RESEARCH ARTICLE

Support vector machine-based equalisation for direct-sequence ultra wideband systems

Mohamed Musbah and Xu Zhu*

Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, U.K.

ABSTRACT

We proposed the support vector machine (SVM)-based equalisation schemes for direct-sequence ultra wideband (UWB) systems. The severe intersymbol interference caused by the UWB channel was formulated as a pattern classification problem in the SVM-based equaliser, which operates in two main modes: training and detection. We also applied the least squares support vector classifiers (LS-SVCs) to reduce the training complexity and sparse LS-SVCs to reduce the detection complexity, with little performance loss compared to SVCs. Simulation results confirm the outperformance of the proposed equalisers over the conventional rake receiver with the same order of complexity for detection, especially when no channel information is known at the receiver. Also, the SVM-based equalisers in the line-of-sight scenario provide a performance close to the case with additive white Gaussian noise only.

KEYWORDS

Ultra wideband (UWB); support vector machines; channel equalisation

*Correspondence

Xu Zhu, Department of Electrical Engineering and Electronics, University of Liverpool, Brownlow Hill, Liverpool L69 3GJ, U.K.
E-mail: xuzhu@liv.ac.uk

1. INTRODUCTION

Ultra wideband (UWB) systems [1] have been proposed as an air interface to the physical layer of wireless personal area networks in the IEEE 802.15 standards because of their high data rate capability in the licence-free spectrum [2]. The direct-sequence (DS) UWB system combines the impulse-based transmission [3] and the DS code division multiple access technology [4]. It has a fine path resolution by transmitting information symbols represented by a sequence of ultrashort pulses of predefined shape and allows the transmission of lower power chips and the capability of asynchronous data traffic.

The channel characteristics in the UWB spectrum of operation suffer from severe intersymbol interference (ISI), which, accordingly, causes tremendous degradation of the overall system performance. Therefore, a proper equalisation technique is required to mitigate the ISI effects. In [5], a rake receiver was applied to collect and exploit multipath components for UWB channels. In [6], a frequency domain equaliser was developed for DS-UWB systems. A combination of the rake receiver and the linear minimum mean square error equaliser was investigated in [7] and [8] but with the assumption of perfect channel knowledge at the receiver. However, the system performance in the previous work was unsatisfactory for low-to-medium signal-to-noise ratio (SNR) range, which corresponds to the assigned power level for UWB systems.

The applications of statistical learning techniques have attracted many researchers. Support vector machines (SVMs) [9,10] are one of the promising learning techniques that have been used to solve the classification problem, by using support vector classifiers (SVCs) [11]. A number of modifications and extensions to conventional SVCs have been developed to reduce the complexity of the training process, such as online SVC training [12] and least squares SVC (LS-SVC) [13]. SVMs have been used for UWB systems in range estimation and positioning applications [14]. In digital communications, however, little work has been reported in the literature on applications of the SVM technique. Mostly, as a pattern recognition solution, SVCs have been applied for channel equalisation, channel estimation [15,16] and multi-user detection [17]. However, most of the previous works only considered simple theoretical channel models (e.g. short FIR models) that represent less practical scenarios.

In this paper, we investigate the SVM-based equalisation for DS-UWB systems by employing SVC, LS-SVC and sparse LS-SVC [18]. This was conducted by employing a bank of independent SVM-based classifiers with optimised
parameters at the receiver in a block-by-block fashion. To the best of our knowledge, this is the first work to apply the SVM technique to UWB communication systems considering practical UWB channels and scenarios. Simulation results show that the SVM-based equalisers outperform conventional rake receivers in terms of bit error rate (BER) especially for the line-of-sight (LOS) scenario. We also apply the LS-SVC based equalisation, in order to reduce the training computational complexity of SVCs without sacrificing the performance. Furthermore, by introducing the imposed sparseness to the LS-SVC, the detection complexity is reduced, while the same level of BER performance is maintained.

The rest of the paper is organised as follows. The system model is presented in Section 2. In Section 3, the conventional rake receiver for DS-UWB systems is reviewed. The SVM-based equalisers for DS-UWB systems are proposed in Section 4. Performance analysis and complexity analysis are provided in Sections 5 and 6, respectively, and simulation results are shown in Section 6. Section 7 presents simulation results. The conclusion is drawn in Section 8.

