. Juang, 2004]

C.F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," *IEEE Trans. Syst., Man, Cybern. B, Cybern*., vol. 34, no. 2, pp. 997–1006, Apr. 2004.

[C.R. Reeves, 1994]

C.R. Reeves, "Genetic algorithms and neighbourhood search," *Evolutionary Computing: AISB Workshop*, pp. 115-130, 1994.

[E. Hartuv, 2000]

E. Hartuv, A. Schmitt, J. Lange, S. Meier-Ewert, H. Lehrach, and R. Shamir, "An algorithm for clustering cDNA fingerprints," *Genomics*, vol. 66, no. 3, pp. 249–256, 2000.

[F.Z. Brill, 1998]

F.Z. Brill, D.E. Brown, and W.N. Martin, “Fast genetic selection of features for neural network classifiers,” *IEEE Trans. Neural Networks*, vol. 3, no. 2, pp. 324-328, 1998.

[G. Angelopoulosnd, 1994]

G. Angelopoulosnd and I. Pitas, “Multichannel wiener filters in color image restoration,” *IEEE Trans. CASVT*, vol. 4, no. 1, pp. 83-87, 1994.

[H. Altunbasak, 2001]

H. Altunbasak and H.J. Trussell, “Colorimetric restoration of digital images,” *IEEE Trans. Image Process*, vol. 10, no. 3, pp. 393-402, 2001.

[I. Daubechies, 1992]

I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1992.

[I. Guyon, 2002]

I. Guyon, J. Weston, and S. Barnhill, “Gene selection for cancer classification using support vector machines,” *Machine Learning*, vol. 46, pp. 389-422, 2002.

[J. Bins, 2001]

J. Bins and B. Draper, “Feature selection from huge features sets,” in *Proc. Int. Conf. Computer Vision*, 2001, vol. 2, pp. 159-165.

[J.H. Hong, 2006]

J.H. Hong and S.B. Cho, “Efficient huge-scale feature selection with speciated genetic algorithm,” *Pattern Recognition Letters*, vol. 27, pp. 143-150, 2006.

[J. Kennedy, 1995]

J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, vol. 4, pp.1942-1948, 1995.

[J. Kennedy, 2001]

J. Kennedy and R. Eberhart, *Swarm Intelligence*. Morgan Kaufmann Publishers, 2001.

[J. Khan, 2001]

J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson, and P.S. Meltzer, "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks," *Nature Medicine*, vol. 7, no. 6, pp. 673-679, 2001.

[J.M. Zurada, 1992]

J.M. Zurada, *Introduction to Artificial Neural Systems*. West Info Access, 1992

[K.A. De Jong, 1975]

K.A. De Jong, *The Analysis and Behaviour of a Class of Genetic Adaptive Systems*. Ph.D. thesis, University of Igan, 1975.

[K.J. Boo, 1997]

K.J. Boo and N.K. Bose, "Multispectral image restoration with multisensor," *IEEE Trans. Geosci. Remote Sensing*, vol. 35, no. 5, pp. 1160-1170, 1997.

[L. Li, 2001]

L. Li, C.R. Weinberg, T.A. Darden, and L.G. Pedersen, “Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method,” *Bioinformatics*, vol. 17, no. 12, pp. 1131–1142, 2001.

[L.M. Fu, 2003]

L.M. Fu and E.S. Youn, “Improving reliability of gene selection from microarray functional genomics data,” *IEEE Trans. Information Technology in Biomedicine*, vol. 7, no. 3, pp. 191-196, 2003.

[M. Barni, 2000]

M. Barni, V. Cappellini, and L. Mirri, “Multichannel m-filtering for color image restoration,” *in Proc. ICIP 2000*, Vancouver, Canada, Sept. 2000, pp. 529-532.

[M.B. Eisen, 1998]

M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Bostein, “Cluster analysis and display of genome-wide expression patterns,” *Proceedings of the National Academy of Science*, vol. 95, no. 25, pp. 14863–14868, 1998.

[M. Bittner, 2000]

M. Bittner, P. Meltzer, and Y. Chen, “Molecular classification of cutaneous malignant melanoma by gene expression profiling,” *Nature*, vol. 406(6795), pp. 536-540, 2000.

