Report: Complex Support Vector Detector for Large MIMO System

Tianpei Chen

Department of Electrical and Computer Engineering

McGill University

October 11, 2015

I. Introduction

One of the biggest challenges the researchers and industry practitioners are facing in wireless communication area is how to bridge the sharp gap between increasing demand of high speed communication of rich multimedia information with high level Quality of Service (QoS) requirements and the limited radio frequency spectrum over a complex space-time varying environment. The most promising technology for solving this problem, Multiple Input Multiple Output (MIMO) technology has been of immense research interest over the last several tens of years is incorporated into the emerging wireless broadband standard like 802.11ac [1] long-term evolution (LTE) [2]. The core idea of MIMO system is to use multiple antennas at both transmitting and receiving end, so that multiplexing gain (multiple parallel spatial data pipelines that can improve bandwidth efficiency) and diversity gain (better reliability of communication link) is obtained by exploiting the spatial domain. Large MIMO (also called Massive MIMO) is an upgraded version of conventional MIMO technology employing hundreds of low power low price antennas at base station (BS), that serves several tens of terminals simultaneously. This technology can achieve full potential of conventional MIMO system while providing additional power efficiency as well as system robustness both to unintended man-made interference and intentional jamming. [3] [4].

The price paid for large MIMO system is the increased complexities for signal processing at both transmitting and receiving end. The uplink Detector is one of the key components in large MIMO systems. With orders magnitude more antennas at the BS, benefits and challenges coexist in designing of detection algorithms for the uplink communication of large MIMO systems. On one hand, a large number of receive antennas provide potential of large diversity gains, on the other hand, complexity of the algorithm becomes crucial to make the system practical.

Vertical Bell Laboratories Layered Space-Time (V-BLAST) architecture for MIMO system can achieve high spectrum efficiency by spatial multiplexing (SM), that is, each transmit antenna transmits independent symbol streams. However the optimal maximum likelihood detector (MLD) for V-BLAST systems that perform exhaustive search has a complexity that increases exponentially with number of transmitted antennas, which is prohibitive for practical applications.

As alternatives to MLD, linear detectors (LD) such as zero-forcing (ZF) and minimum mean square error (MMSE) with optimized ordering sequential interference cancellation (ZF-OSIC, MMSE-OSIC) are exploited in V-BLAST architecture [5] [6] [7], however the performance of ZF-OSIC and MMMSE-OSIC are inferior comparing to MLD.

Sphere Decoder (SD) [8] is the most prominent algorithm that utilizes the lattice structure of

MIMO systems, which can achieve optimal performance with relatively much lower complexity comparing to MLD. However, SD has two major shortages that make it problematic to be integrated into a practical systems. The first shortage is SD has various complexities under different signal to noise ratios (SNR), while a constant processing data rate is required for hardware. The second shortage is SD's complexity still has a lower bound for complexity that increases exponentially with the number of transmit antennas and the order of modulation scheme [9]. The fixed complexity sphere decoder (FCSD) [10] makes it possible to achieve near optimal performance with a fixed complexity under different value of SNR. The FCSD inherits the principle of list based searching algorithms, which first generate a list of candidate symbol vectors and then the best candidate is chosen as the solution. The other sub optimal detectors belong to this class include Generalized Parallel Interference Cancellation (GPIC) [11] and Selection based MMSE-OSIC(sel-MMSE-OSIC) [12]. However, all these list based searching algorithms have the same shortage - their complexities increase exponentially with the number of transmit antennas and the order of modulation scheme [12]. Therefore, such algorithms are prohibitive when it comes to a large number of antennas or a high order modulation scheme, for example in IEEE 802.11ac standard [1], the modulation scheme is 256QAM.

Besides the above detection algorithms designed for conventional MIMO systems, in the last several years, a set of detection algorithms have been proposed for large MIMO systems with complexities that are comparable with MMSE detector and near-optimal performance. such algorithms include likelihood ascend searching (LAS) algorithms [13] [14], Tabu search based algorithms which have superior performance compared to LAS detectors because local minima can be avoided (e.g. Layered Tabu search (LTS) [15], Random Restart Reactive Tabu search (R3TS) [16]), Message passing technique based algorithms (e.g. Belief propagation (BP) detectors based on graphic model and Gaussian Approximation (GA) [17] [18] [19] [20]), Probabilistic Data Association based algorithms [21], Monte Carlo sampling based algorithms (e.g. Markov Chain Monte Carlo (MCMC) algorithm [22]) and Lattice Reduction (LR) aided algorithms [23].

Firmly grounded in framework of statistical learning theory, the Support Vector Machine (SVM) technique has become a powerful tool to solve real world supervised learning problems such as classification, regression and prediction. the SVM method is a nonlinear generalization of Generalized Portrait algorithm developed by Vapnik in 1960s [24] [25], which can provide good generalization performance [26].

