Gaussian Process Regressors for Multiuser Detection in DS-CDMA systems

Juan José Murillo-Fuentes *Member, IEEE*, and Fernando Pérez-Cruz *Senior Member, IEEE.*

Abstract—In this paper we present Gaussian processes for Regression (GPR) as a novel detector for CDMA digital communications. Particularly, we propose GPR for constructing analytical nonlinear multiuser detectors in CDMA communication systems. GPR can easily compute the parameters that describe its nonlinearities by maximum likelihood. Thereby, no cross-validation is needed, as it is typically used in nonlinear estimation procedures. The GPR solution is analytical, given its parameters, and it does not need to solve an optimization problem for building the nonlinear estimator. These properties provide fast and accurate learning, two major issues in digital communications. The GPR with a linear decision function can be understood as a regularized MMSE detector, in which the regularization parameter is optimally set. We also show the GPR receiver to be a straightforward nonlinear extension of the linear minimum mean square error (MMSE) criterion, widely used in the design of these receivers. We argue the benefits of this new approach in short codes CDMA systems where little information on the users' codes, users' amplitudes or the channel is available. The paper includes some experiments to show that GPR outperforms linear (MMSE) and nonlinear (SVM) state-of-the-art solutions.

I. INTRODUCTION

In direct-sequence code-division multiple-access (DS-CDMA) systems [1], [2], [3] the different users transmit over the same frequency band and during the same time slot using different codes, known as spreading codes. Multiuser detectors aim to recover the information of one or more users. There are two main aspects a multi-user detection (MUD) receiver for DS-CDMA must face: near-far [1] and multipath [4] problems. The former is grounded in the nonorthogonality of the users' codes and the high signal-power deviation from user to user. The latter causes inter-symbolic interference (ISI). The optimal solution to MUD in CDMA communication systems is known to be nonlinear [5], [6]. This nonlinearity is stronger for short spreading codes and less pronounced for longer codes. Furthermore, the finite-response multipath-mitigation equalizer and the radiofrequency amplifier at the receiver enhance the nonlinear nature of the optimal multi-user detector. The complexity of the optimal nonlinear MUD receiver [5] grows exponentially with the number of users (joint Viterbi decoding [1]), which limits its practical applicability.

Linear filter receivers, based on zero-forcing (decorrelator) or minimum mean-squared error (MMSE) criteria [7], [8], [9], [1], are widely used to avoid the computational complexity of nonlinear solutions, although they are suboptimal and clearly underperform in some fairly standard scenarios. Regularization [10] has been extensively used to improve the detection capabilities of linear MUD, when the empirical correlation matrix is ill-conditioned because short sequences have been transmitted. Blind algorithms [11], [12], [13] make use of regularizations tools as ridge regression [14] and in [15] the authors review parametric regularized MMSE detectors for CDMA and propose several extensions. However, these papers do not detail how the regularizer shape and its parameters are selected and standard methods for setting them, as cross-validation [16], typically need long training sequences. In [12] the regularization is blindly estimated using a neural network, although the channel is memoryless.

Nonlinear detectors enhance the linear detectors by approximating the optimal solution at a lower complexity cost. There are two major approaches in machine learning that can be followed to obtain nonlinear detectors. On the one hand, if full information on the system is available, we may estimate the symbols transmitted by each user and then cancel the interference (IC) over the user of interest (UoI). These methods usually exhibit near-optimum performance at the expense of perfect information on the number of users, users' signatures, channel impulse response (CIR), ... Furthermore, most of them are based on equal transmitted power for each user, as they assume perfect power control. The detection process in the uplink in wireless cellular CDMA systems can be roughly matched to this scenario and, in statistical inference, these learning problems are commonly referred to as generative modeling. Generative modeling concentrates on describing the dependencies between all the different variables (sources of information), so as to understand the correlations between the variables, gain knowledge about causal interactions, and answer any query using the constructed model. Graphical models [17] and density estimation [14] belong in this category. And good examples of multiuser detectors using generative modeling as IC can be found in [18]-[22].

On the other hand, we may have little knowledge of the whole communication system. We may not know the number of users, nor the other users' signatures, nor the CIR; and the users are received with different signal strength. In this scenario, typically encountered in the downlink of wireless cellular CDMA systems, the objective is to detect the incoming symbols for the UoI. These estimates might not be reliable to apply IC techniques or we might not have enough knowledge to estimate and suppress the other users' symbols. In statistical

inference, we resort to discriminative modeling to provide the most accurate answer for a given supervised learning problem. Discriminative modeling focuses on solving a given task (e.g., regression, classification) without trying to model the underlying probability density of the data [23]. By not fully modeling the data density this approach is limited in its ability to provide interpretations and answer arbitrary queries about the data. However, the predictive performance of discriminative models is superior to that of generative models. The applicability and degree of success of either approach depends on the available information at the receiver. In the discriminative group we have tools such as Gaussian processes [24] and support vector machines [25]. For DS-CDMA, successful approaches have been proposed in the literature using nonlinear discriminative modeling [26]-[30]. In [29] a multilayered perceptron (MLP) was applied to enhance the performance of linear detectors. However, training was time consuming and its convergence rate was unpredictable. A MUD receiver was successfully designed using a Gaussian radial basis function network (RBFN) [26] that significantly improved training times compared to [29]. Neither of them considered multipath wireless channels. There have been several proposals to jointly address MUD and channel equalization: RBFN and Volterra series [30], [27]; and, more recently, Support Vector Machines (SVMs) [28]. SVMs were developed from well-founded learning theory results [23] and they present features such as: a convex functional and universality [31], which makes them a desirable tool for solving general nonlinear detection problems. All these nonlinear methods solve an optimization functional to build a nonlinear MUD receiver, in which either the architecture or the parameters have to be prespecified. They cannot be learnt for each MUD receiver individually, because standard cross-validation techniques [32], typically used for this purpose, are not feasible in digital communication systems, as described in Section III.

Our proposal is motivated by the need of improving synchronous DS-CDMA receivers for multipath channels when the only available information is the spreading code of the UoI and a short training sequence. We present a novel discriminative model based on Gaussian processes for regression (GPR) [33], [34], [24]. The GPR can be used to design linear and nonlinear detectors and we examine both structures in this paper. We have shown some preliminary results applying GPR for MUD in DS-CDMA systems in [35], [36]. In this paper, we provide a deeper understanding of GPR for CDMA and consider its properties in detail. We extend our investigations to include a GPR computational complexity analysis and its relation to linear and nonlinear regularized MMSE. We also face multipath channels.

The GPR solution for MUD receivers is advantageous in several ways, when compared to other nonlinear machine learning tools (e.g. MLPs, RBFNs and SVMs). First, the GPR solution is analytical, given its parameters. There is no need for solving a complex optimization problem at this stage. Second, Given a training dataset, a likelihood function for the GPR parameters can be stated and, hence, its parameters can be optimally set (in maximum likelihood sense). MLPs, RBFNs or SVMs need to specify beforehand its structure

or parameters, which can be suboptimal for each individual instantiation of our problem. These characteristics translate to shorter training sequences and improved convergence for GPR-based CMDA receivers.

