A particle swarm optimization based support vector machine for digital communication equalizers

ZEY-OU LI,CHI-WEN LI AND YING-REN CHIEN*
Department of Electrical Engineering
University of Ilan
Ilan, Taiwan
e-mail : r9941002@ms.niu.edu.tw , cwli@niu.edu.tw , yrchien@niu.edu.tw*

Abstract—The support vector machine (SVM) is a powerful tool for solving problems with high dimensional, nonlinearly, and is of excellent performance for channel equalization in communication systems. In this study, we propose PSO-SVM as channel equalization. To reconstruct the signal that has the inter symbol interference (ISI) and white Gaussian noise which in high speed communications environments. The SVM parameters will affect the identification of the result. Therefore, we use particle swarm optimization (PSO) to find the suit parameters in SVM. The PSO-SVM to realize the Bayesian equalization solution can be achieved efficiently.

Key-words-support vector machine, particle swarm optimization, channel equalization

1. INTRODUCTION

With the network communication developed, wired transmission cannot be satisfied with the present day. Instead of wired transmission, wireless communication take advantage of not be limited to time and space. High-speed digital data transmission always has the adverse effects of the dispersive channel causing inter symbol interference (ISI) [1] in modern wireless communication systems. This distortion causes considerable loss of information. Channel equalization is the process of compensating for the negative effect of the channel on the transmitted signal and removing the resulting ISI.

Traditionally, channel equalization is equivalent to the process of inverse filtering. According to the view of signal detection, the techniques of channel equalization can be distinguished to sequence-estimation and symbol-by-symbol-decision. The optimal sequence-estimation techniques that be called maximum likelihood sequence estimation (MLSE) are based on Viterbi detector (VD) theorem [2, 3]. Unfortunately, the complexity of this theorem grows exponentially with the dimension of the channel impulsive response.

Symbol-by-symbol-decision equalization based on linear filter design has simply computational requirement but cannot achieve the optimal solution. The optimal symbol-by-symbol-decision equalizer structure without decision feedback can be derived by adopting Bayesian theorem. That is known as maximum a posteriori (MAP) symbol-by-symbol-decision equalizer [4]. The Bayesian equalization can be realized by radial basis function networks (RBFNs) [1]. Symbol-by-symbol-decision equalization can be classified into two categories according to whether they estimate a channel model explicitly. The indirect-modeling equalizers usually using an adaptive linear filter that recovers observed symbols [1], without estimating the channel model explicitly. This approach is widely be used as the equalizer model structure. The structure can only design the decision function to recover the observed symbols. In fact, channel equalization can also be viewed as the classification problem when the equalizer as a decision-making device to reconstruct the transmitted symbol sequence as accurately as possible. The support vector machine has shown that can realize the Bayesian equalization solution [5, 6]. So in this paper, we are applying the classifier support vector machine as the decision-making device.

Support vector machine (SVM) is a new machine learning algorithm developed past years. It's based on statistical learning algorithms have a good generalization performance and widely be used in various applications just like financial, recognition as well as weather forecast [7]. For nonlinear SVM, the mercer kernel technique mapping the nonlinear data to a hyper-plane and classify in the feature space by using a nonlinear function. SVM simulations are implemented non-linear problems previously studied by other researchers using neural networks. The results show that SVM performs as well as neural networks on non-linear problems.

In the support vector machine, a small set of parameters, including trade off variable C and the radius basis kernel function parameters σ^2 typically govern the generalization properties of statistical models [8, 9]. Generally, we get the parameters by a lot of tests but this method may be not the best. In this paper, we are applying particle swarm optimal for selecting support vector machine parameters. The simulation result can assure the validity of it, not only time but also on model accuracy [10].

The rest of this paper are organized as follows. Section 2 present the concept of the technique optimal Bayesian equalizer and symbol-decision equalizer model in communication systems. The SVM decision equalizer is also introduced in this section. The procedure of the PSO based SVM algorithm is proposed in Section 3. Section 4 shows the comparison performance of the proposed method and Bayesian equalizer. Finally has some concluding remark in Section 5.

2. SYMBOL-DECISION EQUALIZER

2.1 Bayesian equalizer

A discrete time model of a digital communications system is considered in this paper is shown in Fig. 1 [1], where the data sequence $s(t)$ is an independent identically distributed sequence of random Binary phase-shift keying (BPSK) and Quadrature phase-shift keying (QPSK), and can define as (1) (2), where $j=\sqrt{-1}$.

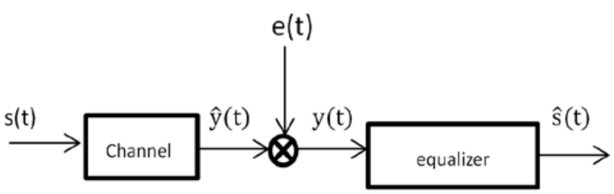


Figure 1. Discrete-time model of data transmission system

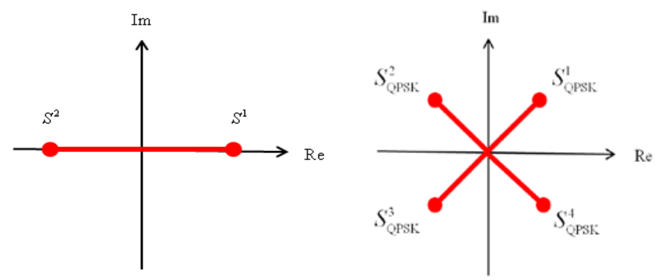
- Binary phase-shift keying (BPSK)

$$s(t)=\mathrm{Re}\left[s(t)\right]+j\,\mathrm{Im}\left[s(t)\right]=\begin{cases}S^1=1,\\ S^2=-1\end{cases} \quad (1)$$

- Quadrature phase-shift keying (QPSK)

$$s_{\mathrm{QPSK}}(t)=\mathrm{Re}\left[s(t)\right]+j\,\mathrm{Im}\left[s(t)\right]=\begin{cases}S^1_{\mathrm{QPSK}}=1+j,\\ S^2_{\mathrm{QPSK}}=-1+j,\\ S^3_{\mathrm{QPSK}}=-1-j,\\ S^4_{\mathrm{QPSK}}=1-j\end{cases} \quad (2)$$

In BPSK and QPSK digital data is represented by 2 points and 4 points around a circle which correspond to 2, 4 phases of the carrier signal. These points are called symbols. Fig. 2 shows this mapping.