Interest in SVM boosted since 1990s, promoted by the works of Vapnik and co-workers at AT& T Bell laboratory [27] [28] [29] [30] [31] [32]. Moreover, the kernel based methods [26] solve nonlinear learning tasks by mapping input data sets into high dimensional feature spaces, and replacing inner products of feature mappings by computational inexpensive kernel functions discarding the actual structure of the feature space. This rationale is supported mathematically by the notion of Reproducing Kernel Hilbert Space (RKHS). Based on the same regularized risk function principle, ϵ-Support Vector Regression (ϵ-SVR) was developed [29] [33].

Similar to SVM, the ϵ-SVR solves an original optimization problem by transforming it into a Lagrange dual optimization problem, which can be solved by Quadratic Programming (QP). Sequential Minimal Optimization (SMO) algorithm was proposed as a fast algorithm to solve this QP problem by decomposing the it into sub QP problems and solving them analytically [34]. Therefore, the computational intensive numerical method can be avoided. A more general method is decomposition solver, which refers to a set of algorithms that separate the optimization variables (Lagrange multipliers) into two sets W and N, W is the work set and N contains the remaining optimization variables. In each iteration, only the optimization variables in the work set is optimized while keeping other variables fixed. The SMO algorithm is an extreme case of decomposition solver. An important issue of decomposition solver is the choice of the work set. One strategy is to choose Karush-Kuhn-Tucker (KKT) condition violators, ensuring final converge [35]. Because of the linear constraint inducted by offset, the SMO algorithm restricts the size of the work set to 2. In [36], a method to train SVM without offset was proposed, with the comparable performance to the SVM with offset. A set of sequential single variable work set selection strategies, which require $O(n)$ searching time are proposed. The optimal double variable work set selection strategy, which performs exhaustively searching, however, requires $O(n^2)$ searching time. The authors demonstrate that with the combination of two proposed single variable work set selection strategies, convergence can be achieved by a iteration time that is as few as optimal double variable work set selection strategy.

The mathematical foundation of kernel based methods is RKHS which is defined in complex domain, however most of the practitioners are dealing with real data sets. In communication and signal processing area, the channel gains, signals, waveforms etc. are all represented in complex form. Recently, a pure complex SVR & SVM based on complex kernel was proposed in [37], which can deal with the complex data set purely in complex domain. The results in [37] demonstrate better performances as well as reduced complexity comparing to simply split learning task into two real case by real kernel. Based on this work, we derive of a complexity-performance controllable detector for large MIMO systems based on a dual channel complex

SVR (CSVR). The detector can work in two parallel real SVR channels which can be solved independently. Moreover, only the real part of kernel matrix is needed in both channels. This means a large amount of computation can be reduced. Based on the discrete time MIMO channel model, in our regression model, this CSVR-detector is constructed without offset, Therefore, for each real SVR without offset, in principle, only one variable is needed to be updated in each iteration, In our scheme, a sequential single variable selection strategy is proposed. By this strategy, two variables can be updated at each iteration, with much smaller searching time.

II. Brief Introduction to ϵ-Support Vector Regression

A. Regression Model

Suppose we are given training data set $((\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_L, y_L))$, L denotes the number of training samples, $\mathbf{x}_i \in \mathbb{R}^V$ denotes input data vector, V is the number of features in $\mathbf{x}_i$. y_i denotes output. Let $\mathbf{w}$ denotes regression coefficient vector, $\Phi(\mathbf{x}_i)$ denotes the mapping of $\mathbf{x}_i$ to higher dimensional feature space, $\mathbf{w}, \Phi(\mathbf{x}_i) \in \mathbb{R}^\Lambda$, $\Lambda \in \mathbb{R}$ denotes the dimension of mapped feature space (For linear model, $\mathbf{x}_i = \Phi(\mathbf{x}_i)$). The regression model (either linear or non-linear) is given by

