

Least Squares Support Vector Machines for Fixed-Step and Fixed-Set CDMA Power Control

Judd A. Rohwer
Sandia National Laboratories
Department 2665
P.O. Box 5800 MS-0986
Albuquerque, NM 87185-0986
Email: jarohwe@sandia.gov

Chaouki T. Abdallah and Christos G. Christodoulou
Department of Electrical and Computer Engineering
MSC01 1100
1 University of New Mexico
Albuquerque, NM 87131-0001
Email: chaouki@ecece.unm.edu

Abstract— This paper presents two machine learning based algorithms for CDMA power control. The Least Squares Support Vector Machine (LS-SVM) algorithms classify eigenvalues estimates into sets of power control commands. A binary LS-SVM algorithm generates fixed step power control (FSPC) commands, while the one vs. one multiclass LS-SVM algorithm generates estimates for fixed set power control.

I. INTRODUCTION

Power control is a critical component in CDMA cellular systems. Power control combats the “near-far” effect by adjusting the transmit power of each mobile. This technique reduces the multiple access interference and if the system capacity is within the set limits the, desired signal-to-interference ratios (SIRs) are achieved at all base stations. Research in the power control field covers centralized, distributed, and stochastic power control [1],[2],[3]. In most of the published research however, it is assumed that the uplink channel gain is known, and no limitations are placed on the word-length and update rate of the transmitted power control command.

The IS-95 and cdma2000/3G systems have an 800 bps up/down power control command rate; the single bit power control command is thus sent to the mobile station every 1.25 milliseconds [4]. This limits the options with regards to power control systems, but the design reduces the problem to that of generating a fixed-step power control command. Standard power control systems implemented in cellular systems use signal-to-interference ratio (SIR) estimates, bit error rates (BER), or frame error rates (FER). Many of the published power control algorithms assume that these estimates are available and accurate.

The algorithms presented in this paper require only the set of eigenvalues from the sample covariance matrix of the received signal; the signal subspace dimension and power estimates are not required. The machine learning algorithm classifies the set of eigenvalues into a SIR set which then determines the power control command. Both binary and multiclass machine learning algorithms are developed to

solve this power control classification problem. The two types of algorithms require unique training sequences, but the optimum performance of both algorithms is achieved with the same kernel and test parameters.

II. MACHINE LEARNING BASICS

A major machine learning application, pattern classification, observes input data and applies classification rules to generate a binary or multiclass labels. In the binary case, a classification function is estimated using input/output training pairs, (\mathbf{x}_i, y_i) $i = 1 \dots n$, with unknown probability distribution, $P(\mathbf{x}, y)$,

$$(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n) \in \mathbb{R}^N \times Y, \quad (1)$$

$$y_i = \{-1, +1\}. \quad (2)$$

The estimated classification function maps the input to a binary output, $f : \mathbb{R}^N \rightarrow \{-1, +1\}$. The system is first trained with the given input/output data pairs then the test data, taken from the same probability distribution $P(\mathbf{x}, y)$, is applied to the classification function. For the multiclass case $Y \in \mathbb{R}^C$ where Y is a finite set of real numbers and C is the size of the multiclass label set. In multiclass classification the objective is to estimate the function which maps the input data to a finite set of output labels $f : \mathbb{R}^N \rightarrow \mathbb{G}(\mathbb{R}^N) \in \mathbb{R}^C$

Support Vector Machines (SVMs) were originally designed for the binary classification problem. Much like all machine learning algorithms SVMs find a classification function that separates data classes, with the largest margin, using a hyperplane. The data points near the optimal hyperplane are the “support vectors”. SVMs are a nonparametric machine learning algorithm with the capability of controlling the capacity through the support vectors.