2. SYSTEM MODEL

2.1. Transmitted signal

Figure 1 illustrates the overall system model. The transmit signal is designed so that the information symbols \( b_m \in \{-1, +1\} (m = 1, \ldots, M) \), which are of unit energy, are arranged in blocks of length \( M \) by a serial-to-parallel converter for spreading by a ternary code. Then a guard interval (GI) of \( L_{GI} \) zero symbols is prefixed to each block, in order to mitigate the effect of interblock interference. A whole transmission session is represented by a packet, which consists of \( P \) pilot blocks for training and \( B \) data blocks for detection. The packet structure is illustrated in Figure 1(a). The \( j \)th extended (GI-inserted) block is expressed as \( b_{GI}(j) = [0^T_{L_{GI}} b_1 \ldots b_M]^T \) where \( \theta_j \) denotes an all-zero column vector of length \( l \). Discrete-time signal representation will be used throughout the paper.

Considering a single user transmission, we use a ternary spreading code \( s = [s_1 s_2 \ldots s_{N_c}]^T \) where \( s_k \in \{-1, 0, 1\} \) \((k = 1, \ldots, N_c)\) and \( N_c \) is the spreading code length (in chips). The \( j \)th block after spreading is given by

\[
b_{DS}(j) = [\theta^T_{N_c L_{GI}} b_1 s^T \ldots b_M s^T]^T
\]

2.2. Ultra wideband indoor channel model

The UWB channel can be characterised by severe multipath with a very slow time variation. The IEEE 802.15.3a [19], based on extensive laboratory measurements, has proposed a channel model with four possible scenarios, which is widely used in evaluating the performance of the UWB communication systems.

The channel model recommended by the IEEE 802.15.3a channel modelling subcommittee is based on measurements in the 2–8-GHz band and thus has a delay resolution of \( T_c = 0.167 \text{ ns} \), and the channel impulse response (CIR) is expressed as

\[
h(t) = \sum_{i=1}^{C_L} \sum_{k=1}^{K} a_{i,k} \delta(t - T_i - \tau_{i,k})
\]

where \( T_i \) is the delay of the \( i \)th \((i = 1, \ldots, C_L)\) cluster and \( \tau_{i,k} \) is the delay of the \( k \)th \((k = 1, \ldots, K)\) path within the \( i \)th cluster relative to \( T_i \). \( T_i \) and \( \tau_{i,k} \) are described with a double-Poisson process and are rounded to integer multiples of the delay resolution \( T_c \). \( a_{i,k} = p_{i,k} \xi_{i,k} \) is the path gain of the \( k \)th path within the \( i \)th cluster, where \( p_{i,k} \in \{+1, -1\} \) denotes the equally likely random polarity (the possible phases for real coefficients) and the fading amplitude \( \xi_{i,k} \) is real valued and follows the lognormal distribution [20].

With \( \tau_{ex} \) denoting the multipath delay spread, \( L = \tau_{ex}/T_c \) is the total number of paths and \( h_l \) is the sum of all \( a_{i,k} \) at time index \( l \) where \( l = [T_i + \tau_{i,k}/T_c] \). Because of the clustering of multipath components [21], the channel does not necessarily have multipath arrivals within each delay bin. This is accounted for by setting \( h_{1} = 0 \) for any \( l/T_c \) that has no path arrival. Therefore, the CIR of this model can be simplified to

\[
h(t) = \sum_{l=1}^{L} h_l \delta(t - (l - 1)T_c)
\]

This discretised model that represents the CIR can be expressed by an \( N_c(L_{GI} + M) \times N_c(L_{GI} + M) \) matrix \( H \) [22] as follows:

\[
H = \begin{bmatrix}
h_1 & 0 & 0 & \cdots & 0 \\
\vdots & h_1 & 0 & \cdots & 0 \\
h_L & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & h_1 & 0 & 0
\end{bmatrix}
\]