Linear GPR detectors can be presented as a regularized linear MMSE and its regularization parameter can be learnt, as well, by maximum likelihood. Consequently, we avoid setting the regularization parameter by hand and rely on the training sequence to optimally set it. The linear GPR detector outperforms linear MMSE receivers for short training sequences, as shown in the experimental section, because its regularization parameter is optimum for every training sequence. We also address its computational complexity, to show it is identical to that of the linear MMSE detector, and hence it can be used in any low complexity decoder.

Finally, we can understand the nonlinear GPR as a linear GPR in which the data has been previously nonlinearly transformed to a high-dimensional space [24]. Thereby the nonlinear GPR is presented as a nonlinear regularized Bayesian MMSE and its natural extension for building nonlinear receivers. The complexity of training a full GPR solution is demanding and it can be a limiting factor for its use over fast-fading channels. Nevertheless, recent advances in GPR training can be applied for designing low complexity receivers.

The remainder of this paper is organized as follows. We introduce DS-CDMA systems and the linear MMSE-MUD receiver in Section II. We present the GPR analytical solution in Section III and describe its likelihood function for the parameters. In Section IV we focus on the design of a MUD receiver for DS-CDMA using GPR and its interpretation as a regularized nonlinear MMSE receiver. We report in Section V the performance of GPR-MUD compared to state-of-the-art linear and nonlinear receivers, using a DS-CDMA system in different scenarios. We conclude with some final remarks and proposed further work in Section VI.

II. SYSTEM MODEL AND MUD

A. The DS-CDMA system model

In this paper we focus on synchronous DS-CDMA [1], [37], and we assume all users transmit at the same symbol-rate using a BPSK modulation. Although, the obtained results can be readily generalized to other scenarios, as asynchronous DS-CDMA, different rates, or modulations. In the discrete chip-rate-sampled baseband synchronous DS-CDMA model proposed in [1], we transmit K symbols (one per user) every T_s sec., $\mathbf{b}_t = [b_t(1), b_t(2), ..., b_t(K)]^\top$. Each user's symbol, $b_t(j)$, is amplified by a different a_j, i.e., in the downlink of a mobile (cellular) communication system larger amplitudes are assigned to users further away, causing the near-far problem to users closest to the base station. Finally, each user is multiplied by its spreading code $\mathbf{s}_j$, which is a sequence of N pseudo-random binary values regarded as chips.

The N-chip signal at the receiver end yields:

$$\mathbf{r}_t = \mathbf{H} \begin{bmatrix} \mathbf{SA} & 0 & \dots & 0 \\ 0 & \mathbf{SA} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{SA} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{b}_t \\ \mathbf{b}_{t-1} \\ \vdots \\ \mathbf{b}_{t-M+1} \end{bmatrix} + \mathbf{n}_t$$

$$= \mathbf{P} \cdot \mathbf{v}_t + \mathbf{n}_t, \tag{1}$$

where $\mathbf{S}$ is an $N \times K$ matrix whose columns contains the spreading codes, $\mathbf{A}$ is a $K \times K$ diagonal matrix containing the user's amplitudes and $\mathbf{n}_t$ is an NM-dimensional column-vector with additive white Gaussian noise (AWGN) of variance $\sigma_{\mathbf{n}}^2 \mathbf{I}$.

We have pre-multiplied the received chips by $\mathbf{H}$ to incorporate the effect of a multipath channel in the CDMA system, [28]. Throughout the paper, we consider a time-invariant channel with inter-symbolic interference (ISI) characterized by its discrete channel impulsive response h(z) of length (or order) M_c in chip periods. The length of the channel response times the chip period is the maximum delay considered in our multipath model. The matrix $\mathbf{P}$ in (1) summarizes the effect of the channel, the spreading codes and the different amplitudes for each user, and $\mathbf{v}_t = [\mathbf{b}_t^\top, \mathbf{b}_{t-1}^\top \dots \mathbf{b}_{t-M+1}^\top]^\top$ is a KM_s-dimensional vector including the transmitted bits.

B. Multi-User Detection

At the receiver, we learn the system model to estimate the transmitted symbols. Usually, a training sequence is transmitted to solve this task. In wireless communications short training sequences are mandatory [38] to increase the number of information bits. The objective for the DS-CDMA MUD receiver is to recover the transmitted bit for a particular user, the UoI. We can use the received vector in (1) or the projection of this vector onto the spreading codes, if available. Linear MUDs are useful when the ISI is negligible and the codes are quasi-orthogonal. When the multipath effect and the near-far problem are strong, the optimal detector becomes highly nonlinear. The nonlinearity of the detector is significantly more disruptive for short spreading codes. In these scenarios nonlinear detectors are useful. Nonlinear MUD for DS-CDMA estimate the symbol of the UoI as $\hat{b}_*(j) = f(\mathbf{r}_*)$. If we knew all the 2^{KM_s} possible received noise free states, we could derive a MUD by studying a Bayes-optimal classifier [28]. This optimal one-shot detector is given by:

$$\hat{b}_*(j) = \text{sign} \left(\sum_{i=1}^{2^{KM_s}} \frac{\tilde{b}^{(i)}(j)}{\sqrt{2\pi}\sigma_{\mathbf{n}}} \exp\left(-\frac{\left\| \mathbf{r} - \mathbf{P}\mathbf{v}^{(i)} \right\|^2}{2\sigma_{\mathbf{n}}^2} \right) \right), \tag{2}$$

where $\tilde{b}^{(i)}(j)$ is the class label $\{\pm 1\}$ for the ith noise free state $\mathbf{P}\mathbf{v}^{(i)}$. This structure resembles that of the Gaussian RBFN used in [26], and it suggests gaussian kernels in SVM, as described in [28]. But its complexity is exponential in the number of users and the length of the channel response.

III. GAUSSIAN PROCESSES FOR REGRESSION

A. The GPR solution

Gaussian Processes for regression (GPRs) [33], [34], [24] is a Bayesian technique for nonlinear regression estimation, which can be extended for classification problems [39], [40]. It assumes a zero-mean GP prior over the space of possible functions and a Gaussian likelihood model. The posterior can be analytically computed, it is a Gaussian density function, and the predictions given by the model are also Gaussians.

Given a labelled training data set ($\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^L$, where the input $\mathbf{x}_i \in \mathbb{R}^D$ and the output $y_i \in \mathbb{R}$) and a new input location $\mathbf{x}_*$, we aim to predict the probability distribution for its output y_*, i.e., $p(y_*|\mathbf{x}_*, \mathcal{D})$. In GPs we assume a generalized linear regressor model for y with Gaussian noise: $p(y|\mathbf{x}, \mathbf{w}) = \mathcal{N}(y; \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}), \sigma_\nu^2)$, where $\boldsymbol{\phi}(\cdot)$ defines a nonlinear transformation of the input space, and a zero-mean Gaussian prior over $\mathbf{w}$, $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}; \mathbf{0}, \sigma_{\mathbf{w}}^2 \mathbf{I})$. The predicted output, $p(y_*|\mathbf{x}_*, \mathcal{D}) = \mathcal{N}(y_*; \mu_{y_*}, \sigma_{y_*})$, yields

$$\mu_{y_*} = \boldsymbol{\phi}^\top(\mathbf{x}_*)\mu_{\mathbf{w}} = \mathbf{k}^\top \mathbf{C}^{-1} \mathbf{y} \tag{3}$$

$$\sigma_{y_*}^2 = k(\mathbf{x}_*, \mathbf{x}_*) + \mathbf{k}^\top \mathbf{C}^{-1} \mathbf{k} \tag{4}$$

where $\mu_{\mathbf{w}}$ is the MAP estimate of $\mathbf{w}$, being

$$(\mathbf{C})_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) + \sigma_\nu^2 \delta_{ij} \tag{5}$$

the covariance matrix of the Gaussian process and $\mathbf{k} = [k(\mathbf{x}_*, \mathbf{x}_1), \dots, k(\mathbf{x}_*, \mathbf{x}_n)]^\top$. The function $k(\cdot, \cdot)$ is known as the kernel function of the nonlinear transformation $\boldsymbol{\phi}(\cdot)$, as it represents the inner product in the transformed feature space:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_{\mathbf{w}}^2 \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\phi}(\mathbf{x}_j). \tag{6}$$