The SVM maps the input space into a higher dimensional feature, F , space via a nonlinear mapping, kernel operation

$$\Gamma : \mathbb{R}^N \rightarrow F \quad (3)$$

$$\mathbf{x} \mapsto \Gamma(\mathbf{x}). \quad (4)$$

The data does not have the same dimensionality as the feature space since the mapping process is to a non-unique generalized surface [5]. The dimension of the feature space is not as

¹Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

important as the complexity of the classification functions. For example, in the input space, separating the input/output pairs may require a nonlinear separating function, but in a higher dimension feature space the input/output pairs may be separated with a linear hyperplane.

A. Binary Classification

In binary classification systems the machine learning algorithm generate the output labels with a hyperplane separation where $y_i \in \{-1, 1\}$ represents the classification "label" of the input vector \mathbf{x} . The input sequence and a set of training labels are represented as $\{\mathbf{x}_i, y_i\}_{i=1}^n$, $y_i = \{-1, +1\}$. If the two classes are linearly separable in the input space then the hyperplane is defined as $\mathbf{w}^T \mathbf{x} + b = 0$, \mathbf{w} is a weight vector perpendicular to the separating hyperplane, b is a bias that shifts the hyperplane parallel to itself. If the input space is projected into a higher dimensional feature space then the hyperplane becomes $\mathbf{w}^T \Gamma(\mathbf{x}) + b = 0$.

The SVM algorithm is based on the hyperplane definition [6],

$$\mathbf{w}^T \Gamma(\mathbf{x}_i) + b \geq 1, \text{ if } y_i = +1, \quad (5)$$

$$\mathbf{w}^T \Gamma(\mathbf{x}_i) + b \leq -1, \text{ if } y_i = -1, \quad (6)$$

This formulation is restated as

$$y_i [\mathbf{w}^T \Gamma(\mathbf{x}_i) + b] \geq 1, i = 1, \dots, n. \quad (7)$$

Given the training sets in (1) the binary support vector machine classifier is defined as

$$y(\mathbf{x}) = \text{sign} \left[\sum_{i=1}^n \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + b \right]. \quad (8)$$

The nonlinear mapping function $\Gamma(\mathbf{x}_i)$ is related to $k(\mathbf{x}, \mathbf{x}_i)$ by $\Gamma(\mathbf{x})^T \Gamma(\mathbf{x}_i) = k(\mathbf{x}, \mathbf{x}_i)$. Kernel functions are used to compute the scalar dot products of the input/output pairs in the feature space F . Four popular kernel functions are the linear kernel, polynomial kernel, radial basis function (RBF), and multilayer perceptrons (MLP).

$$\text{linear, } k(\mathbf{x}, \mathbf{x}_i) = \mathbf{x} \cdot \mathbf{x}_i \quad (9)$$

$$\text{polynomial, } k(\mathbf{x}, \mathbf{x}_i) = ((\mathbf{x} \cdot \mathbf{x}_i) + \theta)^d \quad (10)$$

$$\text{RBF, } k(\mathbf{x}, \mathbf{x}_i) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}_i\|^2}{\sigma^2}\right) \quad (11)$$

$$\text{MLP, } k(\mathbf{x}, \mathbf{x}_i) = \tanh(\kappa(\mathbf{x} \cdot \mathbf{x}_i) + \theta) \quad (12)$$

The non-zero α_i 's are "support values" and the corresponding data points, \mathbf{x}_i , are the "support vectors". Quadratic programming is one method of solving for the α_i 's and b in the standard SVM algorithm.

B. Multiclass Classification

For the multiclass problem the machine learning algorithm produces estimates with multiple hyperplane separations. The set of input vectors and training labels is defined as $\{\mathbf{x}_i, y_i^c\}_{i=1, c=1}^{i=n, c=C}$, $\mathbf{x}_i \in \mathbb{R}^N$, $y_i \in \{1, \dots, C\}$, i is the index of the training pattern and C is the number of classes. There

exist many SVM approaches to multiclass classification problem. Two primary multiclass techniques are one-vs-one and one-vs-rest. One-vs-one applies SVMs to selected pairs of classes. For P distinct classes there are $\frac{P(P-1)}{2}$ hyperplanes that separate the classes. The one-vs-rest SVM technique generates P hyperplanes that separate each distinct class from the ensemble of the rest. In this paper we only consider the one-vs-one multiclass SVM.