2.3. Received signal

For block-by-block transmission, the discrete-time form of the \( j \)th received signal block can be expressed as

\[
r_{GI}(j) = H b_{DS}(j) + v(j)
\]

where \( v(j) \) is an additive white Gaussian noise (AWGN) vector whose elements are independent Gaussian random

\[\text{DOI: 10.1002/wcm}\]
variables with zero mean and variance $\sigma^2$. The GI samples (the first $N_cL_{GI}$ chips of each block) are then removed from $\mathbf{r}_{GI}$ so that the $j$th received signal can be defined as

$$\mathbf{r} = [\mathbf{r}_{GI}[N_cL_{GI} + 1] \cdots \mathbf{r}_{GI}[N_c(L_{GI} + M)]]^T$$  \hspace{1cm} (6)

where $\tau_{f_l}$ is the delay time of the $l$th strongest path gain within one symbol duration. The combined output of the $L_f$-finger rake receiver for the $j$th symbol can be expressed as follows:

$$\tilde{b}_j = \tilde{\mathbf{y}}^T \mathbf{z}_j$$  \hspace{1cm} (8)

where $\mathbf{z}_j = [z_{f_1}^j, \ldots, z_{f_{L_f}}^j]^T$, and $\tilde{\mathbf{y}} = [\tilde{y}_1, \ldots, \tilde{y}_{L_f}]^T$ is the finger weight vector of the rake receiver. Based on the MRC criterion, $\tilde{y}_l$ is given by

$$\tilde{y}_l = \hat{h}_{f_l}$$  \hspace{1cm} (9)

where $\hat{h}_{f_l}$ is the estimate of the $l$th strongest path gain. The estimated symbol is then determined by the decision function as

$$\hat{b}_j = \text{sign}(\tilde{b}_j) = \begin{cases} +1, & \tilde{b}_j \geq 0 \\ -1, & \tilde{b}_j < 0 \end{cases}$$  \hspace{1cm} (10)

3.2. Channel estimation

In this study, a data-aided approach [24] was used to estimate the channel impulse response. The general sliding
correlator method [25,26] was employed for channel estimation. This is accomplished by sending \( P \) known pilot symbols for the training \( b_j^f \), \( j = 1, 2, \ldots, P \). The rake receiver, during the training, gives the output signal vector \( z_{\text{est}}^j = [z_1^j, z_2^j, \ldots, z_{\text{est}}^j]^T \) where \( L_{\text{est}} \) is the number of paths to be estimated, and it is assumed that the receiver knows the optimal value of \( L_{\text{est}} \), that is, \( L_{\text{est}} = L \) [27]. By applying the cross-correlation method, the estimated path gains of the channel vector (\( h \)) can be expressed as follows:

\[
\hat{h} = \frac{1}{P} \sum_{j=1}^{P} b_j^f z_j^{\text{est}} \tag{11}
\]

where \( \hat{h} = [\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_{L_{\text{est}}}]^T \).

### 4. SUPPORT VECTOR MACHINE-BASED EQUALISERS FOR DIRECT-SEQUENCE ULTRA WIDEBAND SYSTEMS

In this section, we apply SVMs to DS-UWB systems by solving equalisation as a pattern recognition problem in the spreading code space. We propose three types of classifiers to develop the SVM-based receiver system, namely, SVC, LS-SVC and sparse LS-SVC. The \( M \) classifiers are arranged in parallel as shown at the receiver in Figure 1(b).

#### 4.1. General support vector machine model for equalisation

Support vector machines are a set of related supervised learning methods used for classification and regression. For classification applications, they belong to a family of generalised linear classifiers that separate data space (attribute space) by an appropriate hyperplane. A special property of SVMs is that they simultaneously minimise the empirical classification error and maximise the geometric margin in the training mode; hence, they provide high-level classification performance [11]. SVM classifiers are also known as the maximum margin classifiers [28]. For nonlinear patterns, the attribute space (data input space) is mapped into a higher dimensional space (feature space). The inner products of feature space can be evaluated, via kernel functions [11], from the attribute space without knowledge of the mapping manner.

For channel equalisation applications, SVMs can be applied to detect some distorted signal to retrieve the original transmitted symbols as a pattern classification problem. This is implemented by training the SVM classifiers with known distorted signal (pilot) so that the classifier parameters are set to compensate for the channel characteristics in the detecting mode. A major advantage that distinguishes SVM from typical rake receiver is that SVM methods work on signals on a nonlinear domain unlike the rake receiver that works as a linear combiner to detect the paths.

In our system, the received discrete signal in Equation (6) is arranged in \( M \) parallel groups of \( N_c \) chip length each, as shown in the receiver part of the system. The signal is passed through \( M \) classifiers, and the input vector to the \( m \)th classifier, \( x_j^m \) classifier, the \( m \)th pilot symbol in each of the \( P \) training blocks are used to the classifier training. The \( m \)th pilot symbol the \( i \)th training block is given by \( y_j^m = b_{m,j} \).