(a) binary (b) quadrature

Figure 2. (a) bpsk constellation diagram (b) qpsk constellation diagram

The digital data sequence $s(t)$ is transmitted to a dispersive channel of finite impulse response (FIR). The observed sequence $y(t)$, is including additive white Gaussian noise $e(t)$ with variance σ_n^2. The relationship between input sequence $s(t)$ and observed sequence $y(t)$ can be written as (3)

$$\mathrm{y}(t)=s(t)*h(t)+e(t) \quad (3)$$

The channel impulse response is：

$$h(t)=\sum_{i=0}^{n_h-1}a_i(t)\delta(t-iT_s) \quad (4)$$

The length of the impulse response is n_h, a_i is the gain on channel.

The optimal Bayesian symbol-decision equalizer depicted in Fig. 3 is characterized by the equalizer order m and delay τ [1].

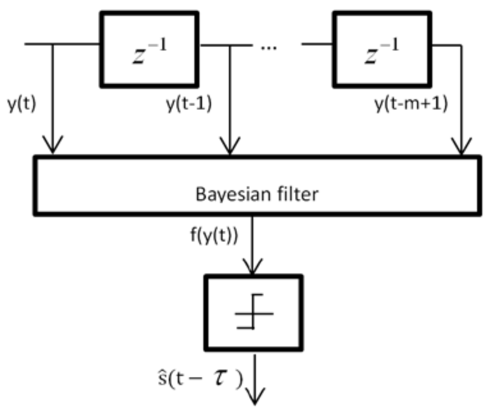


Figure 3. Architecture of symbol-decision equalizer

The complexity of equalizer is determined by the equalizer order m . The Bayesian simply form of the optimal decision function [1] is ：

$$f_B(y(t)) = \sum_{i=1}^{n_s^+} \exp(-\left\| y(t) - y_i^+ \right\|^2 / 2\sigma_e^2) - \sum_{j=1}^{n_s^-} \exp(-\left\| y(t) - y_j^- \right\|^2 / 2\sigma_e^2) \tag{5}$$

With associated boundary

$$f_B(y(t)) = 0 \tag{6}$$

The set of noise free observed symbol is partitioned into two sets conditioned on the transmitted symbol：

$$n_s^+ = \left\{ \hat{y}(t) \middle| s(t-\tau) = 1 \right\} \tag{7}$$

And

$$n_s^- = \left\{ \hat{y}(t) \middle| s(t-\tau) = -1 \right\} \tag{8}$$

An noise free example is given to illustrate in Fig. 4, the channel transfer function $H(z) = 1.0 + 0.8z^{-1} + 0.5z^{-2}$ and for the reason to graphical display, setting the channel order $m = 2$ and decision delay $\tau = 0$.

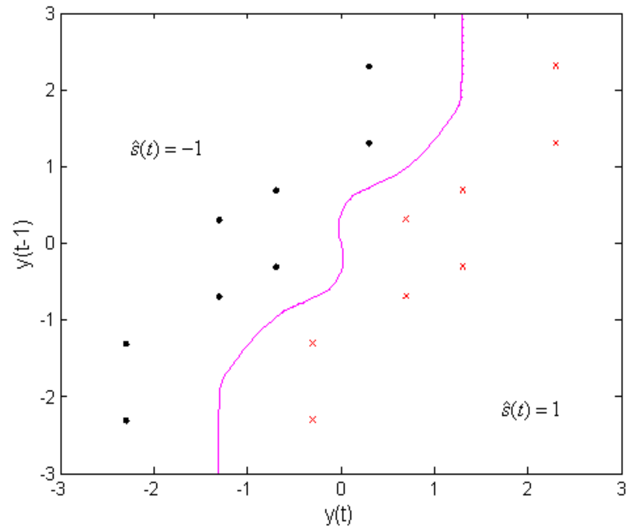


Figure 4. bayesian decision boundaries

2.2 SVM symbol-decision equalizer optimization problem

The simulation results have shown that SVM provides a robust method for channel equalization in wireless communication systems for past researches [11].

For the support vector classification problem, the training set samples consists of vector from pattern space $\mathbf{x}_i \in \Re^d \, (i = 1, 2, \ldots, N)$ and to each vector a classification $y_i \in \{+1, -1\}$. According to Fig. 5, the architecture of SVM symbol-decision equalizer is given an input vector sequence $\mathbf{x}$. The SVM equalizer classifies to (9) [12]

$$\hat{s}(t-d) = sign\{ f_{svm}(x(t)) \} = \begin{cases} 1, & f_{svm}(x(t)) \geq 0 \\ -1, & f_{svm}(x(t)) < 0 \end{cases} . \tag{9}$$

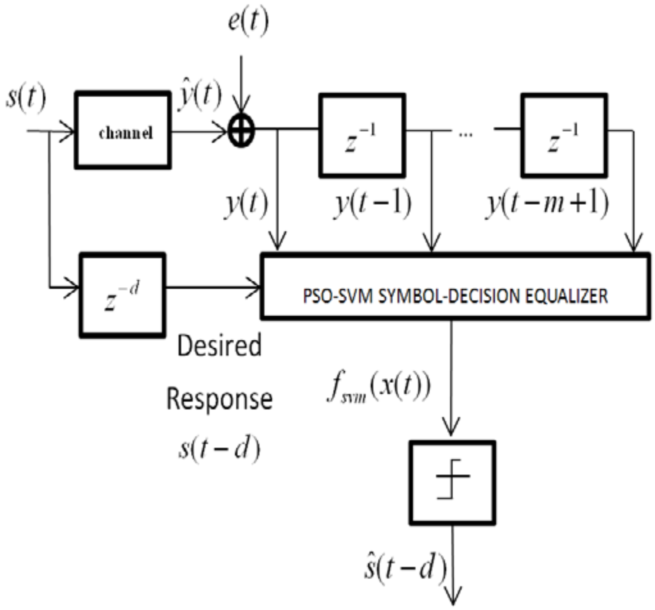


Figure 5. Architecture of SVM symbol-decision equalizer

Where $f_{svm}(x(t))$ is the discriminant function associated with the hyper plane [13, 14] and defined as (10)

$$f_{svm}(x(t)) = \mathbf{w}\cdot\mathbf{x}+b . \quad (10)$$

The normal vector on the hyper-plane is $\mathbf{w}$, $b\in\Re$ is a bias that can separate the two classes without errors. To find the hyper-plane, one should estimate $\mathbf{w}$ and b so that

$$y_i(\mathbf{w}\cdot\mathbf{x}_i+b)>0, \quad \text{with } i=1,2,\ldots,N . \quad (11)$$