Platt, et.al., [7] introduced the decision directed acyclic graph (DDAG) and a Vapnik-Chervonenkis (VC) analysis of the margins. The DDAG technique is based on $\frac{P(P-1)}{2}$ nodes for a P class problem, one node for each pair of classes. The test error of the DDAG depends on the number of classes, P , and the margins between the data points and the hyperplanes, not on the dimension of the input space. In [7] it is proved that maximizing the margins at each node of the DDAG will minimize the generalization error. The performance benefit of the DDAG architecture is realized when the i^{th} classifier is selected at the $i^{\text{th}}/j^{\text{th}}$ node and the j^{th} class is eliminated. Refer to Figure 1 for a diagram of a four class DDAG.

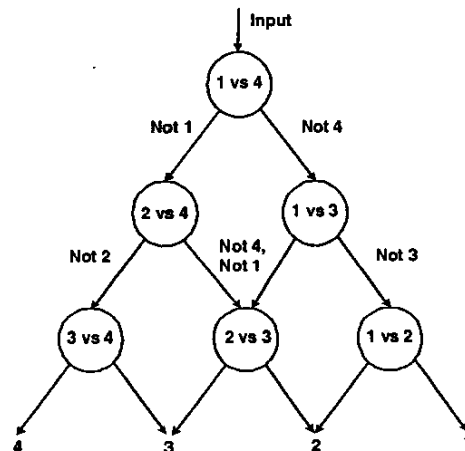


Fig. 1. DAGSVM for Four Classes

C. Least Squares SVM

Suykens, et.al., [8] introduced the LS-SVM which is based on the SVM classifier, refer to equation (8). The LS-SVM classifier is generated from the optimization problem:

$$\min_{\mathbf{w}, b, \phi} \mathcal{L}_{LS}(\mathbf{w}, \phi) = \frac{1}{2} \|\mathbf{w}\|^2 + \frac{1}{2} \gamma \sum_{i=1}^n \phi_i^2, \quad (13)$$

γ and ϕ_i are the regularization and error variables, respectively. The minimization in (13) includes the constraints

$$y_i [\mathbf{w}^T \Gamma(\mathbf{x}_i) + b] \geq 1 - \phi_i, \quad i = 1, \dots, n, \quad (14)$$

The LS-SVM includes one universal parameter, γ , that regulates the complexity of the machine learning model. This parameter is applied to the data in the feature space, the output of the kernel function. A small value of γ minimizes

the model complexity, while a large value of γ promotes exact fitting to the training points. The error variable ϕ_i allows misclassifications for overlapping distributions [9].

The Lagrangian of equation (13) is defined as

$$\mathcal{Z}_{LS}(\mathbf{w}, b, \phi, \alpha) = \mathcal{L}_{LS}(\mathbf{w}, b, \phi) - \sum_{i=1}^n \alpha_i \{y_i [\mathbf{w}^T \Gamma(\mathbf{x}_i) + b] - 1 + \phi_i\} \quad (15)$$

where α_i are Lagrangian multipliers that can either be positive or negative. The conditions of optimality are

$$\frac{d\mathcal{Z}_{LS}}{d\mathbf{w}} = 0, \quad \mathbf{w} = \sum_{i=1}^n \alpha_i y_i \Gamma(\mathbf{x}_i) \quad (16)$$

$$\frac{d\mathcal{Z}_{LS}}{db} = 0, \quad \sum_{i=1}^n \alpha_i y_i = 0 \quad (17)$$

$$\frac{d\mathcal{Z}_{LS}}{d\phi} = 0, \quad \alpha_i = \gamma \phi_i \quad (18)$$

$$\frac{d\mathcal{Z}_{LS}}{d\alpha_i} = 0, \quad y_i [\mathbf{w}^T \Gamma(\mathbf{x}_i) + b] - 1 + \phi_i = 0 \quad (19)$$