In the testing (detection) mode, the estimated \( m \)th symbol of the \( i \)th block \( (i = 1, \ldots, P) \) received data block is obtained from the \( m \)th SVM-based classifier as

\[
\hat{x}_m(i) = \text{sign}(f_{\text{SVM}}^{(m)}(x_i^m)) \tag{12}
\]

where sign is a decision function defined in Equation (10) and \( f_{\text{SVM}}^{(m)} \) is the classification function of the \( m \)th classifier, which is defined as

\[
f_{\text{SVM}}^{(m)}(x_i^m) = \sum_{j \in SV} \alpha_j^{(m)} y_j^{(m)} K(x_i^m, x_j^m) + \beta^{(m)} \tag{13}
\]

where \( SV \) is a set of support vectors’ indices. \( x_j^{(m)} \) and \( y_j^{(m)} \) are the support vectors and their labels, respectively, which can be partial or full of the training vectors. \( \alpha \) are Lagrange multipliers that are calculated in the training mode via training input vectors with a proper optimisation scheme. \( \beta^{(m)} \) in Equation (13) is a threshold term that indicates how far the origin is from the hyperplane (or affine offset [16]). For simplicity of notation, the classifier’s index \( m \) will be omitted in subsequent subsections because the optimisation treatment applies to all classifiers in the receiver.

\[
K(x_a, x_b) = \varphi(x_a)\varphi(x_b) \tag{14}
\]

where \( \varphi(x) \) is the function to transform the input vector \( x \) to the higher dimensional feature space. It is necessary for a mapping function to be a kernel to satisfy Mercer’s condition [29]. Kernel functions have many forms depending on the nature of the pattern to be classified. The most common form of kernel functions in communication applications is the Gaussian radial basis function [15], which can be expressed as

\[
K(x_a, x_b) = e^{-\frac{|x_a - x_b|^2}{2\sigma^2}} \tag{15}
\]

where \( \sigma \) is the width parameter. The problem of finding the optimum values for kernel parameters is concerned by
many researchers [30]. However, some empirical validation tests can be used to evaluate the optimum $\sigma$ value, such as leave one out cross-validation algorithm [31] and its modified version and leave one support vector out cross-validation algorithm [32].

### 4.2. Support vector classifier-based equaliser

A powerful advantage of SVMs is that only some of the training vectors, referred to as support vectors, are used in the classification stage.

The fundamental principle of SVC is to find a linear hyperplane ($w$) in higher dimensional space that maximise the distance (margin) between two different patterns. Hence, the optimisation problem in its primal form is defined as

$$
\text{Minimise } W_p(w, \xi) = \frac{1}{2}w^Tw + C \sum_{i=1}^{P} \xi_i
$$

Subject to $y_j[w^T\phi(x_j) + \beta] \geq 1 - \xi_j, \ j = 1, 2, \ldots, P$ \hspace{1cm} (16)

This optimisation problem is convex, so that the dual gap could be zero. The dual form of Equation (16) can be obtained by introducing Lagrange multipliers ($\alpha$s) and applying Karush–Kuhn–Tucker conditions [33] for optimisation of a constrained function [11], the primal objective function with its constraints in Equation (16) is converted to dual formulation. The optimisation process, according to [9], is then to find the values of $\alpha$s of the $m$th classifier that maximise the resulting dual objective function as follows:

$$
\text{Maximise } W_d(\alpha) = \sum_{i=1}^{P} \alpha_i - \frac{1}{2} \sum_{i=1}^{P} \sum_{j=1}^{P} \alpha_i \alpha_j y_i y_j K(x_i, x_j)
$$

Subject to $\sum_{i=1}^{P} \alpha_i y_i = 0$ \hspace{1cm} $0 \leq \alpha_i \leq C$ \hspace{1cm} (17)

where $C$ is a controlling parameter for the optimisation stability. The dual form facilitates the nonlinear separable data patterns to depend only on the size of the training set, not on the dimension of the high dimensional feature space. A well-known procedure called quadratic programming (QP) [16] may be used to minimise $-W_d(\alpha)$. Nonzero values of the optimising solution correspond to support vectors that are used to construct the classifier in Equation (13).