The classification methodology of SVMs attempts to separate samples belonging to different classes by tracing maximum margin hyper-plane. To trace maximum the distance is equivalent to minimizing the norm of $\mathbf{w}$. Under constraints (11), finding minimizing the norm of $\mathbf{w}$ can use Lagrange optimization problem to maximize [15]

$$L_D=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j(\mathbf{x_i}\cdot\mathbf{x_j}) .$$

(12)

Subject to:

$$\sum_{i=1}^{m}\alpha_i y_i=0 . \quad (13)$$

$$\alpha_i>0 \ , i=1,2,\ldots,m . \quad (14)$$

This is a quadratic programming (QP) problem that may be solved with traditional optimization techniques [16].

In addition, Cortes added slack variables ξ_i , $i=1,2,\ldots l$ [13]. The goal of finding minimizing the norm of $\mathbf{w}$ become :

$$\min\frac{1}{2}\|\mathbf{w}\|^2+C(\sum_{i=1}^{l}\xi_i) . \quad (15)$$

In this paper, we choose parameter C using particle swarm optimization.

In the equation (12), $(\mathbf{x_i}\cdot\mathbf{x_j})$ influence the performance, and if mapping the data to higher dimension feature space and the problem becomes $\Phi(\mathbf{x_i})\cdot\Phi(\mathbf{x_j})$. So that he inner product can replaced by kernel function:

$$k(x_i,x_j)=(\Phi(\mathbf{x_i})\cdot\Phi(\mathbf{x_j})) . \quad (16)$$

So the optimization nonlinear support vector machine problem can be expressed as

$$L_D=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j k(\mathbf{x_i}\cdot\mathbf{x_j}) . \quad (17)$$

Subject to：

$$\sum_{i=1}^{m}\alpha_i y_i=0 . \quad (18)$$

$$0\le\alpha_i\le C \ , i=1,2,\ldots,m . \quad (19)$$

Then SVM equalizer is described as

$$f_{svm}(x)=\mathrm{sgn}(\sum_{i=1}^{m}\alpha_i y_i k(\mathbf{x}_i\cdot\mathbf{x}_j)+b) . \quad (20)$$

2.3 Kernel options

Applying support vector machine for channel equalization is needed to get the features from training sequence and the kernel function mapping the data to higher dimension feature space. The precise effect of the kernel is still an issue for research [17].

The kernel function needs to satisfy the Mercer's condition [18], the four commonly used families of kernels are:

- Linear kernel

$$k(x_i \cdot x_j) = x_i \cdot x_j \quad (21)$$

- Polynomial kernel with degree d

$$k(x_i \cdot x_j) = (x_i^T x_j + 1)^d . \quad (22)$$

Radial basis function (RBF) kernel (σ is a positive parameters for controlling the radius)

$$k(x_i \cdot x_j) = \exp\left(-\frac{|x_i - x_j|^2}{\sigma^2}\right) . \quad (23)$$

- Sigmoid kernel (g is a positive parameter)

$$k(x_i \cdot x_j) = \tanh(g(x_i \cdot x_j) + c) . \quad (24)$$

In this paper, using radial basis function (RBF) as kernel function [19]. The RBF kernel function can classify nonlinear and high dimension data well, in addition the parameter σ^2 we are using particle swarm optimization to search too.

3. Particle Swarm Optimizer

3.1 Review of the PSO

Particle swarm optimization (PSO) is one kind of population-based optimization evolutionary algorithms (EA) firstly proposed by Kennedy and Eberhart in 1995 [20]. The technique is motivated by the behaviors of flocking birds. PSO is initialized with a group of random particles and searched for optima by updating generations. The particles fly over the solution space, remembering the best solution encountered. The positions and velocity of each particle are updated according to their best encountered position and the best position encountered by any particle according to the following equation [21]:

$$v_{id} = w * v_{id} + c_1 * rand() * (p_{id} - x_{id}) + c_2 * rand() * (p_{gd} - x_{id}) . \quad (25)$$

$$x_{id} = x_{id} + v_{id} . \quad (26)$$

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{k_{\max}} \times k . \quad (27)$$

Where v_{id} is the velocity of every particle in d-dimension, w is inertia weight. c_1 and c_2 are learning parameters, rand() is a random function in the range [0,1]. The flow chart of the PSO is depicted in Fig. 6