$$y_i = \mathbf{w}^T \Phi(\mathbf{x}_i) + b \quad i = 1 \cdots L \tag{1}$$

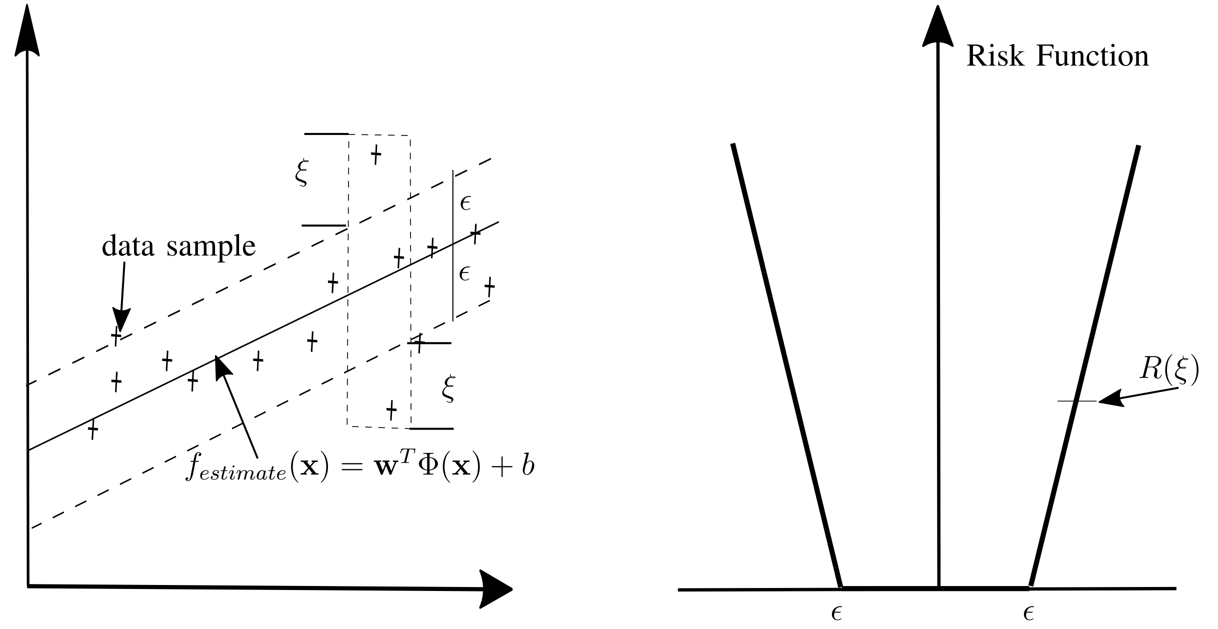

Fig. 1. ϵ-Support Vector Regression and Risk Functional

We present the primal optimization problem directly

$$\min_{\mathbf{w}} f(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{L}(R(\xi_i) + R(\hat{\xi}_i))$$

$$s.t. \begin{cases} y_i - \mathbf{w}^T\Phi(\mathbf{x}_i) - b \leq \epsilon + \xi_i, i = 1, 2\cdots, L \\ \mathbf{w}^T\Phi(\mathbf{x}_i) + b - y_i \leq \epsilon + \hat{\xi}_i, i = 1, 2\cdots, L \\ \epsilon_i, \xi_i, \hat{\xi}_i \geq 0, i = 1, 2\cdots, L \end{cases} \tag{2}$$

In (2), $\frac{1}{2}||\mathbf{w}||^2$ is the regularization term in order to ensure the flatness of regression model, ϵ

denotes the precision, As shown in Fig 1, the area between two dash line is called ϵ tube. Only those data samples located outside ϵ tube contribute to cost. Furthermore ξ_i and $\hat{\xi}_i$ denote slack variables that cope with noise of input data samples, $R(u)$ denotes risk function, the simplest risk function is $R(u) = u$, The type of cost function is determined by the statistical distribution of noise [33]. For example if the noise is Gaussian noise, then the optimal cost function is $R(u) = \frac{1}{2}u^2$. The term $C\sum_{i=1}^{L}(R(\xi_i) + R(\hat{\xi}_i))$ denotes the penalty of noise, $C \in \mathbb{R}$ and $C \geq 0$ that controls the trade off between regularization term and cost function term.

In ϵ-SVR, the objective to exploit slack variables ξ_i and $\hat{\xi}_i$ is to compensate the influences from the outliers that exceed the ϵ-tube which are caused by noise. Therefore in ϵ-SVR, ξ_i and $\hat{\xi}$ are defined as

$$\xi_i = \max(0, \mathbf{y}_i - \mathbf{w}^T\Phi(\mathbf{x}_i) - b - \epsilon) \tag{3}$$

$$\hat{\xi}_i = \max(0, \mathbf{w}^T\Phi(\mathbf{x}_i) + b - \mathbf{y}_i - \epsilon) \tag{4}$$

Because the distance of the estimations $\mathbf{w}^T\Phi(\mathbf{x}_i) + b$ and the observation $\mathbf{y}_i$ can only exceeds the ϵ-tube in one direction, therefore there is at most one of ξ_i and $\hat{\xi}_i$ can be non zero. That is $\xi_i\hat{\xi}_i = 0$.

B. Cost Function

The optimal cost function in (2) can be derived based on maximum likelihood (ML) principle. Assume the data samples $\mathbf{x}_i$ in data set are iid, Let $f_{true}(\mathbf{x}_i), i = 1, 2, \cdots L$ denotes true regression function. the underlying assumption is $y_i = f_{true}(\mathbf{x}_i) + \xi_i, i = 1, 2, \cdots, L$, ξ_i denotes additive noise of the ith data sample, with probability density function (pdf) $Pr(\cdot)$. Let $P(\cdot)$ denotes the pdf of y_i. Based on ML principle we want to

$$\max_f \quad \prod_{i=1}^{L} P(y_i|f(\mathbf{x}_i)) = \prod_{i=1}^{L} P(f(\mathbf{x}_i)+\xi_i|f(\mathbf{x}_i)) = \prod_{i=1}^{L} Pr(\xi_i) = \prod_{i=1}^{L} Pr(y_i - f(\mathbf{x}_i)) \tag{5}$$