The covariance matrix is also referred to as the kernel matrix. We use both terms as synonyms in the paper, as they are typically used in the machine learning literature [31], [41], [24]. The nontrivial steps needed to obtain (3) and (4) are detailed in [34]. It is important to point out that the term $\mathbf{C}^{-1}\mathbf{y}$ in (3) is computed only once using the set of training samples and then used to estimate the output for any test input sample.

B. Covariance Matrix

If either $\boldsymbol{\phi}(\cdot)$ or $k(\cdot, \cdot)$ are known, we can analytically predict the output of any incoming sample. But for most regression problems the best nonlinear transformation (or its kernel) is unknown. We need to describe a parametric kernel that can be adjusted for each regression problem. Kernel design is one of the most challenging open problems in machine learning [24], as we need to incorporate our prior knowledge into the kernel and, at the same time, we want the kernel to be flexible to explain previously unknown trends in the data. The use of a parametric kernel function leads us to the problem of finding the optimal setting of its hyperparameters[1]. For example, if we know the optimal solution to be linear, we could use the linear kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_{\mathbf{w}}^2 \mathbf{x}_i^\top \mathbf{x}_j$, in which $\boldsymbol{\phi}(\mathbf{x}) = \mathbf{x}$. In this case, the only unknown hyperparameters would be σ_ν^2 and $\sigma_{\mathbf{w}}^2$ in (5)-(6), as we do not need to know

---

[1]We refer to the kernel parameters as hyperparameters to distinguish them from the parameters of the regression model (the $\mathbf{w}$s).

these variances a priori. In general, kernel functions are more complex than the linear one and they incorporate several hyperparameters. To set the hyperparameters of the covariance function for each specific problem, we can proceed as we do for the parameters $\mathbf{w}$. We first define a prior over these hyperparameters, then compute its posterior using Bayes rule, and finally integrate them out to obtain predictions. However in this case, the posterior is non-analytical. Hence the integration has to be done either by sampling or approximation. Although this approach is well principled [34], it is computational intensive and it is not feasible for communications systems. Alternatively, we can use the likelihood function of the hyperparameters and compute its maximum to obtain its optimal setting, which is used to describe the kernel for the test samples [34]. We use this second approach, because it is less computational demanding [24]. Moreover, as the number of training samples increases the posterior distribution peaks around its maximum likelihood estimate and the solutions of both approaches should not differ significantly.

C. Discussion

Gaussian Processes for regression is a general nonlinear regression tool that, given the covariance function, provides an analytical solution to any regression estimation problem. Moreover, it does not only give point estimates, but it also assigns confidence intervals for them. We perform the optimization step to set the hyperparameters of the covariance function by maximum likelihood, unlike SVM or other nonlinear machine learning tools, in which the optimization is used to set the optimal parameters. In SVM and other tools the hyperparameters have to be either prespecified or estimated by cross-validation [32]. Cross-validation optimizes several functionals, typically less than 10, for each possible setting of the hyperparameters [14]. The number of hyperparameters that can be tuned is quite limited (at most 2 or 3), as the computational complexity of cross-validation increases exponentially with the number of hyperparameters. These remarkable drawbacks limit the application of these nonlinear tools in digital communications receivers, since we face complex nonlinear problems with reduced computational resources and short training sequences. By exploiting the GPs framework, as stated in this paper, we can avoid them to build precise nonlinear receivers for DS-CDMA applications.

As mentioned earlier, the Gaussian processes framework can be extended for solving classification tasks. Gaussian process for classification (GPC) solution is non-analytical and we need to approximate its posterior distribution to make posterior probability predictions for the new samples and to train its hyperparameters. These approximations are computationally intensive and make GPC harder to train than GPR. Moreover, as detailed in [24] (Chapter 6), in many cases GPR performs equally well to GPC, if we are only interested in minimizing the misclassification rate. Therefore, although GPC seems the natural tool for solving this task, we decided to use GPR because it is less computationally demanding and its misclassification rate is similar to that of GPC.

IV. THE GPR-MUD DETECTOR

A. The GPR-MUD

The GPR mean prediction in (3) can be directly used as a nonlinear multi-user detector,

$$\hat{b}_*(j) = \text{sign}(\boldsymbol{\phi}^\top(\mathbf{x}_*)\mu_{\mathbf{w}_j}) = \text{sign}(\mathbf{k}^\top \mathbf{C}^{-1}\mathbf{y}). \qquad (7)$$

This GPR estimate resembles that of a linear detector. It has a weight vector, either $\mu_{\mathbf{w}}$ or $\mathbf{C}^{-1}\mathbf{y}$, multiplied by a nonlinear transformation, either $\boldsymbol{\phi}(\mathbf{x}_*)$ or $\mathbf{k}$, of the input to be predicted. The output of the GPR detector is the prediction of the bit transmitted by the UoI ($\mathbf{y}_i = b_i(j)$). The input to the GPR are Q consecutives samples from the channel:

$$\mathbf{x}_i = [\mathbf{r}_i^\top, \mathbf{r}_{i-1}^\top, \ldots, \mathbf{r}_{i-Q+1}^\top]^\top, \qquad (8)$$

where $\mathbf{r}_i$ are the N chips received at time step i in (1). If the codes of the other users are available, we could first project the received chips onto them as follows,

$$\mathbf{x}_i = [\mathbf{r}_i^\top \mathbf{S}, \mathbf{r}_{i-1}^\top \mathbf{S}, \ldots, \mathbf{r}_{i-Q+1}^\top \mathbf{S}]^\top. \qquad (9)$$

Next, we need to design the most suitable kernel function for our GPR-MUD receiver, which should be able to capture our prior knowledge and allow extracting unknown interactions. We propose, for the GPR-MUD, the following versatile covariance function

$$(\mathbf{C}_\theta)_{ij} = k(\mathbf{r}_i, \mathbf{r}_j) = \alpha_1 \exp\left(-\alpha_3 \sum_{\ell=1}^{d}(\mathbf{r}_i(\ell) - \mathbf{r}_j(\ell))^2\right) + \alpha_2 \mathbf{r}_i^\top \mathbf{r}_j + \alpha_0 \delta_{ij}, \qquad (10)$$

where the α_i weights need to be nonnegative for $\mathbf{C}_\theta$ to be positive-definite and, to deal with an unconstrained optimization problem, we define a vector of hyperparameters as $\theta = [\log \alpha_1, \log \alpha_2, \log \alpha_3, \log \alpha_0]$. This covariance function contains three terms. The second term is the linear covariance function. Therefore, the GPR model contains as a particular case the linear regressor ($\alpha_1 = 0$). The third term corresponds to $\sigma_\nu^2 \delta_{ij}$ in the definition of $\mathbf{C}$ in (5), which is considered as an extra hyperparameter of the covariance function. The first term is a radial basis kernel with the same length-scale for each input dimension, since in DS-CDMA all chips affect the solution in the same manner.