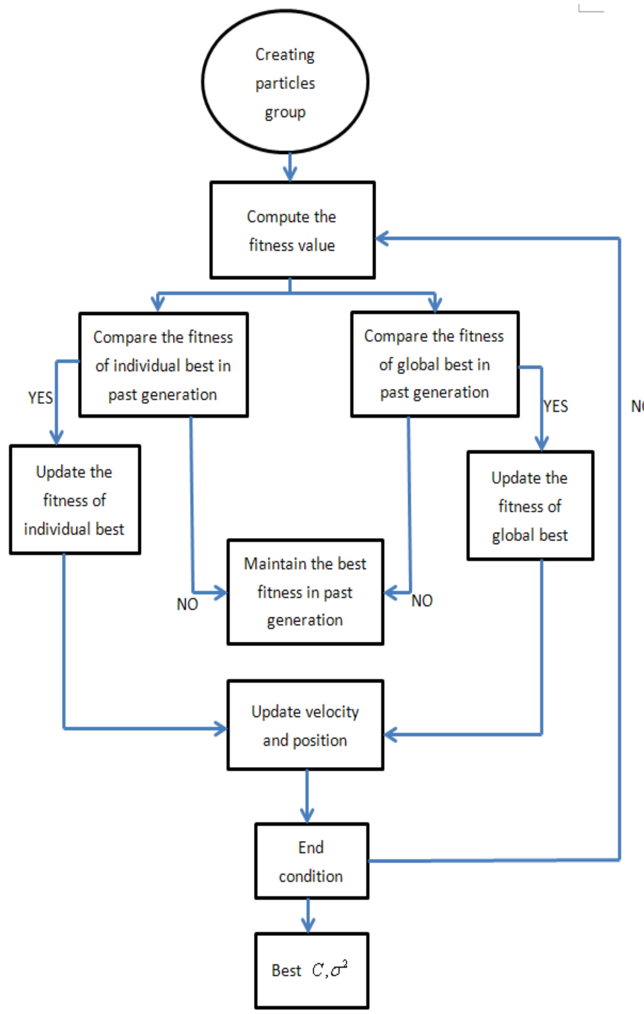


Figure 6. Flow chart of the PSO method

In this paper the fitness value is defined as the k-fold cross-validation accuracy in SVM.

3.2 K-fold cross-validation

In k-fold cross-validation [22], the original sample is randomly partitioned into k subsamples. Of the k subsamples, every single subsample is rotated as the test data and the remaining k-1 subsamples are used as training data. The cross-validation process is then repeated k times, with each of the k subsamples used exactly once as the validation data. The k results from the folds then can be averaged (or otherwise combined) to produce a single estimation. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. 10-fold cross-validation is commonly used. So in this paper, we are applying 10-fold cross-validation average accuracy as the PSO fitness value. Every particle in the flow chart of 10-fold cross-validation is depicted in Fig. 7.

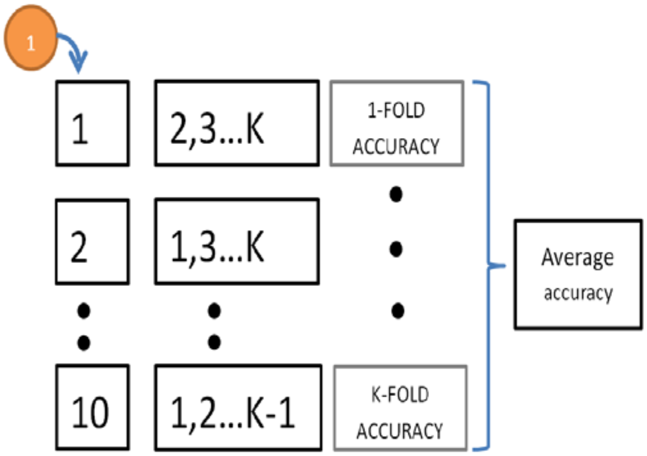


Figure 7. Flow chart of the 10-fold cross-validatio

3.3 Parameter selection

At the beginning of the algorithm we have 20 particles in a group and the initial value of position and velocity are randomly. To select parameters of SVM, every particle has two attributes : the cost C and kernel parameters σ^2 [12]. In this paper we adopt the advised 20 particles in a group, the maximum evolution amount is 100, the learning parameters $c_1 = 1.5$, $c_2 = 1.7$. The searching range of C, σ^2 is [0,100]. According to the PSO concept, more iterative times will have more chance to get the global best value. When w is medium (0.8< w <1.2). The PSO will have the best chance to find the global optimum but also take a moderate number of iterations [23, 24], so we set the inertia weight (0.8< w <1.2). From equation (27), with the iterative times addition the inertia weight will decrease. This method can let the PSO convergence speed up. The flow chart of PSO-SVM structure is depicted in Fig. 8.

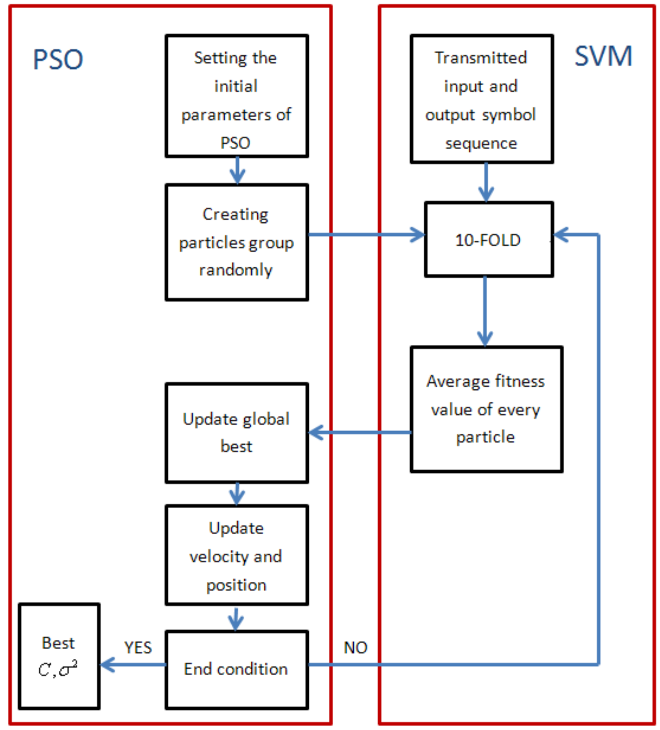


Figure 8. The proposed flow chart of PSO-SVM structure

4. SIMULATION RESULTS

In this section we are simulating a practical application for PSO-SVM. We are considering BPSK symbol over linear, non-linear channel and QPSK over complex channel. According to the simulation performance, we can show that the method of this paper proposed PSO-SVM channel equalizer can realize the optimal Bayesian equalizer performance and the performance even can degrade nearly 1dB at SNR is increased. Due to the advantage of PSO such as simple algorithm structure and good convergence, so PSO provides an effective route for SVM to select parameters.