A linear system can be constructed from equations (16)–(19) [8],

$$\begin{bmatrix} I & 0 & 0 & -Z^T \\ 0 & 0 & 0 & -Y^T \\ 0 & 0 & \gamma I & -I \\ Z & Y & I & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ b \\ \phi \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vec{1} \end{bmatrix} \quad (20)$$

$$Z = [\Gamma(\mathbf{x}_1)^T y_1, \dots, \Gamma(\mathbf{x}_n)^T y_n] \quad (21)$$

$$Y = [y_1, \dots, y_n], \quad \vec{1} = [1, \dots, 1] \quad (22)$$

$$\phi = [\phi_1, \dots, \phi_n], \quad \alpha = [\alpha_1, \dots, \alpha_n] \quad (23)$$

By eliminating weight vector \mathbf{w} and the error variable ϕ , the linear system is reduced to:

$$\begin{bmatrix} 0 & Y^T \\ Y & ZZ^T + \gamma^{-1}I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \vec{1} \end{bmatrix} \quad (24)$$

In the linear systems defined in (20)–(24) the support values α_i are proportional to the errors at the data points. In the standard SVM case many of these support values are zero, but most of the least squares support values are non-zero. In [8] a conjugate gradient method is proposed for finding b and α , which are required for the SVM classifier in equation (8).

III. LS-SVM FOR CDMA POWER CONTROL

In this paper we design LS-SVM techniques to generate power control commands based on eigenvalues estimates, not on assumed knowledge of the link gain, SIR, BER, or FER. The LS-SVM algorithm classifies the set of eigenvalues from the sample covariance matrix of the received signal into a SIR set. The LS-SVM output label determines the power control command. An intelligent design of the training sequences allows for the machine learning system to accurately

TABLE I
SET OF EIGENVALUES FOR MACHINE LEARNING BASED POWER CONTROL

	Set of Eigenvalues							
	1	2	3	4	5	6	7	8
1D SS	208.2	83.4	60.7	71.0	58.6	49.4	50.1	43.2
2D SS	196.9	119.1	66.1	59.4	59.4	43.1	49.3	46.3
3D SS	248.2	191.1	93.7	59.1	52.4	52.0	46.0	47.3

classify the power estimates into fixed-step and fixed-set power control commands.

The eigenvalues and eigenvectors of a signal subspace are computed with principle component analysis (PCA). The eigenvalues of the signal and noise subspaces directly relate to the signal power and interference power. In [10] subspace estimation and tracking algorithms are presented. The projection approximation subspace tracking algorithm (PASTd) algorithm calculates the signal subspace as a solution of an unconstrained minimization problem. By using projection approximations, the eigenvalues and eigenvectors of the signal subspace are computed. Table I includes three sets of estimated eigenvalues, one for each one-dimensional signal subspace, two-dimensional signal subspace, and three-dimensional signal subspace.

A. Signal Model

Simulations of the LS-SVM power control algorithm are based on a complex system model that includes amplitude and phase distributions representative of the CDMA communication channel. The received signal at the receiver is modeled as

$$\mathbf{x}_r(t) = \sum_{d,l=1}^{D,L} \bar{\mathbf{a}}(\theta_l) \alpha_{dl} s_d(t - \tau_{dl}) \cos(w_c(t - \tau_{dl})) + n_d(t) \quad (25)$$

This model includes D antenna array elements with steering array vector $\bar{\mathbf{a}}(\theta_l)$ and additive Gaussian noise $n_d(t)$. The model assumes L independent, resolvable signal paths. The multipath variable α_{dl} is defined as $\alpha_{dl} = \rho_{dl} e^{j(2\pi f_c(t - \tau_{dl}) + \phi_c)}$. The amplitude of the received signal ρ_{dl} , includes the transmitted power $\sqrt{p_t(t)}$ and the attenuation due to the link gain and shadowing q^l . This variable is modeled as a fixed, Rayleigh, Ricean, or log-normal distributed random variable. The Doppler shift for each resolvable path is defined by $f_c = \frac{v_a w_c}{2\pi c}$; v_a is the velocity of the mobile in $\frac{m}{sec}$, w_c is the carrier frequency, and c is the speed of propagation. A uniformly distributed carrier phase shift, ϕ_c , and a time delay for each signal path, τ_{dl} , are also included in the multipath variable, α_{dl} . The CDMA spreading code, $s_d(t - \tau_{dl})$, provides the processing gain at the correlator output.