The covariance function in (10) is a good kernel for solving the GP-MUD, because it contains a linear and a nonlinear part. The optimal decision surface for MUD is nonlinear. However, in many cases, a linear detector is close to optimal as spreading codes are almost orthogonal to each other and to their delayed replicas. Typically, a minor nonlinear correction allows for optimal decisions. In this sense the proposed GPR covariance function is ideal for this problem. The linear part can mimic the best linear decision boundary and the nonlinear part modifies it, where the linear explanation is not optimal. Using a radial basis kernel for the nonlinear part is a good choice to achieve optimal nonlinear decisions. Because, the received chips form a constellation of 2^{KM_s} clouds with Gaussian spread around their centers. But we do not need to have exponentially many of them, as the linear terms explains away most of the received symbols.

B. The GP solution: a nonlinear MMSE

The MMSE criterion minimizes:

$$f_{\text{mmse}}(\cdot) = \underset{f(\cdot)}{\operatorname{argmin}} \operatorname{E}\left[(b - f(\mathbf{r}))^2\right] \tag{11}$$

in which b and $\mathbf{r}$ are, respectively, the transmitted and received symbols and $f(\cdot)$ is the function that estimates b from $\mathbf{r}$. It is widely known [42] that the minimum of (11) for AWGN channels is given by the conditional mean of b given $\mathbf{r}$:

$$\hat{b}_* = f_{\text{mmse}}(\mathbf{r}_*) = \operatorname{E}\left[b_*|\mathbf{r}_*\right], \tag{12}$$

which is a nonlinear function of the inputs, unless the inputs are Gaussian distributed.

In DS-CDMA, linear MMSE estimators are typically used, due to their simplicity and nearly optimal results in many standard scenarios. The linear, $f(\mathbf{r}) = \mathbf{r}^\top \mathbf{w}$, MMSE detector can be expressed as a Wiener filter [1]:

$$\mathbf{w}_{\text{mmse}} = \underset{\mathbf{w}}{\operatorname{argmin}} \operatorname{E}\left[\left(b - \mathbf{r}^\top \mathbf{w}\right)^2\right] = (\mathcal{R}_{\mathbf{rr}})^{-1}\mathcal{R}_{\mathbf{rb}}, \tag{13}$$

where $\mathcal{R}_{\mathbf{rr}}$ and $\mathcal{R}_{\mathbf{rb}}$ are, respectively, the autocorrelation of the inputs and the cross-correlation between the inputs and outputs.

Under mild conditions, the MMSE linear detector can be built from the spreading codes of every user and the channel's impulsive response:

$$\mathbf{w}_{\text{mmse}} = (\mathbf{P}\mathbf{P}^\top + \sigma_{\mathbf{n}}^2 \mathbf{I})^{-1}\mathbf{p}_j, \tag{14}$$

where $\mathbf{p}_j$ denotes the jth column of $\mathbf{P}$. If the other users' spreading codes or the channel state information are unknown, we need to replace $\mathcal{R}_{\mathbf{rr}}$ and $\mathcal{R}_{\mathbf{rb}}$ by their sampled versions:

$$\mathbf{w}_{\text{smmse}} = (\mathbf{R} \cdot \mathbf{R}^\top / L)^{-1}\mathbf{R} \cdot \mathbf{b}/L, \tag{15}$$

where the columns of $\mathbf{R}$ are the received chips at every symbol in the training set. We denote this solution as sampled MMSE (SMMSE).

The GPR-MUD estimates the output for a new incoming sample using (7), which can be simplified if we use a linear kernel ($\phi(\mathbf{r}) = \mathbf{r}$) to:

$$\mathbf{w}_{\text{gpr}} = \mu_{\mathbf{w}} = \sigma_{\mathbf{w}}^2 \mathbf{R} \left(\sigma_{\mathbf{w}}^2 \mathbf{R}^\top \mathbf{R} + \sigma_\nu^2 \mathbf{I}\right)^{-1}\mathbf{b}. \tag{16}$$

After some algebraic transformations we can express the linear GPR solution, up to a scaling constant, as

$$\mathbf{w}_{\text{gpr}} = \left(\mathbf{R}\mathbf{R}^\top + \frac{\sigma_\nu^2}{\sigma_{\mathbf{w}}^2}\mathbf{I}\right)^{-1}\mathbf{R}\mathbf{b}. \tag{17}$$

Since $\mathbf{R}\mathbf{R}^\top$ and $\mathbf{R}\mathbf{b}$ are, respectively, the empirical estimates of the correlations matrices $\mathcal{R}_{\mathbf{rr}}$ and $\mathcal{R}_{\mathbf{rb}}$, this expression is equal to (15) except for the term $\sigma_\nu^2/\sigma_{\mathbf{w}}^2\mathbf{I}$. This term endows the SMMSE solution in (15) with ridge regularization [14]. Ridge regression is a quite standard tool to avoid overfitting and improve the performance of the SMMSE detector. The regularization term must fade away as the number of training samples increases [10], thereby achieving the optimal solution for infinite samples. For short training sequences, the estimated correlation matrices represent poorly $\mathcal{R}_{\mathbf{rr}}$ and $\mathcal{R}_{\mathbf{rb}}$ and the regularization term should be kept high to avoid overfitting. A

good example of a regularized solution is the CMOE detector in [11],

$$\mathbf{w}_{\text{cmoe}} = (\mathbf{R}\mathbf{R}^\top / L + \alpha \mathbf{I})^{-1}\mathbf{R} \cdot \mathbf{b}/L, \tag{18}$$

where α is set using an *ad-hoc* procedure. Others examples of more involved methods for regularization can be found in [22], [15]. A common drawback of these approaches is that they do not clearly indicate how the regularization parameter is set and how it affects the solution. GPR provides a simple method, based on maximum likelihood, to compute the regularization. GPR trains its regularization parameter as to automatically choosing the best linear detector between the matched filter ($\sigma_\nu^2/\sigma_{\mathbf{w}}^2 \to \infty$) and the SMMSE ($\sigma_\nu^2/\sigma_{\mathbf{w}}^2 = 0$), improving the performance of the MUD-SMMSE.

The nonlinear GPR-MUD inherits this property from the linear case, as the nonlinear GPR can be understood as a linear solution in a transformed space. Therefore, if the proposed kernel is flexible to approximate any desired function, the GPR predictions are optimal in the MMSE sense. The kernel we proposed in Section IV-A is universal and it is able to approximate (12) and produce optimal decisions.