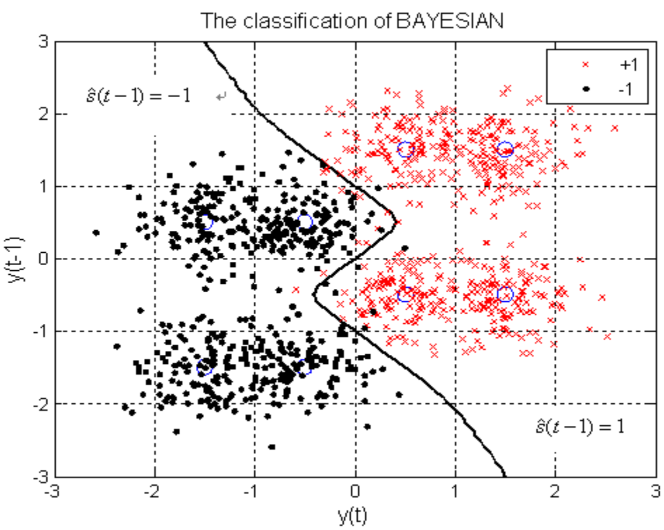
The observed symbol sequence $y(t)$ is a stochastic process having a Gaussian density function with a mean equal to the given state and a variance equal to that of the noise. For the signal to noise ratio (SNR) is 10 dB, the channel output 1000 samples plotted in Fig. 9, Fig. 11 and Fig. 13. The right hand of decision boundaries is class '+1' and is marked by cross 'x' . The left hand of decision boundaries is class '-1' and is marked by solid dots ' • '. The circles are the desired state of the channel output without noise.

- Example 1 BPSK symbol and linear channel

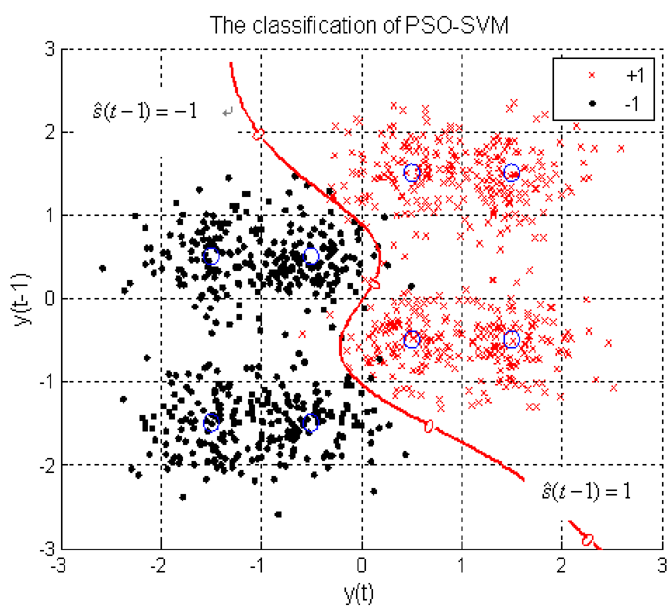
The channel transfer function is defined by

$$H(z) = 0.5 + 1.0z^{-1}, m = 2, \tau = 1$$

Which is adopted in [1]. In Fig. 9 (a), the optimal Bayesian equalizer has the error number 19 in 1000 samples and in Fig. 9 (b) PSO-SVM equalizer only has the error number 13 in 1000 samples. The PSO select parameters $C = 2.2429$ and $\sigma^2 = 0.9497$.



(a)



(b)

Figure 9. Comparison of equalizer decision boundaries. Channel transfer function is $H(z) = 0.5 + 1.0z^{-1}$, $m = 2, \tau = 1$. Constellation of 1000 channel output samples for SNR=10dB (a) Decision boundaries of Bayesian equalizer (b) Decision boundaries of PSO-SVM

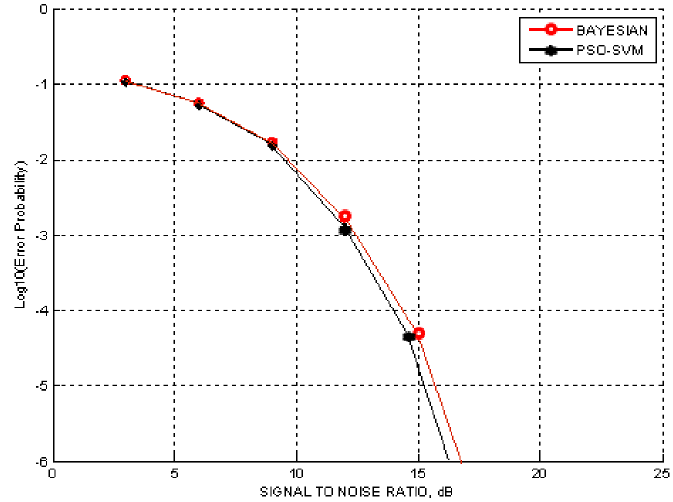


Figure 10. The comparison of performance in channel transfer function

$H(z) = 0.5 + 1.0z^{-1}$, $m = 2, \tau = 1$

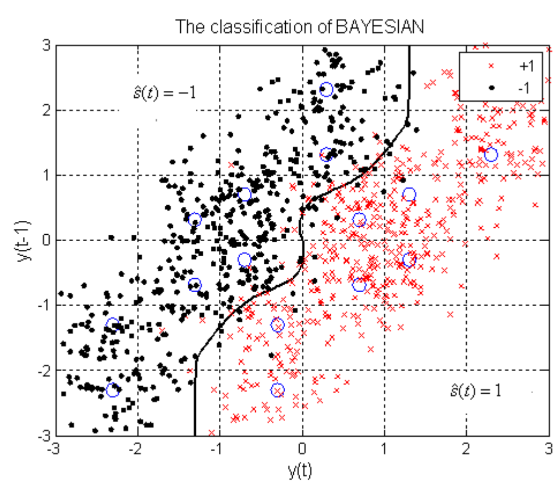
The comparison of performance in channel transfer function $H(z) = 0.5 + 1.0z^{-1}, m = 2, \tau = 1$ is depicted in Fig. 10.