Take the logarithm of $\prod_{i=1}^{L} Pr(y_i - f(\mathbf{x}_i))$, we have

$$\sum_{i=1}^{L} log(Pr(y_i - f(\mathbf{x}_i))), \tag{6}$$

maximizing (6) is equivalent to minimizing $-\sum_{i=1}^{L} log(Pr(y_i - f(\mathbf{x}_i)))$ Let $c(\mathbf{x}_i, y_i, f(\mathbf{x}_i))$ denotes the ith cost function, which is defined as

$$c(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) = -log(P(y_i - f(\mathbf{x}_i))). \tag{7}$$

Thus (5) can be rewritten as

$$\min_f \quad \sum_{i=1}^{L} c(\mathbf{x}_i, y_i, f(\mathbf{x}_i)), \tag{8}$$

In ϵ-SVR, in order to provide more flexibility to precision control, Vapnik's ϵ-insensitive function, as shown in (9), is applied to (7).

$$|u|_\epsilon = \begin{cases} 0 & if \quad |u| < \epsilon \\ |u| - \epsilon & otherwise \end{cases} \tag{9}$$

Thus the final form of cost function in ϵ-SVR can be written as

$$\tilde{c}(y_i, \mathbf{x}_i, f(\mathbf{x}_i)) = -log(Pr(|y_i - f(\mathbf{x}_i)|_\epsilon)), \tag{10}$$

Consider the cost function term in (2),

$$\sum_{i=1}^{L} R(\xi_i) + R(\hat{\xi}_i) = \sum_{i=1}^{L} \tilde{c}(y_i, \mathbf{x}_i, f(\mathbf{x}_i)) = \sum_{i=1}^{L} -log(Pr(|y_i - f(\mathbf{x}_i)|_\epsilon)) \tag{11}$$

the cost function term is determined according to noise distribution.

C. Lagrange Duality

According to Lagrange Theorem, the constraint optimization problem (2) can be transformed to Lagrangian dual form by combining the original optimization function with inequality constraints, the combination coefficient is called Lagrange multiplier. The Lagrange function is given by.

$$\begin{aligned} & L(\mathbf{w}, b, \xi_i, \hat{\xi}_i, \alpha_i, \hat{\alpha}_i, \eta_i, \hat{\eta}_i) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{L}(R(\xi_i) + R(\hat{\xi}_i)) - \sum_{i=1}^{L}(\eta_i\xi_i + \hat{\eta}_i\hat{\xi}_i) \\ & + \sum_{i=1}^{L} \alpha_i(y_i - \mathbf{w}^T\Phi(\mathbf{x}_i) - b - \epsilon - \xi_i) + \sum_{i=1}^{L} \hat{\alpha}_i(\mathbf{w}^T\Phi(\mathbf{x}_i) + b - y_i - \epsilon - \hat{\xi}_i) \\ & s.t. \begin{cases} \eta_i, \hat{\eta}_i, \alpha_i, \hat{\alpha}_i \geq 0, i = 1, 2, \cdots L \\ \xi_i, \hat{\xi}_i \geq 0, i = 1, 2, \cdots L \end{cases} \end{aligned} \tag{12}$$

where η_i, $\hat{\eta}_i$, α_i, $\hat{\alpha}_i$ are Lagrange multipliers.

The sufficient and necessary conditions such that a solution $\mathbf{w}$ of the constrained optimization problem in (2) satisfies, are called Karush-Kuhn-Tucker (KKT) conditions. Here we elaborate a little further about how the dual objective problem is derived from KKT conditions.

Assume a constraint optimization problem is given by

$$\begin{aligned} & \min_{\mathbf{w}} \quad f(\mathbf{w}) \\ & s.t. \quad c_i(\mathbf{w}) \leq 0, i = 1, 2, \ldots L, \end{aligned} \tag{13}$$

its Lagrange function is given by

$$L(\mathbf{w}, a_i) = f(\mathbf{w}) + \sum_{i=1}^{L} a_i c_i(\mathbf{w}), \tag{14}$$

where a_i denote Lagrange multipliers. Based on Theorem 6.21 in [26], For a variable set $[\bar{\mathbf{w}}, \bar{a}_i]$, $\bar{\mathbf{w}}$ is the solution to (13) only when the following inequalities are satisfied

$$L(\mathbf{w}, \bar{a}_i) \geq L(\bar{\mathbf{w}}, \bar{a}_i) \geq L(\bar{\mathbf{w}}, a_i) \tag{15}$$

This inequalities yield KKT conditions (see Theorem 6.26 [26]), which are

$$\partial_{\mathbf{w}} L(\bar{\mathbf{w}}, a_i) = \partial_{\mathbf{w}} f(\bar{\mathbf{w}}) + \sum_{i=1}^{L} a_i \partial_{\mathbf{w}} c_i(\bar{\mathbf{w}}) = 0, \tag{16}$$

$$\partial_{a_i} L(\bar{\mathbf{w}}, \bar{a}_i) = c_i(\bar{\mathbf{w}}) \leq 0, i = 1, 2, \ldots L \tag{17}$$

$$\bar{a}_i c_i(\bar{\mathbf{w}}) = 0, i = 1, 2, \ldots L \tag{18}$$

In order to satisfy the first inequality in (15), (16) has to hold, applying (16) to (12), which are