C. Computational load

GPR uses the 'kernel trick' [31], [41], [24] to achieve nonlinear solutions without explicitly describing its nonlinear transformation to the feature space, which can be infinite dimensional. The computational complexity of GPRs is limited by the inversion of the covariance matrix, which is $\mathcal{O}(L^3)$. This inversion is needed in every step of any gradient-based optimization method to estimate the hyperparameters [24] in the training stage. Then, the result of the training stage is used to estimate the output for any new received data at a computational cost given just by the computation of $\mathbf{k}$ in (3).

For large training sequences this computational complexity can be prohibitively large and we need to approximate the GPR solution to deal with thousands of samples, as we show in short. In the linear GPR solutions, if the input dimension is lower than the number of training samples ($L > D$), the computational complexity reduces to $\mathcal{O}(D^3)$ as we derive next, which makes the complexity for this algorithm independent of the number of training examples and similar to that of the linear SMMSE-MUD.

1) The general case: There are several proposals in the literature to reduce the complexity of the GPR training and prediction stages [43]-[47]. A detailed review of all these methods can be found in [48]. The main motif of these approaches is representing the covariance matrix of the GP with a reduced set of J samples (or pseudo-samples) much smaller than L, with the same accuracy as the whole training dataset.

The algorithms in [43]-[47] propose different ways of selecting the J samples in the reduced set. Some of them train the locations of the pseudo-samples. This allows obtaining a further reduced set at a larger computational complexity during training. Other approaches use heuristics to select the pseudo-samples from the training set. Also, the pseudo-samples can be choosen incrementally until a desired accuracy is achieved.

2) Linear GPR: If we are interested only in linear detectors, the complexity of each iteration of the optimization procedure can be significantly reduced from $\mathcal{O}(L^3)$ to $\mathcal{O}(\min\{L, D\})$ without loss of accuracy. We propose to run a very simple algorithm for computing the linear GPR-MUD. For a linear kernel the covariance function becomes:

$$\mathbf{C}_\theta = e^{\theta_1}\mathbf{X}^\top\mathbf{X} + e^{\theta_2}\mathbf{I} = e^{\theta_1}\left(\mathbf{X}^\top\mathbf{X} + e^{\theta_2-\theta_1}\mathbf{I}\right), \quad (19)$$

where the columns of $\mathbf{X}$ are the sample inputs in (8) and θ_1, θ_2 are the to be estimated hyperparameters. We can write $\mathbf{X}^\top\mathbf{X}$ in terms of its eigenvalues:

$$\mathbf{X}^\top\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top, \quad (20)$$

where $\mathbf{U}$ is an orthonormal matrix, containing the eigenvectors of $\mathbf{X}^\top\mathbf{X}$, and $\mathbf{\Lambda}$ is a diagonal matrix with its eigenvalues. This representation is useful to compute the inverse of $\mathbf{C}_\theta$ as

$$\mathbf{C}_\theta^{-1} = e^{-\theta_1}\mathbf{U}\left(\mathbf{\Lambda} + e^{\theta_2-\theta_1}\mathbf{I}\right)^{-1}\mathbf{U}^\top. \quad (21)$$

This eigenvalue-eigenvector representation simplifies the gradient with respect to θ_i, as follows:

$$\frac{\partial l(\theta)}{\partial\theta_1} = \frac{1}{2}\sum_{i=1}^{D}\frac{\lambda_i}{\lambda_i + e^{\theta_2-\theta_1}} + \frac{1}{2}\sum_{i=1}^{D}\frac{\lambda_i z_i^2 e^{-\theta_1}}{(\lambda_i + e^{\theta_2-\theta_1})2} \quad (22)$$

$$\frac{\partial l(\theta)}{\partial\theta_2} = \frac{1}{2}\sum_{i=1}^{D}\frac{e^{\theta_2-\theta_1}}{\lambda_i + e^{\theta_2-\theta_1}} + \frac{1}{2}\sum_{i=1}^{D}\frac{e^{\theta_2-2\theta_1}z_i2}{(\lambda_i + e^{\theta_2-\theta_1})2}, \quad (23)$$

where $\mathbf{z} = \mathbf{U}^\top\mathbf{y}$. At most, the number of nonzero eigenvalues is $\min(L, D)$. We have used D in the previous equations as in most cases $L > D$.

The complexity of the linear SMMSE-MUD is $\mathcal{O}(D^3)$ as it needs to invert a $D \times D$ matrix. The complexity of training the GPR is linear in D, once we have computed its eigenvalue decomposition which is $\mathcal{O}(D^3)$. Each optimization step of the GPR is insignificant with respect to the matrix inversion. Therefore, the complexity of the GPR detector with a linear kernel is of the same order of magnitude as that of the SMMSE-MUD.

V. EXPERIMENTAL RESULTS

A. Regularization

We first propose the same scenario as in [15] with $K = 10$ active users and Gold spreading sequences of length $N = 31$. The amplitudes of the interferer were equal to that of the user of interest. We include the average for 300 simulated chip-spaced channels of length $M_c = 15$ with equally distributed zero-mean random Gaussian for the channel paths. We used $Q = 2$ in (8).

We compare the BER of the SMMSE detector in (15) to the GPR with linear kernel ($\alpha_1 = 0$ in (10)), the SVM with linear kernel and soft margin 0.5, the CMOE in (18) and the tappered regularized method (CMT) in [15]. The regularization parameters for the CMOE and the CMT where set as described in [49], where the covariance matrix was estimated with $L = 372$ samples. The training sequences were generated randomly for every channel. The BER was estimated with test inputs different from the training sequences. We report the BER as
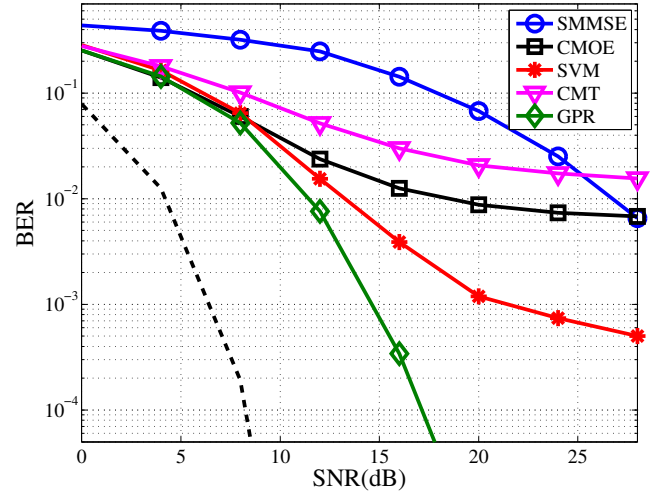