- Example 2 BPSK symbol and linear channel

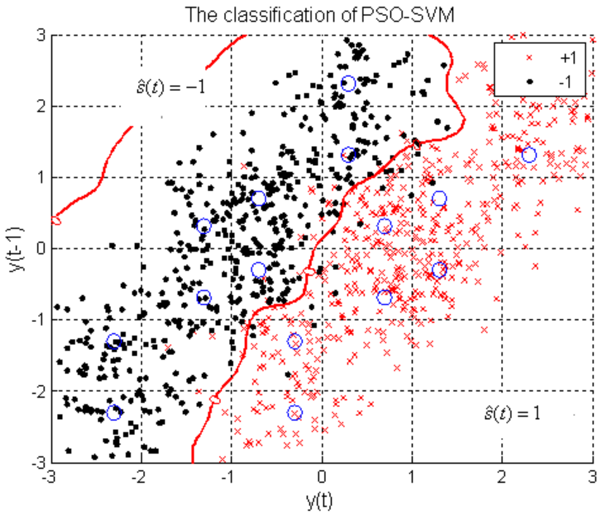
The channel transfer function is defined by

$$H(z) = 1.0 + 0.8z^{-1} + 0.5z^{-2}, m = 2, \tau = 0$$

Which is adopted in [1]. In Fig. 9 (a), the optimal Bayesian equalizer has the error number 54 in 1000 samples and in Fig. 9 (b) PSO-SVM equalizer only has the error number 45 in 1000 samples. The PSO select parameters $C = 0.8847$ and $\sigma^2 = 0.7483$.



(a)



(b)

Figure 11. Comparison of equalizer decision boundaries. Channel transfer function is $H(z)=1.0+0.8z^{-1}+0.5z^{-2}$ $, m=2, \tau=0$. Constellation of 1000 channel output samples for SNR=10dB (a) Decision boundaries of Bayesian equalizer (b) Decision boundaries of PSO-SVM

The comparison of performance in channel transfer function $H(z)=1.0+0.8z^{-1}+0.5z^{-2}, \tau=0$ is depicted in Fig. 12.

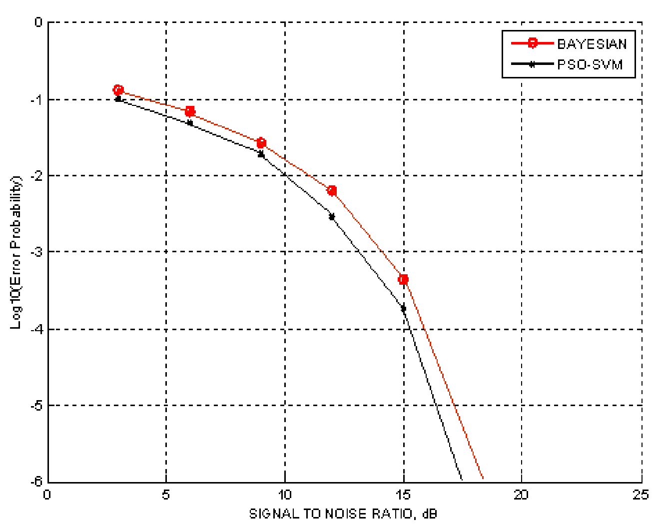


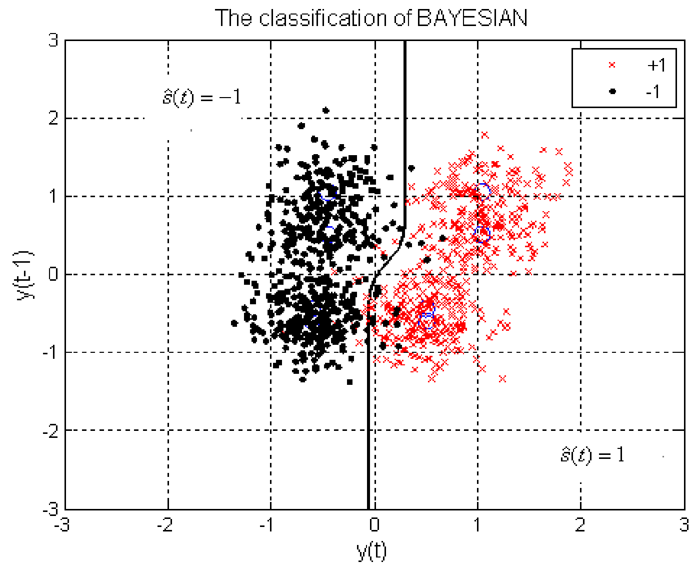
Figure 12. The comparison of performance in channel transfer function $H(z)=1.0+0.8z^{-1}+0.5z^{-2}, \tau=0$

- Example 3 BPSK symbol and non-linear channel

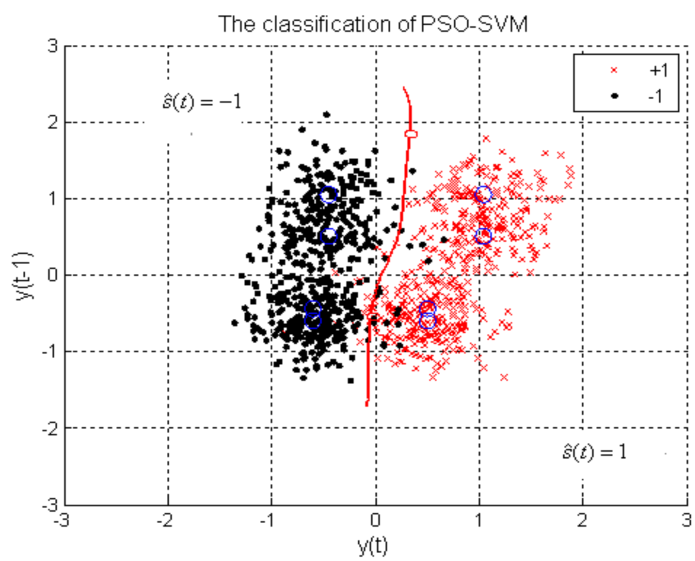
The channel transfer function is defined by

$$x(t)=\hat{y}(z)/s(z)=g(t)+0.5g(t-1)$$
$$\hat{y}(t)=x(t)+0.1x^2(t)-0.2x^3(t) \qquad ,m=2,\tau=0$$

Which is adopted in [1]. In Fig. 13 (a), the optimal Bayesian equalizer has the error number 30 in 1000 samples and in Fig. 13 (b) PSO-SVM equalizer only has the error number 28 in 1000 samples. The PSO select parameters $C=0.7015$ and $\sigma^2=5$.