B. Fixed-Step Power Control (FSPC)

This section presents a binary machine learning approach to generating the power control commands based on eigenvalue estimates. The training process of the binary system

must include data that spans the expected SIR range of the test data. In addition, the LS-SVM algorithm for power control is sensitive to kernel parameters. Simulation results prove that the best performance is achieved with both the linear kernel and a one degree polynomial kernel, $d = 1$, refer to equations (9), (10). The one degree polynomial kernel is effectively a linear kernel with a added bias, θ . Simulation results with the radial basis function (RBF) and multilayer perceptron (MLP) kernels include error rates 40% to 50% higher than error rates from simulations based on the linear and polynomial kernel.

Figure 2 presents simulation results for a binary LS-SVM power control system. The top window shows the received SIR data points and the 7dB SIR threshold. The bottom window shows the binary output of the LS-SVM system. A "0" represents a received SIR greater than 7dB and therefore corresponds to a power down command, a "1" represents a received SIR less than 7dB and therefore corresponds to a power up command.

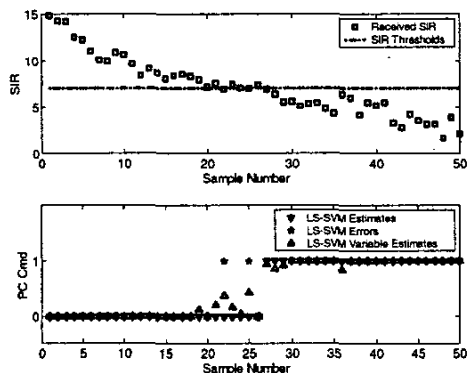


Fig. 2. Binary LS-SVM power control. The plot includes the received SIR data points, 7dB SIR threshold, ML estimates, and markers for the ML errors.

Table II presents the percentage of errors for five independent simulations. The simulations include 1500 data samples that include received SIRs from 15dB to 0dB. Included are error rates for one, two, and three-dimensional signal subspaces and the average error distance from the data points to the SIR decision boundary. The total error rate is the average of these subspace data points. The data shows that the binary LS-SVM algorithm correctly classifies over 90% of all eigenvalue sets.

C. Fixed-Set Power Control

A fixed set of power control commands is generated with a multiclass machine learning system. Each class represents a range of received SIRs, which is translated into a power control command. The multiclass system is based on the binary label system, $y_i \in \mathbb{k}$, where \mathbb{k} is a set of real numbers that represent an appropriate range of expected SIR values, $\mathbb{k} \in \{3, 5, 7, 9, 11\}$. The machine learning based fixed-set power control command is applied to the standard CDMA power

TABLE II
RATE OF ERRORS FOR FIXED STEP POWER CONTROL

SIR Thresholds	% of Errors				Avg Dist (dB)
	Total	1D SS	2D SS	3D SS	
11 dB	4.67	7.80	3.20	3.00	0.827
9 dB	9.27	15.80	5.40	6.60	0.870
7 dB	8.07	9.80	6.20	8.20	0.563
5 dB	7.67	7.80	8.20	7.00	0.449
3 dB	7.80	15.40	4.60	3.40	0.800

control system. The size of the command is directly related to the size of the one-vs-one multiclass DDAG structure.