Fig. 1

BER ALONG THE SNR IN A CDMA SCENARIO WITH $K = 10$ USERS AND GOLD SEQUENCES WITH $N = 31$ FOR LINEAR SVM ($*$), CMT (∇), CMOE ($\square$), SMMSE ($\circ$), AND LINEAR GPR ($\diamond$) MUD FOR $L = 64$ TRAINING SAMPLES.
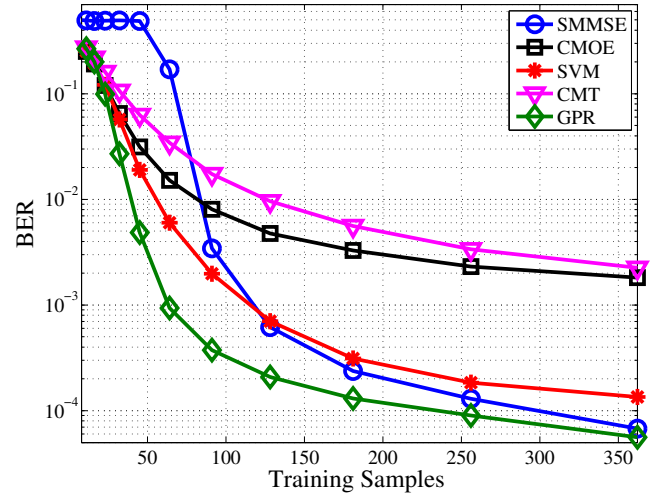

Fig. 2

BER ALONG THE NUMBER OF TRAINING DATA IN A CDMA SCENARIO WITH $K = 10$ USERS AND GOLD SEQUENCES WITH $N = 31$ FOR LINEAR SVM ($*$), CMT (∇), CMOE ($\square$), SMMSE ($\circ$), AND LINEAR GPR ($\diamond$) MUD FOR A SNR=15DB.

a function of the SNR for $L = 64$ samples in Fig. 1 and the BER as a function of the training samples for an SNR of 15dB in Fig. 2.

The linear GPR-MUD receiver clearly outperforms the other detectors for short training sequences for the SNR range of interest, see Fig. 1. The SVM is the second best procedure, although its performance is significantly poorer than that of the GPR detector. The CMOE and CMT use a fixed regularization procedure that precludes them to perform well for all training

sequences and SNRs, which is a severe limitation as these detectors must perform well in many different scenarios. The SMMSE convergence is significantly slower than that of the GPR and, for short training sequences, its BER is orders of magnitude above the GPR-MUD detector.

In Fig. 2, we compare the performance of the different methods as the number of training samples increases. As we expected, once the number of training samples is long enough, the SMMSE and GPR detectors tend to coincide. But for shorter training sequences the GPR-MUD receiver learns much faster than the SMMSE one. This is a very important feature, as the longer the training sequence needs to be the fewer information bits we can transmit in each burst of data. The linear SVM solution follows the GPR although there is a constant gap between its solution and the GPR; this is due to the fixed soft margin parameter used for all lengths of the training sequence. The CMOE and CMT present poor performances, even for $L = 372$. This experiment illustrates one of the major advantages of GPR detectors for CDMA communications: they can tune their hyperparameters to find the best regularized linear solution. Its capability of resorting to nonlinear solutions is its other main advantage, as illustrated next.

We now consider a detector in a simple situation for $K = 2$ users transmitting with the same power. The spreading factor is $N = 4$ and the spreading codes are $[+1 \ +1 \ -1 \ -1]$ and $[+1 \ -1 \ -1 \ +1]$. The UoI corresponds to the second code, i.e. $j = 2$, and the channel impulse response is given by

$$c(z) = 0.3 + 0.7z^{-1} + 0.3z^{-2}. \tag{24}$$

The inputs were as in (9) with $Q = 1$. Both the SVM and the GPR use a Gaussian kernel ($\alpha_2 = 0$ in (10)). Following [28] for the SVM, the width of the kernel is set to the noise standard deviation and the soft margin to 0.6. In Fig. 3 we depict the decision boundaries for a signal to noise ratio (SNR) of 9 dB. We include the optimal one-shot detector in (2) (dash-dotted), the SVM-MUD (dotted) and the GPR-MUD (solid). The bit error rate (BER) achieved by the SVM-MUD and the GPR-MUD are similar, although the boundaries are quite different. We observe that the SVM-MUD is unable to generalize properly. The SVM boundary is sinuous, causing more than two decision regions, and in an eventual decrease in SNR, it would present a poor BER performance. In comparison, GPR-MUD would degrade its performance gracefully, due to its regularized solution.

B. A flexible solution

In the previous experiments the kernels were either lineal or nonlinear. Since in digital communications the transmitter, radio channel and receivers are typically close to linear systems, we propose the kernel in (10) to estimate the incoming bits. The linear part of this covariance matrix allows a fast learning of the system response and the nonlinear part adapts the linear solution to accommodate the nonlinearities. Using the same kernel in other machine learning tools, such as SVM, is cumbersome, as we have a large number of hyperparameters to learn by means of cross-validation.
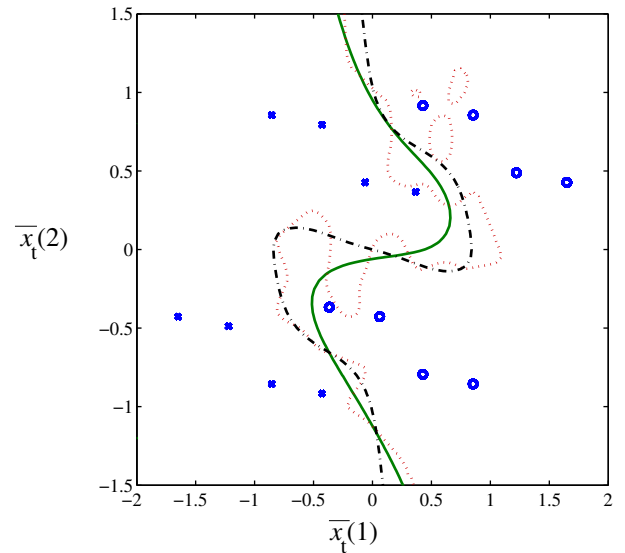

Fig. 3

DECISION BOUNDARY IN A CDMA SCENARIO WITH 2 USERS, 160 TRAINING SAMPLES AND A SNR OF 9 dB FOR SVM (DOTTED), GP (SOLID) AND THE OPTIMUM ONE-SHOT (DASH-DOTTED) CENTRALIZED MUD.

We repeat Example 2 in [28], where the spreading factor is $N = 8$ and we have $K = 3$ users with equal power. We report the BER for Users 2 and 3. The spreading codes are, respectively, $[+1 \ +1 \ +1 \ +1 \ -1 \ -1 \ -1 \ -1]$, $[+1 \ -1 \ +1 \ -1 \ -1 \ +1 \ -1 \ +1]$ and $[+1 \ -1 \ -1 \ +1 \ -1 \ +1 \ +1 \ -1]$ and the channel response is given by:

$$c(z) = 0.4 + 0.9z^{-1} + 0.4z^{-2}. \tag{25}$$

For comparison purposes, we include the BER for the SVM-MUD in [28] and SMMSE-MUD. Unlike [28], were the authors use the chips projected onto the users spreading codes, we use the chips as inputs (8) with $Q = 1$. We also include the BER for the ideal case, memoryless channel without interfering users (dashed). The covariance matrix of the GPR-MUD is given by (10). The SVM-MUD is trained using a Gaussian kernel with its width equal to 3 times the noise standard deviation and the soft margin parameter [2] set to 0.6.

In Fig. 4 we report the BER along the SNR. We depict the averaged results for 1000 independent experiments with 10^5 test samples and the same 32 training samples in each run with different noise. For such short training sequence, the SVM cannot learn a good classifier and it is outperformed by the GPR-MUD and the linear SMMSE-MUD. The GPR, as the SVM, does not have enough training examples to build a nonlinear classifier. However, the GPR is able to "see" that the training sequence is too short to train a nonlinear classifier and it resorts to the linear SMMSE solution.