(a)



(b)

Figure 13. Comparison of equalizer decision boundaries. Channel transfer function is $x(t)=\hat{y}(z)/s(z)=g(t)+0.5g(t-1)$, $\hat{y}(t)=x(t)+0.1x^2(t)-0.2x^3(t)$ $,m=2,\tau=0$. Constellation of 1000 channel output samples for SNR=10dB (a)

Decision boundaries of Bayesian equalizer (b) Decision boundaries of PSO-SVM

The comparison of performance in channel transfer functions

$$x(t) = \hat{y}(z)/s(z) = g(t) + 0.5g(t-1)$$

$$\hat{y}(t) = x(t) + 0.1x^2(t) - 0.2x^3(t) \qquad ,m = 2, \tau = 0$$ is depicted in Fig. 14.

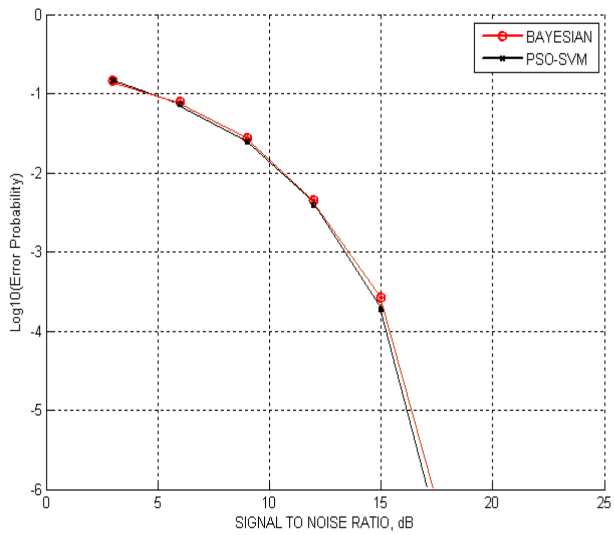


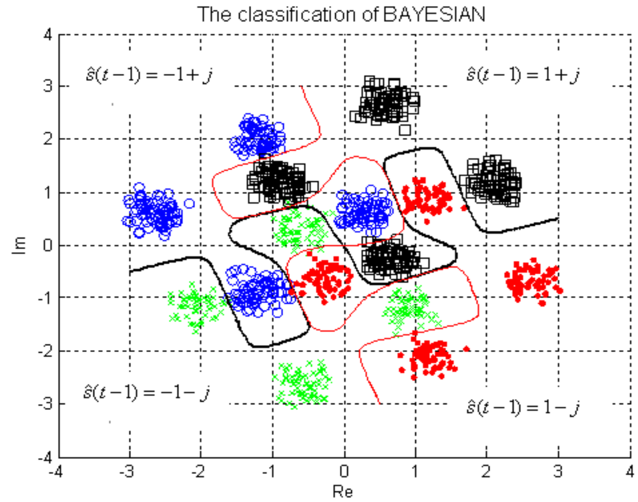
Figure 14. The comparison of performance in channel transfer function $x(t) = \hat{y}(z)/s(z) = g(t) + 0.5g(t-1)$ $\hat{y}(t) = x(t) + 0.1x^2(t) - 0.2x^3(t) \quad ,\tau = 0$

- Example 4 QPSK symbol and complex channel

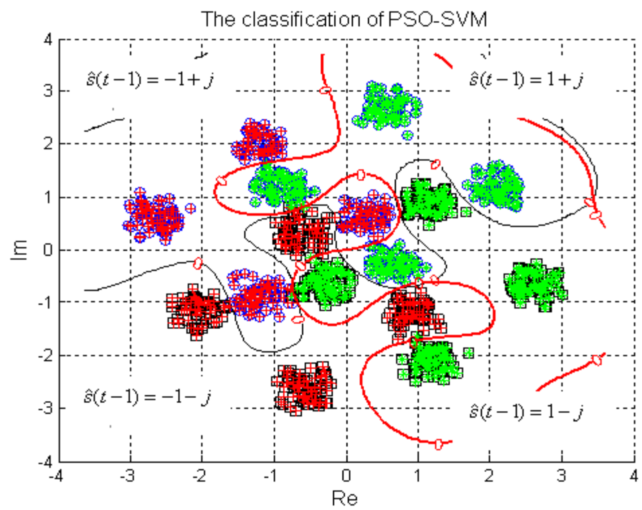
The channel transfer function is defined by

$$H(z) = (0.7409 - 0.7406i) \times (1 - (0.2 - 0.1i)z^{-1}) \times (1 - (0.6 - 0.3i)z^{-1}) \; m = 1, \tau = 1$$

In the complex channel the decision boundaries is partitioned into real part and imaginary part. For the signal to noise ratio (SNR) is 15 dB, the same 1000 samples is plotted in Fig. 15 In Fig. 15 (a), the optimal Bayesian equalizer has the error number 11 in 1000 samples and in Fig. 15 (b). PSO-SVM equalizer only has the error number 6 in 1000 samples. The PSO select parameters in real boundaries $C_1 = 1.0844$ and $\sigma_1^2 = 1.0008$, and in imaginary boundaries $C_2 = 2.6815$ and $\sigma_2^2 = 1.2747$.



(a)



(b)

Figure 11. Comparison of equalizer decision boundaries. Channel transfer function is $H(z) = (0.7409 - 0.7406i) \times (1 - (0.2 - 0.1i)z^{-1}) \quad (1 - (0.6 - 0.3i)z^{-1})$ $m = 1, \tau = 1$. Constellation of 1000 channel output samples for SNR=15dB (a) Decision boundaries of Bayesian equalizer (b) Decision boundaries of PSO-SVM

$$H(z) = (0.7409 - 0.7406i) \times (1 - (0.2 - 0.1i)z^{-1}) \times (1 - (0.6 - 0.3i)z^{-1}) \; m=1, \tau=1$$

The comparison of performance in channel transfer functions $H(z) = (0.7409 - 0.7406i) \times (1 - (0.2 - 0.1i)z^{-1})$ $\times (1 - (0.6 - 0.3i)z^{-1}) \; m = 1, \tau = 1$ is depicted in Fig. 12.