- Preprocessing for SVM Training
 - 1) Generate the training signal vectors for the P SVM classes, each class represents a SIR threshold.
- LS-SVM Training
 - 1) With the P eigenvalue vectors train the $\frac{P(P-1)}{2}$ nodes with the one-vs-one LS-SVM algorithm.
 - 2) Store the LS-SVM variables, α_i and b from equation (8), which define the hyperplane separation for each DDAG node.
- LS-SVM Testing for the i^{th}/j^{th} DDAG Node
 - 1) Acquire the input signal from the antenna array, the dimension of the signal subspace and the SIR in not known.
 - 2) Calculate the eigenvalues of the sample covariance matrix.
 - 3) Test the input eigenvalue vector against the LS-SVM hyperplane for the i^{th}/j^{th} node.
 - 4) Calculate the mean value of the LS-SVM output vector (labels). Compare this value to the label definition at the node, then select the proper SIR label.
 - 5) Repeat process for the next DDAG node in the evaluation path or declare the final SIR label.

The LS-SVM system must be trained for all possible SIRs and the training process must include an equal number of training samples for each DDAG class. Due to the multi-dimensional input data the training vectors must account for all possible combinations of signal subspace dimensions and eigenvalues.

Refer to Figure 3 for a plot of the LS-SVM power control results with a three class DDAG. The plot includes the estimated received SIRs, the LS-SVM estimates, markers for the LS-SVM errors, and the SIR thresholds.

Selection of optimum training and test parameters is achieved by analyzing empirical results. Correctly classifying eigenvalue sets into received SIR sets is dependent on two critical variables; the lengths of the training and test vectors, and the sample size of the averaging windows applied in the preprocessing steps. Figure 4 displays the percentage of errors versus the length of training and test vectors. The simulations are based on a five class DDAG with SIR thresholds set at [3dB, 5dB, 7dB, 9dB, 11dB] and include

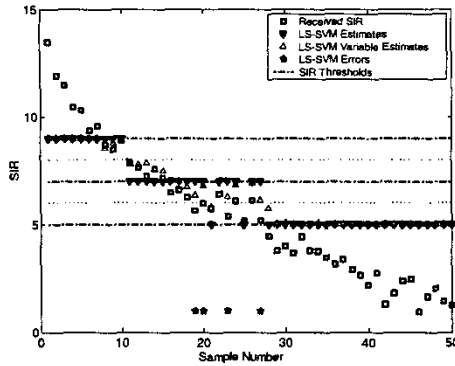


Fig. 3. LS-SVM power control with a three class DDAG. The plot includes the received SIR data points, ML estimates, SIR thresholds, and markers for the ML errors.

1500 independent test cycles. The results show that the lowest percentage of errors occurs with 700 training samples and 60 test samples; the length of the test vector influences the percentage of errors to a greater degree than the training vectors.

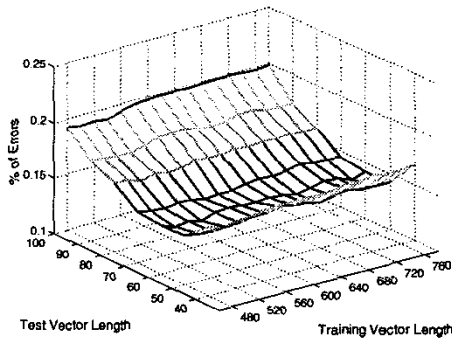


Fig. 4. Performance characterization of the LS-SVM algorithm for fixed set power control based on eigenvalue estimates, percentage of errors is a function of the lengths of the training and test vectors.

The window size for averaging a fixed set of training and test data affects the accuracy of classifying the input set of eigenvalues. The minimum error rate is achieved with a training window that averages six samples and a test window that averages five input data samples. In general, the error rates are relatively constant for training window sizes between two and ten samples, and the error rate increases as the size of the test windows increases beyond five samples.