[M. Daly, 2002]

M. Daly and R. Ozol, “The search for predictive patterns in ovarian cancer: proteomics meets bioinformatics,” *Cancer Cell*, vol. 1, issue 2, pp. 111-112, 2002.

[M.D. Radmacher, 2002]

M.D. Radmacher, L.M. McShane, and R. Simon, “A paradigm for class prediction using gene expression profiles,” *Journal of Computational Biology*, vol. 9, pp. 505-511, 2002.

[M. Mese, 2001]

M. Mese and P.P. Vaidyanathan, “Look-Up Table (LUT) method for inverse halftoning,” *IEEE Trans. Image Process*, vol. 10, pp.1566–1578, Oct 2001.

[M.M. Noel, 2004]

M.M. Noel and T.C. Jannett, “Simulation of a new hybrid particle swarm optimization algorithm,” in *Proc 36th Southeastern Symp. Syst. Theory*, 2004, pp. 150–153.

[M.P. Link, 2002]

M.P. Link, M.C. Gebhardt, and P.A. Meyers, “Osteosarcoma,” in P.A. Pizzo and D.G. Poplack (Eds.), *Principles and Practice of Pediatric Oncology* (4th ed.). Philadelphia, PA: Lippincott, Williams & Wilkins, pp. 1051-1089, 2002.

[M.T. Orchard, 1991]

M.T. Orchard and C.A. Bouman, “Color quantization of images,” *IEEE Trans. Signal Process.*, vol. 39, no. 12, pp. 2677-2690, 1991.

[N.P. Galatsanos, 1989]

N.P. Galatsanos and R.T. Chin, "Digital restoration of multichannel images," *IEEE Trans. Acoust. Speech Signal Process*., vol. 37, pp. 415-421, 1989.

[N.P. Galatsanos, 1991a]

N.P. Galatsanos, A.K. Katsaggelos, R.T. Chin, and A.D. Hillery, "Least squares restoration of multichannel images," *IEEE Trans. Signal Process*, vol. 39, no. 10, pp. 2222-2236, 1991.

[N.P. Galatsanos, 1991b]

N.P. Galatsanos and R.T. Chin, "Restoration of color images by multichannel Kalman filtering," *IEEE Trans. Signal Process*., vol. 39, no. 10, pp. 2237-2252, 1991.

[P.A. Meyers, 1992]

P.A. Meyers, G. Heller, J. Healey, A. Huvos, J. Lane, R. Marcove, A. Applewhite, V. Vlamis, and G. Rosen, "Chemotherapy for nonmetastatic osteogenic sarcoma: the Memorial Sloan-Kettering experience," *Journal of Clinical Oncology*, vol. 10, pp. 5-15, 1992.

[P. Angeline, 1998]

P. Angeline, "Using selection to improve particle swarm optimization," in *Proc. IEEE Int. Conf. Evol. Comput.*, Anchorage, AK, May 1998, pp. 84-89.

[P. Heckbert, 1982]

P. Heckbert, “Color image quantization for frame buffer displays,” *Comput. Graph.*, vol. 16, no. 4, pp. 297-307, 1982.

[R.C. Eberhart, 1998]

R.C. Eberhart and Y. Shi, “Comparison between genetic algorithms and particle swarm optimization,” in *Lecture Notes in Computer Science: Evolutionary Programming VII*. New York: Springer-Verlag, 1998, vol. 1447, pp. 611-616.

[R.C. Eberhart, 2000]

R.C. Eberhart and Y. Shi, “Comparing inertia weights and constriction factors in particle swarm optimization,” in *Proc. Congress on Evolutionary Computing*, Jul. 2000, vol. 1, pp.84-88.

[R. Eberhart, 1995]

R. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” in *Proc. 6th Int. Symp. Micro Machine and Human Science*, Nagoya, Oct. 1995, pp.39-43.

[R.J.M. Vaessens, 1992]

R.J.M. Vaessens, E.H.L. Aarts, and J.K. Lenstra, “A local search template,” *Proceedings of Parallel Problem-Solving from Nature 2*, pp. 65-74, 1992.

[R. Kohavi, 1997]

R. Kohavi and G. John, “Wrappers for feature subset selection,” *Artificial Intelligence*, vol. 97, pp. 273-324, 1997.

[R.S. Gentile, 1990]

R.S. Gentile, E. Walowit, and J.P. Allebach, "Quantization and multi-level halftoning of color images for near original image quality," in *Proc. SPIE 1990*, pp. 249-259.