---

[2]In [28] the authors do not report the width used in the experiments, but they say it is related to the noise standard deviation. We found that 3 times the standard deviation of the noise for the kernel width provided good results for the SVM-MUD.
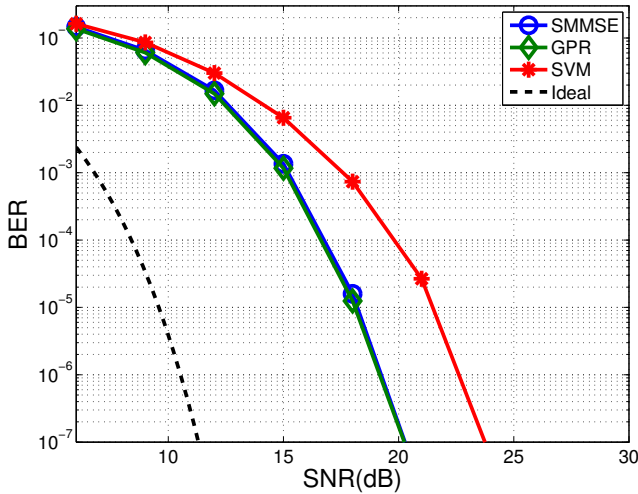
Fig. 4

BER FOR USER 2 IN EXPERIMENT 2: A CDMA SCENARIO WITH $K = 3$ USERS, $N = 8$ AND 32 TRAINING SAMPLES. SMMSE ($\circ$), SVM ($*$) AND GPR ($\diamond$) MUD.

Fig. 5

BER FOR USER 3 IN EXPERIMENT 2: A CDMA SCENARIO WITH $K = 3$ USERS, $N = 8$ AND 100 TRAINING SAMPLES. SMMSE ($\circ$), SVM ($*$) AND GPR ($\diamond$) MUD.

In Fig. 5, we plot the BER for User 3 averaged over 1000 experiments with 10^5 test samples and 100 random training examples. The optimal decision boundary is nonlinear. This scenario is illustrative of the nice properties of the GPR detector compared to the SVM-MUD receiver. For low SNR both the GPR and SVM-MUD obtain a nonlinear detector that outperforms the linear SMMSE. For high SNR and short training sequence, the nonlinear algorithms are unable to improve the linear solution. The GPR-MUD mimics the linear SMMSE and the SVM solution degrades, unable to improve the linear SMMSE detector.

Since for real scenarios the number of training samples is limited, the GPR provides optimal results either by obtaining the best nonlinear detector or by mimicking the linear SMMSE-MUD, if there is not enough information available.

VI. CONCLUSIONS AND FURTHER WORK

In this paper we have introduced Gaussian Processes for Regression (GPR) as a nonlinear detector for DS-CDMA digital communications systems. The GPR is a discriminative learning tool and it does not assume anything about the CDMA communication system. It does not need to know how many users are active and what spreading codes they are using. It does not need to know the channel model or its length. It only relies on a training sequence for the UoI to detect the incoming chips and it can train linear and nonlinear models depending on which suits the application best. This makes it a very desirable tool for designing CDMA MUD receivers.

GPR solution is analytical and its hyperparameters can be learnt by maximum likelihood. This is a considerable improvement compared to other nonlinear tools as neural networks or SVMs, which need to prespecify its structure or hyperparameters because the optimization step is taken to find the optimal pa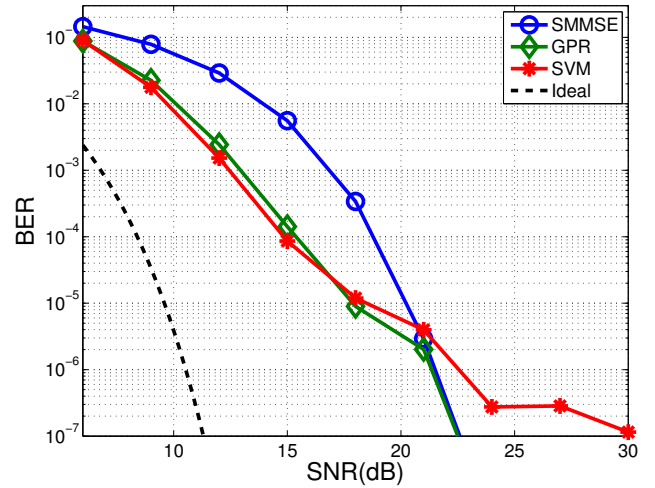rameters. This extra flexibility allows GPR to outperform these nonlinear classifiers for MUD in CDMA. They provide better solutions at shorter training sequences.

We have shown that the GPR solution can be understood as a nonlinear MMSE. The linear part of the GPR-MUD performs as the linear MMSE for large training sequences. For short training sequences, the GPR-MUD outperforms the linear MMSE-MUD, because it trains its regularization hyperparameter to accommodate the received training sequence. The GPR-MUD receiver with linear kernel is able to optimally set the regularization parameter, instead of relying on fixed or ad-hoc procedures for selecting it.

The linear GPR-MUD, as presented in this paper, can be implemented directly for fast-fading multi-path channels and its computational complexity is similar to that of the SMMSE. For the nonlinear GPR, we have used the full covariance matrix and its optimization can be computationally costly in some scenarios. There are several proposals that address this computational complexity issue [43]-[47], which can be used to implement the proposed GPR-MUD with low computational complexity with nonlinear kernels.

REFERENCES

[1] S. Verdú, *Multiuser Detection*. Cambridge University Press, 1998.
[2] R. Prasad, *CDMA for Wireless Personal Communications*. Norwood, MA: Artech House, 1996.
[3] M. Zeng, A. Annamalai, and V. K. Bhargava, "Recent advances in cellular wireless communications," *IEEE Communication Magazine*, vol. 37, no. 1, pp. 128–138, Sep 1999.
[4] J. G. Proakis, *Digital Communications*, 4th ed. New York, NY: McGraw-Hill, 2000.

[5] S. Verdú, "Minimum probability of error for asynchronous gaussian multiple access channels," *IEEE Transactions on Information Theory*, vol. 32, pp. 85–96, Jan 1986.

[6] R. Lupas and S. Verdu, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Transactions on Information Theory*, vol. 35, no. 1, pp. 123–136, Jan 1989.

[7] Z. Xie, R. Short, and C. Rushforth, "A family of suboptimum detectors for coherent multiuser communications," *IEEE J. Selected Areas on Communications*, vol. 8, pp. 683–690, 1990.

[8] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Transactions on Communications*, vol. 42, pp. 3178–3188, 1994.

[9] H. V. Poor and S. Verdú, "Probability of error in MMSE multiuser detection," *IEEE Transactions on Information Theory*, vol. 43, pp. 858–871, 1997.

[10] A. N. Tikhonov and V. Y. Arsenin, *Solution of Ill-posed Problems*. Wiston, Washington DC, 1977.

[11] U. Madhow, "Blind adaptive interference suppression for direct sequence CDMA," *Proceedings of the IEEE*, vol. 86, pp. 2048–2069, Oct 1998.