Table III includes the percentage of errors for three, four, and five class DDAG systems. The simulations include 1500 data samples that include received SIRs from 15dB to 0dB. Included are error rates for one, two, and three-dimensional signal subspaces, the MSE between the received data and the LS-SVM estimates, and the average distance from the data points in error to the SIR decision boundary. The data shows that the three class LS-SVM DDAG algorithm correctly

TABLE III
RATE OF ERRORS FOR FIXED SET POWER CONTROL

DDAG Classes	% of Errors				MSE	Avg Dist (dB)
	Total	1D SS	2D SS	3D SS		
3	6.40	7.60	7.80	3.80	0.081	0.271
4	13.33	13.40	12.40	14.20	0.184	0.272
5	13.47	16.80	14.20	9.40	0.206	0.248

classifies over 93.6% of all eigenvalue sets and the four and five class systems correctly classify over 86.5%. The average distance of the data points in error to the decision boundary is less than 0.275dB. This statistic shows that the classification errors are located very close to the decision boundaries, proving the robust design of the algorithm.

1) *Mobile Capacity and Convergence*: Two standard methods for characterizing the performance capabilities of power control algorithms is the convergence rate and the mobile capacity [11]. A performance indicator for both methods is the probability of outage, the probability that the mobile's received SIR is below the desired threshold. The rate of convergence is defined as the number of power control iterations required for the system's probability of outage to converge to a steady state value. The capacity of a mobile cellular system is the number of mobile users that can be supported while achieving the required Quality of Service (QoS), usually defined in terms of received SIR, BER, or FER.

The simulation system includes randomly generated link gains for the number of mobiles simultaneously entering the system. Refer to [11] for a complete overview of the simulation environment. Through an iterative process the transmit powers are updated and the received SIRs converge to the desired level. Figure 5 plots the probability of outage versus the number of iterations for FSPC and three, four, and five class fixed-set power control. The simulation setup includes eighteen mobiles entering the cellular system. The FSPC system converges to zero probability of outage after eighteen power control iterations. The three, four, and five class fixed-set systems converge after eleven, eight, and seven iterations, respectively. Figure 6 includes the probability of outage versus the number of mobiles in the cellular system. Eight iterations are completed for each algorithm before the SIR and outage probability is calculated. For fixed-set power control the capacity decreases with the size of the power control set. The five class fixed-set system supports twenty mobiles with zero probability of outage. For the small number of iterations the capacity of the FSPC system is marginal, at best. The mobile capacity for each system may increase with a greater number of power control iterations, but as the number of iterations increases, so does the required computational time.

From the simulation results, the performance of the fixed-set power control systems far exceeds the capabilities of the FSPC system, in terms of convergence, mobile capacity, and performance versus computational effort. The IS-95 and

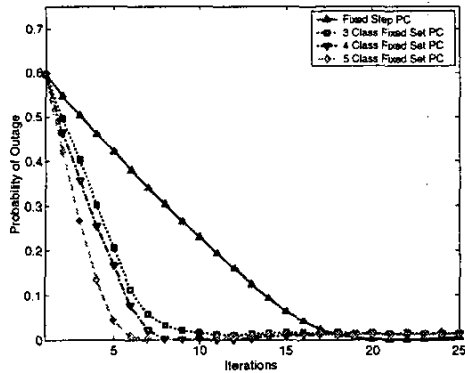


Fig. 5. Comparison of outage probabilities versus power control iterations for FSPC, and 3, 4, and 5 class fixed set power control.

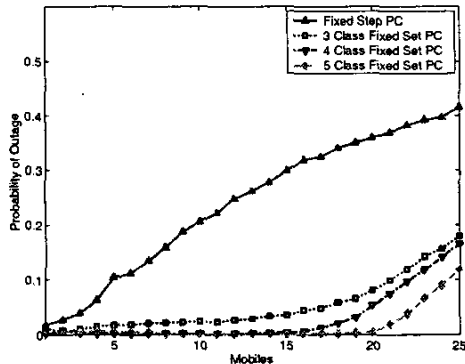


Fig. 6. Comparison of outage probabilities versus mobile capacity for FSPC, and 3, 4, and 5 class fixed set power control.

cdma2000/3G systems have an 800 bps up/down power control command rate. Based on these architectures the power control systems are limited to a single power control bit is sent to the mobile station every 1.25 milliseconds [4]. This constraint forces the cellular system to a FSPC approach. The fixed-set power control system requires additional power control bits. The three and four class fixed-set designs could be implemented with two power control bits and the five class system could be implemented with three power control bits. These bit requirements deviate from the 3G system specifications, but could be achieved with the use of auxiliary channels as defined in [12].