### 4.3. Least squares support vector classifier-based equaliser

To reduce the computational complexity of the QP process in standard SVCs, we apply the LS-SVC technique [13] for equalisation, by modifying the inequality constraints in Equation (16) to equality constraints. The classification problem in LS-SVC, therefore, is formulated, in a primal form, as

$$
\begin{align*}
\text{Minimise } & W_{LS}(w, e) = \frac{1}{2}w^Tw + \frac{1}{2}y^T \sum_{j=1}^{P} e_j^2 \\
\text{Subject to } & y_j[w^T\phi(x_j) + \beta] = 1 - e_j, \ j = 1, 2, \ldots, P
\end{align*}
$$

(18)

where $w$ is the hyperplane coefficient vector, $e_j$ is the misclassification error due to the equality constraint and $\gamma$ is a regularisation parameter that is predefined to control the error tolerance weight.

By introducing Lagrange multipliers, we construct a Lagrange function (the dual form of the cost function) from Equation (18) as

$$
W_{LS}(w, \beta, e, \alpha) = W_{LS}(w, e) = \sum_{j=1}^{P} \alpha_j \{ y_j[w^T\psi(x_j) + \beta] - 1 + e_j \}
$$

(19)

where $\alpha_j \in \mathbb{R}$ and the conditions for optimality become

$$
\begin{align*}
\frac{\partial L}{\partial w} &= 0 \Rightarrow w = \sum_{j=1}^{P} \alpha_j y_j \psi(x_j) \hspace{1cm} (20) \\
\frac{\partial L}{\partial \beta} &= 0 \Rightarrow \sum_{j=1}^{P} \alpha_j y_j = 0 \hspace{1cm} (21) \\
\frac{\partial L}{\partial e} &= 0 \Rightarrow y_j[w^T\psi(x_j) + \beta] - 1 + e_j = 0, \ j = 1, 2, \ldots, P \hspace{1cm} (22) \\
\frac{\partial L}{\partial \alpha} &= 0 \Rightarrow \alpha_j = \gamma e_j, \ j = 1, 2, \ldots, P \hspace{1cm} (23)
\end{align*}
$$

It can be derived that Equations (20)–(23) can be expressed in a matrix form as

$$
\begin{bmatrix}
0 & -y^T \\
y & \Omega + \gamma^{-1} I
\end{bmatrix}
\begin{bmatrix}
\beta \\
\alpha
\end{bmatrix} =
\begin{bmatrix}
0 \\
1
\end{bmatrix}
$$

(24)

where $y = [y_1 \ldots y_P]^T$ and the element in the $i$th row, $j$th column of $\Omega$, is defined as

$$
\Omega_{ij} = y_i y_j \phi(x_i) \phi(x_j) = y_i y_j K(x_i, x_j)
$$

(25)

The linear equation in (24) can be easily solved by many existing algorithms rather than the QP technique used in SVCs in Subsection 4.2. The concept of support vector disappears in the LS-SVC case because the solution contains a spectrum of values rather than few nonzero values as in SVCs. This, however, introduces an increase in detection complexity. Once classifier coefficients ($\alpha$, $\beta$) have been evaluated, the classification process for the information data can be accomplished by using Equations (12) and (13) but with the whole range of training symbols.

### 4.4. Sparse least squares support vector classifier-based equaliser

To alleviate the detection complexity resulted from the full spectrum of LS-SVC support values, some sort of sparseness can be imposed by using the pruning approach [18], whose procedure is as follows: train the classifier by the
training data set; sort the spectrum of resulting classifiers’ coefficients and remove the least important coefficients according to some acceptable degree in performance. This process can be terminated at this stage or extended to re-train the classifier by inputting the remaining corresponding data set.

As concluded from Subsection 4.3, a drawback of the LS-SVC in comparison with the original SVC formulation is that sparseness is lost in the LS-SVC case. This is because the support values are proportional to the errors at the data points, as can be seen in condition (22). However, by plotting the spectrum of the sorted support values, one can evaluate which data are the most significant for contribution to the LS-SVC classifier. Sparseness is imposed then by gradually omitting the least important data from the training set and re-estimating the LS-SVC. This algorithm can be summarised as

1. Train each of the $M$ LS-SVCs on $P$ pilot symbols.
2. Remove a predefined number (of sparseness ratio $\rho$) of pilot symbols that correspond to the smallest support values in the spectrum.
3. Re-train the LS-SVCs based on the reduced training set.