[12] J. W. Shubao Liu, "Blind adaptive multiuser detection using a recurrent neural netowrk," in *IEEE Int. Conf. on Communications, Circuits and Systems (ICCCAS)*, vol. 2, Jun 2004, pp. 1071–1075.

[13] K. Zarifi, S. Shahbazpanahi, A. B. Gershman, and Z.-Q. Luo, "Robust blind multiuser detection based on the worst-case performance optimization of the mmse receiver," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 295–296, Jan 2005.

[14] C. M. Bishop, *Neural Networks for Pattern Recognition*. Clarendon Press, 1995.

[15] L. Rugini, P. Banelli, and S. Cacopardi, "A full rank regularization technique for MMSE detection in multiuser CDMA systems," *IEEE Communications Letters*, vol. 9, no. 1, p. 200, May 2005.

[16] G. Wahba and S. Wold, "A completely automatic french curve: fitting spline functions by cross-validation," *Communications in Statistics*, vol. Series A 4, no. 1, pp. 257–263, 1975.

[17] M. I. Jordan, Ed., *Learning in Graphical Models*. Cambridge, MA: MIT Press, 1999.

[18] K.-S. C. Anders Høst-Madsen, "MMSE/PIC multiuser detection for DS/CDMA systems with inter- and intra-cell interference," *IEEE Transactions on Communications*, vol. 47, no. 2, p. 1999, Feb 1999.

[19] Y. Kabashima, "A CDMA multiuser detection algorithm on the basis of belief propagation," *Journal of Physics A: Mathematical and General*, vol. 36, no. 2003, pp. 11 111–11 121, Oct 2003.

[20] M. O. T. Tanaka, "Approximate belief propagation, density evolution, and statistical neurodynamics for CDMA multiuser detection," *IEEE Transactions on Information Theory*, vol. 51, no. 2, pp. 700–706, Feb 2005.

[21] L. K. R. P. H. Tan, "Asymptotically optimal non-linear MMSE multiuser detection based on multivariate gaussian approximation," *IEEE Transactions on Communications*, vol. 54, no. 8, pp. 1427–1438, Aug 2006.

[22] D. D. Lin and T. J. Lim, "A variational free energy minimization interpretation of multiuser detection in CDMA," in *IEEE Global Telecommunications Conference (Globecom)*, vol. 3, St. Louis, MO, USA, Dec 2005.

[23] V. N. Vapnik, *Statistical Learning Theory*. New York: John Wiley & Sons, 1998.

[24] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[25] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," Royal Holloway College, University of London, UK, Tech. Rep. NC-TR-98-030, 1998, ftp://www.neurocolt.com/pub/neurocolt/tech_reports/1998/98030.ps.Z.

[26] U. Mitra and H. V. Poor, "Neural network techniques for adaptive multiuser demodulation," *IEEE Journal Selected Areas on Communications*, vol. 12, pp. 1460–1470, 1994.

[27] R. Tanner and D. G. M. Cruickshank, "Volterra based receivers for DS-CDMA," in *IEEE Int. Symp. Personal, Indoor, Mobile Radio Commun.*, vol. 3, September 1997, pp. 1166–1170.

[28] S. Chen, A. K. Samingan, and L. Hanzo, "Support vector machine multiuser receiver for DS-CDMA signals in multipath channels," *IEEE Transactions on Neural Network*, vol. 12, no. 3, pp. 604–611, December 2001.

[29] B. Aazhang, B. P. Paris, and G. C. Orsak, "Neural networks for multiuser detection in code-division multiple-access communications," *IEEE Transactions on Communications*, vol. 40, pp. 1212–1222, 1992.

[30] D. G. M. Cruickshank, "Radial basis function receivers for DS-CDMA," *IEE Electronic Letter*, vol. 32, no. 3, pp. 188–190, 1996.

[31] B. Schölkopf and A. Smola, *Learning with kernels*. M.I.T. Press, 2001.

[32] G. S. Kimeldorf and G. Wahba, "Some results in Tchebycheffian spline functions," *Journal of Mathematical Analysis and Applications*, vol. 33, pp. 82–95, 1971.

[33] C. K. I. Williams and C. E. Rasmussen, "Gaussian processes for regression," in *Proc. Conf. Advances in Neural Information Processing Systems, NIPS*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds., vol. 8. MIT Press, 1995.

[34] C. Williams, "Prediction with gaussian processes: From linear regression to linear prediction and beyond," in *Learning and Inference in Graphical Models*, M. Jordan, Ed. Kluwer Academic Press, 1998.

[35] J. Murillo-Fuentes, S. Caro, and F. Perez-Cruz, "Gaussian processes for multiuser detection in CDMA receivers," in *In Advances in Neural Information Processing Systems, NIPS*, Vancouver, CA, Dec. 2005.

[36] F. Pérez-Cruz and J. Murillo-Fuentes, "Gaussian processes for digital communications," in *ICASSP*, vol. V, Tolousse, France, May 2006, pp. 781–784.

[37] T. S. Rappaport, *Wireless Communications*. Prentice Hall PTR, 2002.

[38] J. G. Proakis, "Adaptive equalization for TDMA digital mobile radio," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 2, pp. 333–341, Feb. 1991.

[39] C. K. I. Williams and D. Barber, "Bayesian classification with gaussian processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1342–1351, 1998.

[40] M. Kuss and C. E. Rasmussen, "Assessing approximations for Gaussian process classification," in *Proc. Conf. Advances in Neural Information Processing Systems, NIPS*, Y. Weiss, B. Schölkopf, and J. Platt, Eds., vol. 18. MIT Press, 2006, pp. 699–706.

[41] F. Pérez-Cruz and O. Bousquet, "Kernel methods and their potential use in signal processing," *Signal Processing Magazine*, vol. 21, no. 3, pp. 57–65, 2004.

[42] D. Guo, S. Shamai, and S. Verdú, "Mutual information and minimum mean-square error in gaussian channels," *IEEE Trans. Information Theory*, vol. 51, no. 4, pp. 1261–1283, Apr. 2005.

[43] A. J. Smola and P. L. Bartlett, "Sparse greedy gaussian process regression," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. Cambridge, MA: MIT Press, 2001.

[44] L. Csat'o and M. Opper, "Sparse online gaussian processes," *Neural Computation*, vol. 14, no. 3, pp. 641– 669, 2002.

[45] M. Seeger, C. K. I. Williams, and N. Lawrence, "Fast forward selection to speed up sparse gaussian process regression," in *International Workshop on Artificial Intelligence and Statistics*, C. Bishop and B. J. Frey, Eds., 2003.

[46] E. Snelson and Z. Ghahramani, "Sparse gaussian processes using pseudo-inputs," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006.

[47] E. Snelson and Z. G. (2007), "Local and global sparse gaussian process approximations," in *Artificial Intelligence and Statistics*, vol. 11, 2007.

[48] J. Quiñonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate gaussian process regression," *Journal of Machine Learning Research*, vol. 6, pp. 1935–1959, 2005.

[49] L. Rugini, P. Banelli, and S. Cacopardi, "Regularized MMSE multiuser detection using covariance matrix tapering," in *EEE International Conference on Communications*, vol. 4, May 2003, pp. 2460 – 2464.