IV. CONCLUSION

In this paper we present two solutions to LS-SVM power control. Both binary and multiclass machine learning algorithms are developed to solve this power control classification problem. These techniques are based on eigenvalue estimates of the sample covariance matrix and do not require the dimension of the signal subspace or the estimates of the received signal and interference powers. The LS-SVM power control algorithms classify the set of eigenvalues into the

received SIR set which then determines the power control command. The two algorithms require unique training sequences, but the optimum performance of both algorithms is achieved with the same kernel and test parameters.

Performance characterization is based on simulations of the machine learning algorithms. The LS-SVM training requirements are presented along with the effects of the training and test vector lengths. The results show that LS-SVM algorithms for fixed-step and fixed-set power control achieve low percentages of errors and are insensitive to kernel parameters. The best performance is obtained with the linear kernel, which is ideal for real-time implementation because of the short training and testing times.

REFERENCES

- [1] Sennur Ulukus and Roy D. Yates, "Stochastic Power Control for Cellular Radio Systems", *IEEE Transactions On Communications*, vol. 46, no. 6, pp. 784-798, June 1998.
- [2] Lei Song, Narayan B. Mandayam, and Zoran Gajic, "Analysis of an Up/Down Power Control Algorithm for the CDMA Reverse Link Under Fading", *IEEE Journal On Selected Areas In Communications*, vol. 19, no. 2, pp.277-286, February 2001.
- [3] Mohammad Saquib, Roy D. Yates, and Anand Ganti, "Power Control for an Asynchronous Multirate Decorrelator", *IEEE Transactions On Communications*, vol. 48, no. 5, pp. 804-812, May 2000.
- [4] Joseph C. Liberti, Jr. and Theodore S. Rappaport, *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, Prentice Hall, Upper Saddle River, NJ, 1999.
- [5] Daniel J. Sebald and James A. Bucklew, "Support Vector Machine Techniques for Nonlinear Equalization", *IEEE Transactions On Signal Processing*, vol. 48, no. 11, pp. 3217-3226, November 2000.
- [6] Johan A.K. Suykens, "Support Vector Machines: A Nonlinear Modelling and Control Perspective", *European Journal of Control*, vol 7, pp. 311-327, 2001
- [7] John C. Platt, Nello Christianini, and John Shawe-Taylor, "Large Margin DAGs for Multiclass Classification", in *Advances in Neural Information Processing Systems*, vol. 12, pp. 547-553, Cambridge, MA, MIT Press, 2000.
- [8] J.A.K. Suykens, L. Lukas, P. Van Dooren, B. DeMoor, and J. Vandewalle, "Least Squares Support Vector Machine Classifiers: a Large Scale Algorithm", *ECCTD'99 European Conf. on Circuit Theory and Design*, pp. 839-842, August 1999.
- [9] Johan A.K. Suykens, Tony Van Gestel, Jos De Brabanter, Bart De Moor, and Joos Vandewalle, *Least Squares Support Vector Machines* World Scientific, New Jersey, 2002.
- [10] Bin Yang, "Projection Approximation Subspace Tracking", *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 95-107, January 1995.
- [11] Aly El-Osery and Chaouki Abdallah, "Distributed Power Control in CDMA Cellular Systems", *IEEE Antennas and Propagation Magazine*, Vol. 42, no. 4, pp. 152-159, August 2000.
- [12] Vijay K. Garg, *IS-95 CDMA and cdma2000*, Prentice Hall, Upper Saddle River, NJ, 2000.