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{l} (\alpha_i - \hat{\alpha}_i) \Phi(\mathbf{x}_i) = 0 \tag{19}$$

$$\frac{\partial L}{\partial \xi_i} = C_i R^{'}(\xi_i) - \eta_i - \alpha_i = 0, i = 1, 2, \cdots L \tag{20}$$

$$\frac{\partial L}{\partial \hat{\xi}_i} = C_i R^{'}(\hat{\xi}_i) - \hat{\eta}_i - \hat{\alpha}_i = 0, i = 1, 2, \cdots L \tag{21}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{l} (\alpha_i - \hat{\alpha}_i) = 0 \tag{22}$$

Then by substituting (19)-(22) to (12), (12) can be rewritten as :

$$\theta(\alpha_i,\hat{\alpha}_i) = \frac{1}{2}\sum_{i=1}^{L}\sum_{j=1}^{L}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)\Phi^T(\mathbf{x}_j)\Phi(\mathbf{x}_i) + C\sum_{i=1}^{L}[R(\xi_i)+R(\hat{\xi}_i)] - \sum_{i=1}^{L}[(CR^{'}(\xi_i)-\alpha_i)\xi_i$$
$$+(CR^{'}(\hat{\xi}_i)-\hat{\alpha}_i)\hat{\xi}_i] + \sum_{i=1}^{L}\alpha_i(y_i - \sum_{j=1}^{L}(\alpha_j-\hat{\alpha}_j)\Phi^T(\mathbf{x}_j)\Phi(\mathbf{x}_i) - b - \epsilon - \xi_i)+$$
$$\sum_{i=1}^{L}\hat{\alpha}_i(\sum_{j=1}^{L}(\alpha_j-\hat{\alpha}_j)\Phi^T(\mathbf{x}_j)\Phi(\mathbf{x}_i) + b - y_i - \epsilon - \hat{\xi}_i) \tag{23}$$

notice in (22), $\sum_{i=1}^{L}(\alpha_i-\hat{\alpha}_i)=0$, define $\tilde{R}(u)=R(u)-uR^{'}(u)$, (23) can be further simplified to

$$\theta(\alpha_i,\hat{\alpha}_i) = -\frac{1}{2}\sum_{i=1}^{L}\sum_{i=1}^{L}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)\Phi(\mathbf{x}_j)^T\Phi(\mathbf{x}_i) + C\sum_{i=1}^{L}[(\tilde{R}(\xi_i)+\tilde{R}(\hat{\xi}_i)]$$
$$+\sum_{i=1}^{L}[(\alpha_i-\hat{\alpha}_i)y_i-(\alpha_i+\hat{\alpha}_i)\epsilon]$$
$$s.t.\begin{cases}\sum_{i=1}^{l}(\alpha_i-\hat{\alpha}_i)=0\\ 0<\alpha<C\tilde{R}^{'}(\alpha)\\ 0<\hat{\alpha}<C\tilde{R}^{'}(\hat{\alpha})\end{cases} \tag{24}$$

In order to satisfy the second inequality in (15), (17) and (18) have to hold. Condition (17) is satisfied when the maximum of $L(\bar{\mathbf{w}},a_i)$ is found, notice $\theta(\alpha_i,\hat{\alpha}_i)$ is equivalent to $L(\bar{\mathbf{w}},a_i)$ in (15), thus yielding the dual optimization problem, which is given by

$$\max_{\alpha_i,\hat{\alpha}_i}\quad \theta(\alpha_i,\hat{\alpha}_i) = -\frac{1}{2}\sum_{i=1}^{L}\sum_{j=1}^{L}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)\Phi^T(\mathbf{x}_j)\Phi(\mathbf{x}_i) + C\sum_{i=1}^{L}(\tilde{R}(\xi_i)+\tilde{R}(\hat{\xi}_i))$$
$$+\sum_{i=i}^{L}[(\alpha_i-\hat{\alpha}_i)y_i-(\alpha_i+\hat{\alpha}_i)\epsilon] \tag{25}$$

Define $\mathbf{a} = [\alpha_1, \alpha_2, \ldots, \alpha_L]^T$, $\hat{\mathbf{a}} = [\hat{\alpha}_1, \hat{\alpha}_2, \ldots, \hat{\alpha}_L]^T$, $\mathbf{y} = [y_1, y_2, \ldots y_L]^T$, $\mathbf{e} = [1, 1, \ldots, 1]^T \in \mathbb{R}^L$, $\mathbf{e}_i$ denotes the vector that only ith component is 1 while the rest are all 0, $\mathbf{R}_\xi = [\tilde{R}(\xi_1), \tilde{R}(\xi_2), \ldots, \tilde{R}(\xi_L)]^T$, $\mathbf{R}_{\hat{\xi}} = [\tilde{R}(\hat{\xi}_1), \tilde{R}(\hat{\xi}_2), \ldots, \tilde{R}(\hat{\xi}_L)]^T$, $[\mathbf{K}]_{ij} = \Phi(\mathbf{x}_j)^T \Phi(\mathbf{x}_i)$ denotes a component of data kernel matrix at ith row and jth column. An alternative vector form of (25) can be written as