This procedure can also be implemented iteratively by removing a small part of training data until some preset performance index is reached. This represents pruning of the LS-SVC. As a result, the number of detection coefficients becomes less, which reduces the detection complexity. It is worth noting that the pruning does not involve a computation of a Hessian matrix. Instead, it is immediately performed based upon the physical meaning of the solution vector $\alpha$.

5. PERFORMANCE ANALYSIS

In order to appreciate the performance advantages of the SVM-based equalisation, we explain in this section the conceptual differences between conventional equalisation and equalisation as a pattern recognition solution. Considering the signal constellation for a digital communication signalling with ISI effects, the equalisation problem is to find the optimal mapping that returns signals shift due to ISI, into their original positions on the constellation. While in pattern recognition, the problem is to find the optimal decision boundary (a hyperplane in an appropriate higher dimensional space) that is used for symbol detection.

There are two main implications of the pattern recognition-based equalisation. First, for an increasing number of observed symbols, a stable decision boundary can be obtained with a faster convergence speed, achieving a better detection accuracy. Second, the time-varying nature of channels has a small impact on the decision boundary, because the time variation can be viewed as a small noise added to the constellation in the detection regions of the learning machine.

A special case that clearly shows the superior performance of the SVM-based equaliser is the LOS scenario, where most of the channel energy are concentrated on first components, which results in two (for binary signalling) separable intensive clouds of observations in a high dimensional feature space with, most importantly, a proper kernel mapping. The detection then performs on average as a simple AWGN detection, if a well-chosen kernel function is used for the mapping. An illustrative example is shown in Figure 2, where two 4-tap FIR channel models with Rayleigh fading for each tap are used to visualise the signal constellation in the feature space for a binary signalling assuming no added noise. For a non-LOS (NLOS) scenario (Channel 1), Figure 2(a), the average channel taps’ powers are equal so that the signal centres are spread in the feature region. Hence, more classification errors are potential when noise is added at the receiver. On the other hand, Figure 2(b) (Channel 2) shows the signal pattern for a LOS scenario where most of the channel power is concentrated on the first tap. With the assumption of unit average channel energy, the classification performance resembles the detection performance of the simple case of AWGN only, that is, the distance $d$ in Figure 2(b) reflects the signal amplitude.

Figure 2. An example of line-of-sight (LOS) versus non-LOS signal constellation viewed in pattern space.
6. COMPLEXITY ANALYSIS

In this section, a complexity analysis is provided for the proposed SVM-based equalisation, in terms of the number of multiplication operations in both the training and detection modes for a single training session, that is, one transmission packet. The analysis given in this section is normalised to one classifier, so that only a corresponding symbol of each block is considered.

Table I summarises the complexity of the proposed SVC, LS-SVC and sparse SVC methods. The complexity of rake-MRC is also included as a benchmark. In the rake-MRC receiver, the training mode is used for channel estimation. The computational complexity of one training session is of $O(N_c P L_{est})$ multiplication operations, whereas the detection complexity is of $O(N_c L_f B)$ operations for $B$ detected symbols per packet.

For the standard SVM method, solving the QP problem for optimisation in an SVC training for $P$ pilot symbols requires $O(P^3)$ multiplication operations plus $N_c P^2$ multiplications in generating Hessian matrix $K$ for nonlinear mapping [34], where $N_c$ represents the dimension of received symbol that is the spreading code length in this system. Thus, the total number of multiplication operations in training the SVC can be approximated by $(N_c P^2 + P^3)$.

The most important reason for using LS-SVC is to reduce the training complexity over the standard SVC. In order to evaluate the multiplication operations in LS-SVC training, the linear equations system in Equation (24) requires $\left(\frac{1}{2}\right) (P + 1)^3$ operations for Gaussian elimination plus $\left(\frac{1}{2}\right) (P + 1)^2$ for back substitution process. In addition, one should consider another $P$ multiplications for adjusting $\Omega$ by $\gamma^{-1}I$. Moreover, the same number of $K$-generating operations is considered here. Therefore, the total number of operations in LS-SVC training would be on the order of $\left[N_c P^2 + P + \left(\frac{1}{2}\right) P^2 + \left(\frac{1}{4}\right) P^3 \right]$. For detection complexity, the numbers of multiplications for one detection session ($B$ data symbols) can be calculated for each tested symbol as follows: $O(N_{SV} N_c B)$ for SVC and $O(P N_c B)$ for LS-SVC, where $N_{SV}$ is the average number of support vectors of the SVC method. It is clearly noticed that the number of support vectors reduces the detection computational complexity in comparison with that of LS-SVC. However, practical experiments show that a large number of support vectors are obtained in DS-UWB systems.