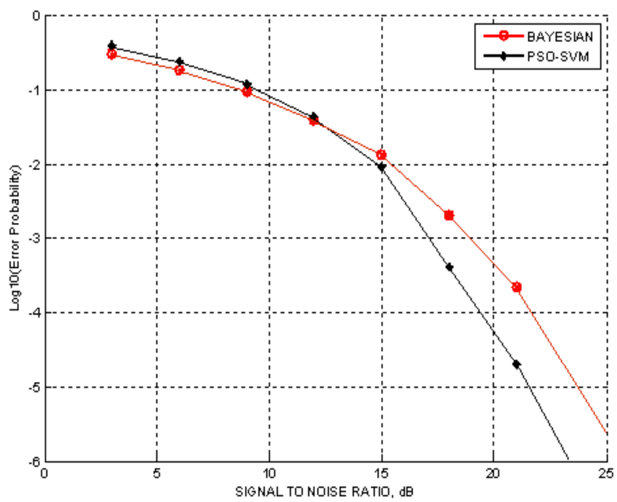


Figure 12. The comparison of performance in channel transfer function $H(z)=(0.7409-0.7406i)\times(1-(0.2-0.1i)z^{-1})\times(1-(0.6-0.3i)z^{-1})$ $m=1, \tau=1$

5. CONCLUSIONS

This paper constructs the PSO-SVM equalizer on the linear, non-linear channel and complex channel successfully. The PSO-SVM equalizer simulations results have shown better than Bayesian equalizer. Due to the advantage of PSO such as simple algorithm structure and good convergence, so PSO provides an effective route for SVM to select parameters.

ACKNOWLEDGMENTS

This work was supported in part by National Science Council, R.O.C., under Grant NSC 101-2218-E-197-001.

REFERENCES

[1] Chen S., Mulgrew B., Grant P.M.: A clustering technique for digital communications channel equalization using radial basis function networks. IEEE Transactions on Neural Networks, 4(4), pp. 570-590, 1993.

[2] Qureshi S.U.H.: Adaptive equalization. IEEE Proceedings, 73(9), pp. 1349-1387, 1985

[3] Forney G. Jr.: Maximum-likelihood sequence estimation of digital sequences in the presence of inter symbol interference. IEEE Transactions on Information Theory, 18(3): p. 363-378, 1972

[4] Tuchler M., Singer A.C., Koetter R.: Minimum mean squared error equalization using a priori information. IEEE Transactions on Signal Processing, 50(3): p. 673-683, 2002.

[5] Sebald D.J., Bucklew J.A.: Support vector machine techniques for nonlinear equalization, IEEE Transactions on Signal Processing, 48(11): p. 3217-3226, 2000.

[6] Chwenker F.: Hierarchical support vector machines for multi-class pattern recognition. Proceedings of Int. Conf in Knowledge-Based Intelligent Engineering Systems and Allied Technologies. pp. 561-565, 2000.

[7] Mori H., Kanaoka D.: Application of Support Vector Regression to Temperature Forecasting for Short-term Load Forecasting. International Joint Conference on Neural Networks, pp. 1085-1090 ,2007

[8] Ying-Chun G.:, An integrated PSO for parameter determination and feature selection of SVR and its application in STLF, International Conference on Machine Learning and Cybernetics, pp. 359-364, 2009.

[9] Thadani, K., Ashutosh, Jayaraman, V.K., Sundararajan, V.: "Evolutionary Selection of Kernels in Support Vector Machines," International Conference on in Advanced Computing and Communications, pp. 19-24, 2006.

[10] Shi Y., Eberhart R.: A modified particle swarm optimizer. IEEE World Congress on Computational Intelligence in Evolutionary Computation Proceedings, pp. 69-73, 1998.

[11] Chen S., Samingan A.K., Hanzo L.: Optimal decision feedback equaliser for M-PAM signals using support vector machine solution. Electronics Letters, 36(20), p. 1742-1744, 2000.

[12] Sebald D.J., Bucklew J.A.: Support vector machine techniques for nonlinear equalization. IEEE Transactions on Signal Processing, 48(11): p. 3217-3226, 2000.

[13] Hsu C. W., Chang C. C., Lin C. J.: A practical guide to support vector classification. [web page] http://www.csie.ntu.tw/~cjlin/papers/guide/guide.pdf. [Accessed on 5 Jul. 2012]

[14] Burges C.J.C:. A tutorial on support vector machines for pattern recognition. Data Mining and knowledge Discover. 2(2);pp. 955-974, 1998.

[15] Cortes C., Vapnik V.:. Support vector networks. Machine Learning, 20, pp. 273-297, 1995.

[16] Kaufman L.: Solving the Quadratic Programming Problem Arising in Support Vector Classification. Advance in Kernel Methods-Support Vector

Learning, pp.147-167, Cambridge, MA, 1999, MIT Press

[17] Ying T., Jun W.: A support vector machine with a hybrid kernel and minimal Vapnik-Chervonenkis dimension. IEEE Transactions on Knowledge and Data Engineering, 16(4), p. 385-395, 2004.

[18] Suykens J.A.K., Vandewalle J.: Recurrent least squares support vector machines, IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 47(7), p. 1109-1114, 2000.

[19] Hsu C.W. , Chang C. C., Lin C. J.: A practical guide to support vector classification. [web page] :http://www.csie.ntu.tw/~cjlin/papers/guide/guide.pdf. [Accessed on 5 Jul. 2012]

[20] Eberhart R., Kennedy J.: A new optimizer using particle swarm theory. International Symposium on Micro Machine and Human Science,pp. 39-43, 1995.

[21] J. Kennedy J., Eberhart R. C.: Empirical study of particle swarm optimization, Proceedings of the Congress Proceedings of the Congress on Evolutionary Computation, (3), pp. 1945-1950, 1999.

[22] Stone M.:, Cross-Validatory Choice and Assessment of Statistical Predictions, Journal of the Royal Statistical Society B, 36(1), 1974, pp111-147

[23] Sebald D.J., Bucklew J.A.:, Support vector machine techniques for nonlinear equalization. IEEE Transactions on Signal Processing, 48(11): p. 3217-3226, 2000.

[24] PSO Tutorial, Online Available [web page] http://www.swarmintelligence.org/index.php. [Accessed on 1 Jan 2013]