$$\max_{\mathbf{a}, \hat{\mathbf{a}}} \theta(\mathbf{a}, \hat{\mathbf{a}}) = -\frac{1}{2}(\mathbf{a} - \hat{\mathbf{a}})^T \mathbf{K} (\mathbf{a} - \hat{\mathbf{a}}) + (\mathbf{y} - \epsilon \mathbf{e})^T \mathbf{a} + (-\mathbf{y} - \epsilon \mathbf{e})^T \hat{\mathbf{a}} + C \mathbf{e}^T (\mathbf{R}_\xi + \mathbf{R}_{\hat{\xi}}), \quad (26)$$

We define the following $2L$ vectors $\mathbf{a}^{(*)}$= $[\begin{smallmatrix} \mathbf{a} \\ \hat{\mathbf{a}} \end{smallmatrix}]$, $\mathbf{v} \in \mathbb{R}^{2L}$,

$$[\mathbf{v}]_i = \begin{cases} 1 & i = 1, \ldots, l \\ -1 & i = l + 1, \ldots, 2l \end{cases} \quad (27)$$

(26) can also be reformulate as

$$\max_{\mathbf{a}^*} \quad \theta(\mathbf{a}^*) = -\frac{1}{2}(\mathbf{a}^*)^T \left[\begin{smallmatrix} \mathbf{K} & -\mathbf{K} \\ -\mathbf{K} & \mathbf{K} \end{smallmatrix}\right] \mathbf{a}^{(*)} + [(\mathbf{y} - \epsilon)^T, (-\mathbf{y} - \epsilon)^T] \mathbf{a}^{(*)} + C \mathbf{e}^T (\mathbf{R}_\xi + \mathbf{R}_{\hat{\xi}}), \quad (28)$$

Condition in (18) is called KKT complementary condition, the value of $\sum_{i=1}^L \bar{a}_i c_i(\bar{\mathbf{w}})$ can be used to monitor how close the solution is to the global optimum. Thus it can be used as a stopping criterion. In the constraint optimization problem of (12), the KKT complementary conditions are given by

$$\begin{cases} \alpha_i (y_i - \mathbf{w}^T \Phi(\mathbf{x}_i) - b - \epsilon - \xi_i) = 0, i = 1, 2 \cdots L \\ \hat{\alpha}_i (\mathbf{w}^T \Phi(\mathbf{x}_i) + b - y_i - \epsilon - \hat{\xi}_i) = 0, i = 1, 2 \cdots L \end{cases} \quad (29)$$

Based on the definitions of slack variables in (3) an (4), only when there is a outlier exists, ξ_i

or $\hat{\xi}_i$ can be non zero, that is

$$\xi_i(\hat{\xi}_i) = |\mathbf{y}_i - \mathbf{w}^T\Phi(\mathbf{x}_i)|_\epsilon, \tag{30}$$

because the distance of the estimation $\mathbf{w}^T\Phi(\mathbf{x}_i) + b$ and the observation $\mathbf{y}_i$ can only exceeds the ϵ-tube in one direction, as shown in Fig. 1. Therefore at most one of $(y_i - \mathbf{w}^T\Phi(\mathbf{x}_i) - b - \epsilon - \xi_i)$ and $(\mathbf{w}^T\Phi(\mathbf{x}_i) + b - y_i - \epsilon - \hat{\xi}_i)$ can be zero. In order to satisfy the KKT complementary conditions in (29), at least one of α_i and $\hat{\alpha}_i$ need to be zero, that is $\alpha_i\hat{\alpha}_i = 0$.

III. Dual Channel Complex Support Vector detection for Large MIMO system

A. System Model

Consider a uncoded complex large MIMO uplink spatial multiplexing (SM) system with N_t users, where each is equipped with transmit antenna. The number of receive antennas at the Base Station (BS) is N_r, $N_r \geq N_t$. Typically large MIMO systems have hundreds of antennas at the BS, as shown in Fig 2.

Bit sequences, which are modulated to complex symbols, are transmitted by the users over a flat fading channel. The discrete time model of the system is given by:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{31}$$

where $\mathbf{y} \in \mathbb{C}^{N_r \times 1}$ is the received symbol vector, $\mathrm{s} \in \mathbb{C}^{N_t}$ is the transmitted symbol vector, with components that are mutually independent and taken from a finite signal constellation alphabet $\mathbb{O}$ (e.g. BPSK, 4-QAM, 16-QAM, 64-QAM), $|\mathbb{O}| = M$. The transmitted symbol vectors $\mathrm{s} \in \mathbb{O}^{N_t}$, satisfy $\mathbb{E}[\mathrm{ss}^H] = \mathbf{I}_{N_t}E_s$, where E_s denotes the symbol average energy, $\mathbb{E}[\cdot]$ denotes the expectation operation, $\mathbf{I}_{N_t}$ denotes identity matrix of size $N_r \times N_t$. Furthermore