Table I. Complexity comparison.

<table>
<thead>
<tr>
<th>Training complexity</th>
<th>Detection complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rake with MRC</td>
<td>$O(N_c P L_{est})$</td>
</tr>
<tr>
<td>SVC</td>
<td>$O(P^3)$</td>
</tr>
<tr>
<td>LS-SVC</td>
<td>$O(\frac{1}{2}P^3)$</td>
</tr>
<tr>
<td>Sparse LS-SVC</td>
<td>$O((1 - \rho)P N_c B)$</td>
</tr>
</tbody>
</table>

$N_c$, code length; $P$, pilot size; $B$, number of data blocks; $L_f$, rake fingers; $N_{SV}$, average number of support vectors; $\rho$, sparseness ratio and $L_{est}$, length of the estimated channel.

Imposing sparseness will reduce the detection complexity of LS-SVC according to a predefined cutting ratio $\rho$—the ratio between removed points to the total training points. Hence, the detection complexity of sparse LS-SVC is $O((1 - \rho)P N_c B)$.

Figure 3 illustrates the training complexity comparison for both SVC and LS-SVC with respect to the number of pilot symbols.

For comparison purposes, define $\delta = \frac{N_{SV}}{P}$ ($0 \leq \delta \leq 1$). The overall complexity saving of (sparse) LS-SVC over SVC can be evaluated by a ratio $\eta$, which is defined as

$$\eta = \frac{(\text{Total number of multiplications})_{\text{LS-SVC}}}{(\text{Total number of multiplications})_{\text{SVC}}}$$

Figure 4 shows the reduction in the overall computational complexity for different numbers of pilot symbols considering two average numbers of support vectors and three cutting ratios of sparseness of $\rho = 0$ (no sparseness), 30% and 60%. The number of data symbols is assumed to be six times of the number of pilot symbols, that is, $B = 6P$.

7. SIMULATION RESULTS

We use simulation results to demonstrate the performance of the proposed SVM-based equalisers. The simulations were executed using two channel models proposed in IEEE802.15.3a [19]: CM1 for the case of LOS and CM3 with NLOS based on 4–10-m measurements. We set the GI length to $L_{GI} = 15$ for CM1 and $L_{GI} = 40$ for CM3, respectively, to match the maximum delay spread of these channels. The channels are assumed to be constant over one transmission session (one packet). We assume BPSK modulation with data rate varying from 138 to 163 Mbps.

![Figure 3. Training complexity comparison. SVC, support vector classifier; LS-SVC, least squares SVC.](image-url)
Support vector machine-based equalisation for DS-UWB systems
M. Musbah and X. Zhu

Figure 4. Complexity saving of least squares support vector classifier over support vector classifier, where \( \delta \) is the average support vectors fraction and \( \rho \) is the removed pilot symbols ratio by imposing sparseness (sparseness ratio).

As specified in the proposal in [35], a ternary code of length \( N_c = 32 \) is used for spreading, with a chip width of \( T_c = 0.167 \) ns. Ternary codes have proven a better compromise between auto-correlation and cross-correlation properties. The number of symbols per block is \( M = 200 \), and the sizes of pilot blocks per packet were chosen to be \( P = 100, 200 \) and \( 500 \). The number of data blocks per packet is \( B = 2500 \). The SNR is defined as the ratio between the average received signal power and the noise power.

Figure 5 depicts the BER performance of the SVM-based equalisers in comparison with the rake-MRC receiver for CM1, where a LOS scenario is considered. The number of rake fingers was chosen to be the same as the number of pilot symbols of the classifiers, that is, \( L_f = P = 100 \), in order to fix the detection complexity for all equalisers. The SVM-based equalisers significantly outperform the rake receiver. For instance, at \( \text{SNR} = 10 \) dB, the average BER of the SVM-based equalisers is around \( 2 \times 10^{-5} \), whereas it is \( 7 \times 10^{-5} \) for rake receiver with perfect channel state information (CSI). As shown in Figure 5, the performance of the SVM-based equaliser is nearly the same as the performance of an AWGN detector. This is because the data patterns (in high dimensional space) of the received chips are concentrated around far apart centres that represent dominant LOS components, as described in Section 5.