[R. Ulichney, 1987]

R. Ulichney, *Digital Halftoning*. Cambridge, MA: MIT Press, 1987.

[S.H. Ling, 2007]

S.H. Ling, C.W. Yeung, K.Y. Chan, H.H.C. Iu, and F.H.F. Leung, "A new hybrid particle swarm optimization with wavelet theory based mutation operation," in *Proc. 2007 IEEE Congress on Evolutionary Computation (CEC 2007)*, Singapore, Sep. 25-28, 2007, pp. 1977-1984.

[S. Dudoit, 2002]

S. Dudoit, J. Fridlyand, and T.P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American Statistical Association*, vol. 97(457), pp. 77-87, 2002.

[S. Peng, 2003]

S. Peng, Q. Xu, X.B. Ling, X. Peng, W. Du, and L. Chen, "Molecular classification of cancer types from microarray data using the combination of genetic algorithms and support vector machines," *FEBS Letters*, vol. 555, pp. 358-362, 2003.

[S.S. Dixit, 1991]

S.S. Dixit, "Quantization of color images for display/printing on limited color output devices," *Comput. Graph.*, vol. 15, no. 4, pp. 561-567, 1991.

[T.K. Man, 2005]

T.K. Man, M. Chintagumpala, J. Visvanathan, J. Shen, L. Perlaky, J. Hicks, M. Johnson, N. Davino, J. Murray, L. Helman, W. Meyer, T. Triche, K.K. Wong, and C.C. Lau, "Expression profiles of osteosarcoma that can predict response to chemotherapy," *Cancer Research*, vol. 65, no. 18, pp. 8142-8150, 2005.

[T.M. Huang, 2005]

T.M. Huang and V. Kecman, "Gene extraction for cancer diagnosis by support vector machines - An improvement," *Artificial Intelligence in Medicine*, vol. 35, pp. 185-194, 2005.

[T.R. Golub, 1999]

T.R. Golub, D.K. Slonim, P. Tamayo, C. Hurd, M. GassenBeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Blomfield, and E.S. Lander, "Molecular classification of cancer: class discovery and class prediction by gene-expression monitoring," *Science*, vol. 286, pp. 531-537, 1999.

[T.S. Furer, 2000]

T.S. Furer, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data," *Bioinformatics*, vol. 16, no. 10, pp. 906-914, 2000.

[U. Alon, 1999]

U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, and A.J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and

normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences of the United States of America*, pp. 6745-6750, 1999.

[V.N. Vapnik, 1998]

V.N. Vapnik, *Statistical Learning Theory*. Wiley Interscience, 1998.

[V. Vapnik, 2000]

V. Vapnik and O. Chapelle, "Bounds on error expectation for support vector machines," *Neural Computation*, vol. 12, pp. 2013-2036, 2000.

[X. Wu, 1992]

X. Wu, "Color quantization by dynamic programming and principal analysis," *ACM Trans. Graph.*, vol. 11, no. 4, pp. 372-384, 1992.

[X. Yao, 1999]

X. Yao and Y. Liu, "Evolutionary programming made faster," *IEEE Trans. Evolutionary Computation*, vol. 3, no. 2, pp. 82-102, July 1999.

[Y.D. Zhao, 2003]

Y.D. Zhao, C. Pinilla, D. Valmori, R. Martin, and R. Simon, "Application of support vector machines for T-cell epitopes prediction," *Bioinformatics*, vol. 19, no. 15, pp. 1978-1984, 2003.

[Y.H. Fung, 2004]

Y.H. Fung and Y.H. Chan, “POCS-based algorithm for restoring colour-quantised images,” *IEE Proc. Vision Image and Signal Process*, vol. 151, no. 2, pp.119-127, Feb 2004

[Y.H. Fung, 2006]

Y.H. Fung and Y.H. Chan, "A simulated annealing restoration algorithm for restoring halftoned color-quantized images," *Signal Processing: Image Communication*, vol. 21, issue 4, Apr 2006, pp.280-292.

[Z. Michalewicz, 1994]

Z. Michalewicz, *Genetic Algorithm + Data Structures = Evolution Programs*, 2nd extended ed. Springer-Verlag, 1994.