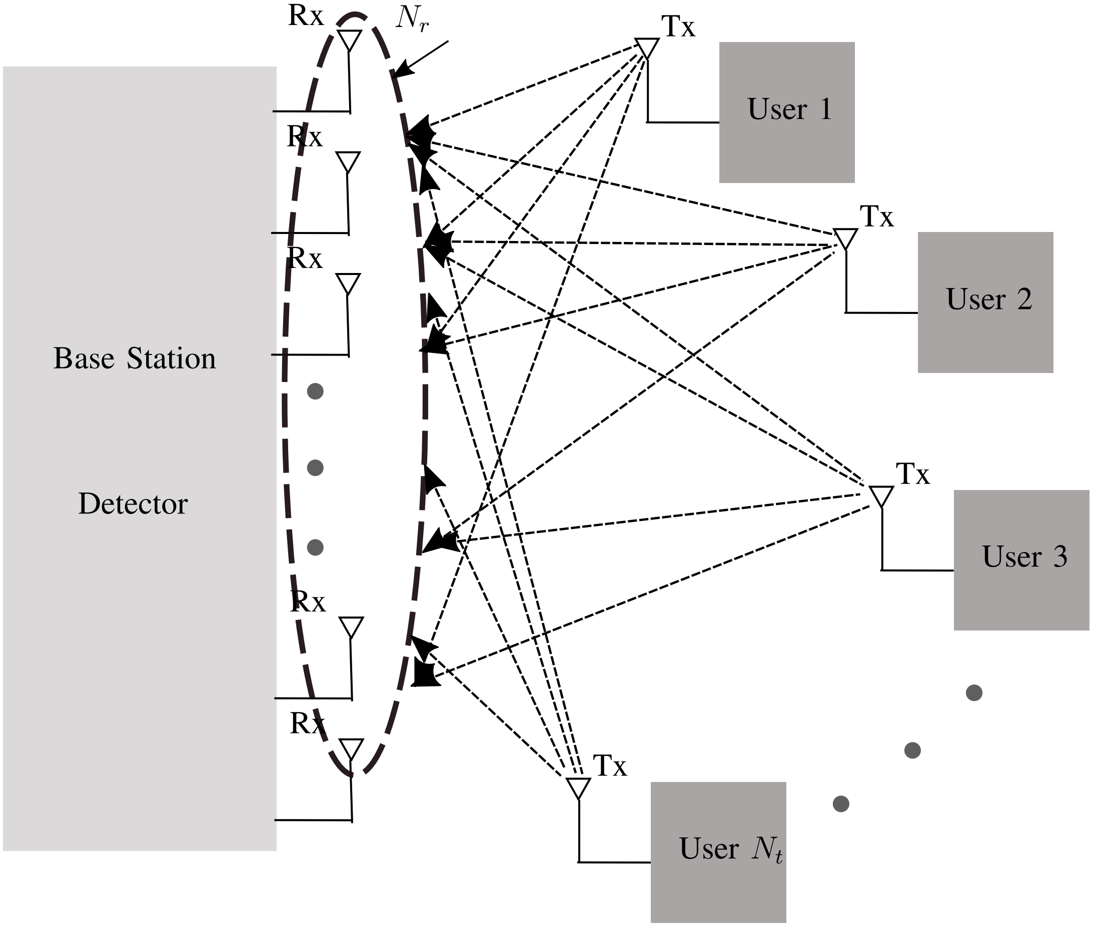

Fig. 2. Large MIMO uplink system

$\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ denotes the Rayleigh fading channel propagation matrix, each component is independent identically distributed (i.i.d) circularly symmetric complex Gaussian random variable with zero mean and unit variance. Finally, $\mathbf{n} \in \mathbb{C}^{N_r}$ is the additive white Gaussian noise (AWGN) vector with zero mean components and $\mathbb{E}[\mathbf{n}\mathbf{n}^H] = \mathbf{I}_{N_r} N_0$, where N_0 denotes the noise power spectrum density, and hence $\frac{E_s}{N_0}$ is the signal to noise ratio (SNR).

The task of a MIMO detector is to estimate the transmit symbol vector $\mathbf{s}$, based on the knowledge of receive symbol vector $\mathbf{y}$ and channel matrix $\mathbf{H}$.