The BER performance with CM3 is illustrated in Figure 6 where the number of rake fingers and the number of pilot symbols of the classifiers are set to be \( L_f = P = 200 \). Similar to Figure 5, the proposed SVM-based equalisers outperform the rake receiver, at the same detection complexity. It is also inferred that, for medium-to-high SNR range, the cross-correlator channel estimator in the rake receiver results in an irreducible error floor at \( \text{BER} \approx 10^{-3} \), while the SVM-based equalisers with implicit channel estimation even outperform the rake receiver with perfect CSI. Moreover, by introducing the LS-SVC based equalisers, the system performance is almost retained with the benefit of saving up to 50% of the training complexity, as shown in Figure 3.

The effect of imposing sparseness was examined for two levels of cutting ratios (\( \rho = 30\% \), 60%) representing the detection complexity that could be saved. The effectiveness of the sparse LS-SVC based equaliser is illustrated in Figure 7 for three pilot sizes \( P = 100, 200 \) and 500. The results show that for a large enough size of pilot symbols
Sparse LS–SVC Performance

Average BER

SNR (dB)

Figure 7. Sparse least squares support vector classifier performance for different numbers of pilot symbols. BER, bit error rate; LS-SVC, least squares support vector classifier; SNR, signal-to-noise ratio.

Sparse LS–SVC provides nearly the same performance as SVC, even with a reduction of 60% in detection complexity. The spectrum of the resulting average support values is depicted in Figure 8.

Choosing the appropriate number of symbols for training is a matter of a trade-off between BER performance and bandwidth efficiency. Figure 9 illustrates the learning curve of the LS-SVC based equaliser with CM3, which can guide to choose the appropriate pilot size. For our simulations, we have chosen up to \( P = 500 \) pilot symbols so that no significant improvement is achieved beyond that size, compared to the entailed increase in complexity.

Because of the sensitivity of kernel parameters, an optimal width parameter (\( \sigma \)) value in the Gaussian RBF kernel is required. In the above simulations, empirical tests have been conducted to obtain the optimum value of \( \sigma \).

Figure 10 shows the effect of the value of \( \sigma \) on the performance of LS-SVC based equaliser with CM3, for different SNRs levels. It is interestingly noticed that the optimal value of \( \sigma \) is around \( 10^{0.5} \) for different SNR levels.

8. CONCLUSION

The SVM technique has been applied for DS-UWB channel equalisation. Results show superior performance of SVCs compared to the conventional rake receiver. In particular, the SVM-based equalisation in the LOS scenario provides a close performance to the AWGN case. The LS-SVC based equalisers have been employed to reduce the training complexity of the standard SVC-based equaliser.
significantly, at a cost of small increase in detection complexity. The detection complexity, however, can be reduced by imposing some sparseness to the LS-SVC taps. Simulation results show that with a relatively large number of pilot symbols, the proposed sparse LS-SVC based equaliser can save up to 60% of the detection complexity, while maintaining the superior BER performance of SVCs.

REFERENCES


**AUTHORS’ BIOGRAPHIES**

**Mohamed Musbah** received the BSc degree from Garyonis University, Libya, in 1995, the MSc degree from Sheffield Hallam University, UK, in 2005, and the PhD degree from the University of Liverpool, UK, in 2010, respectively. He is currently a lecturer at Omar Al-Mukhtar University, Libya. His research interests include ultra wideband (UWB) communications, channel equalisation and machine learning, etc.

**Xu Zhu** received the BEng degree (with first class honors) from the Department of Electronics and Information Engineering, the Huazhong University of Science and Technology, Wuhan, China, in 1999, and the PhD degree from the Department of Electrical and Electronic Engineering, the Hong Kong University of Science and Technology, Hong Kong, in 2003. Since May 2003, she has been with the Department of Electrical Engineering and Electronics, the University of Liverpool, Liverpool, U.K., where she is currently a senior lecturer. Dr. Zhu was the vice chair of the 2006 and 2008 ICA Research Network International Workshops. She has acted as session chair and technical program committee member for more than 20 international conferences, such as IEEE ICC 2010. Her research interests include MIMO systems, equalisation, machine learning, cooperative communications, cognitive networks, and cross-layer design, etc.