B. Complex Regression Model

Based on the discrete time model of the large MIMO uplink system of (31), our regression model is set such that the training data samples at the detector are $(\mathbf{h}_1, y_1)(\mathbf{h}_2, y_2), \ldots, (\mathbf{h}_{N_r}, y_{N_r})$, where $\mathbf{h}_i$ denotes ith row of matrix $\mathbf{H}$. This yields a regression task without offset b in (1) :

$$y_i = f_{true}(\mathbf{h}_i) + n_i, i = 1, 2 \ldots L \tag{32}$$

$$f_{true}(\mathbf{h}_i) = \mathbf{h}_i \cdot \mathbf{s}, i = 1, 2, \ldots, L \tag{33}$$

$$\tag{34}$$

where $f_{true}()$ denotes the underlying true regression function, n_i denotes discrete sample of additive noise. In this regression problem, received symbols y_i are the output data, $\mathbf{h}_i$ are the input data samples, transmitted symbol vector $\mathbf{s}$ contains the regression coefficients. We employ complex support vector regression (CSVR) without offset term b, in order to deal with the complex large MIMO systems. As shown in section II, As shown in (16) the first KKT condition is satisfied by finding the saddle point $\bar{\mathbf{w}}$ by taking paritial derivatives of $L(\mathbf{w}, a_i)$ with respect to $\mathbf{w}$. Mathematical results of Wirtinger's calculus in Reproducing Kernel Hilbert Space (RKHS) are employed to calculate partial derivatives of Lagrangian function of CSVR in complex manner [39]. First we generalize our regression model by complex RKHS, Define a real RKHS $\mathcal{H}$, $<, >_{\mathcal{H}}$ denotes the corresponding inner product in $\mathcal{H}$. define complex RKHS $\mathbb{H} = \{f = f^r + \imath f^i, f^r, f^i \in \mathcal{H}\}$, $<, >_{\mathbb{H}}$ denotes the corresponding inner products in $\mathbb{H}$. Assume $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$, μ, $\chi \in \mathbb{C}$, complex Hilbert space has the following properties

Property 1. $< \mathbf{x}, \mathbf{y} >_{\mathbb{H}} = (< \mathbf{y}, \mathbf{x} >_{\mathbb{H}})^*$

Property 2. $< \mu\mathbf{x} + \chi\mathbf{y}, \mathbf{z} >_{\mathbb{H}} = \mu < \mathbf{x}, \mathbf{z} >_{\mathbb{H}} + \chi < \mathbf{y}, \mathbf{z} >_{\mathbb{H}}$

Property 3. $< \mathbf{z}, \mu\mathbf{x} + \chi\mathbf{y} >_{\mathbb{H}} = \mu^* < \mathbf{z}, \mathbf{x} >_{\mathbb{H}} + \chi^* < \mathbf{z}, \mathbf{y} >_{\mathbb{H}}$

Lemma 1. *Assume* $\mathbf{a}$*,* $\mathbf{b} \in \mathbb{R}^V$*, define a real RKHS* $\mathcal{H}$ *that satisfy* $< \mathbf{a}, \mathbf{b} >_{\mathcal{H}} = \mathbf{a}^T\mathbf{b}$*. For the corresponding complex RKHS* $\mathbb{H} = \{f = f^r + \imath f^i, f^r, f^i \in \mathcal{H}\}$*,* $< \mathbf{m}, \mathbf{l}^* >_{\mathbb{H}} = \mathbf{m} \cdot \mathbf{l}$

Proof. $\mathbf{g}$,$\mathbf{h} \in \mathbb{C}^V$, and $\mathbf{g} = \Re(\mathbf{g}) + \imath\Im(\mathbf{g})$, $\mathbf{h} = \Re(\mathbf{h}) + \imath\Im(\mathbf{h})$. From Property 1 and Property 3,

$$< \mathbf{g}, \mathbf{h} >_{\mathbb{H}} = < \Re(g) + \imath\Im(g), \Re(h) + \imath\Im(h) >_{\mathbb{H}} \tag{35}$$

According to Property 2, (35) can be rewritten as

$$< \mathbf{g}, \mathbf{h} >_{\mathbb{H}} = < \Re(\mathbf{g}), \Re(\mathbf{h}) + \imath\Im(\mathbf{h}) >_{\mathbb{H}} + \imath < \Im(\mathbf{g}), \Re(\mathbf{h}) + \imath\Im(\mathbf{h}) >_{\mathbb{H}}, \tag{36}$$

According to Property 3

$$\begin{aligned} &< \mathbf{g}, \mathbf{h} >_{\mathbb{H}} = < \Re(\mathbf{g}), \Re(\mathbf{h}) >_{\mathbb{H}} - \imath < \Re(\mathbf{g}), \Im(\mathbf{h}) >_{\mathbb{H}} + \imath(< \Im(\mathbf{g}), \Re(\mathbf{h}) >_{\mathbb{H}} - \imath < \Im(\mathbf{g}), \\ &\Im(\mathbf{h}) >_{\mathbb{H}}) = < \Re(\mathbf{g}), \Re(\mathbf{h}) >_{\mathbb{H}} + < \Im(\mathbf{g}), \Im(\mathbf{h}) >_{\mathbb{H}} + \imath[< \Im(\mathbf{g}), \Re(\mathbf{h}) >_{\mathbb{H}} - < \Re(\mathbf{g}), \\ &\Im(\mathbf{h}) >_{\mathbb{H}}] \end{aligned} \tag{37}$$

Because $< \mathbf{f}_1, \mathbf{f}_2 >_{\mathbb{H}} = < \mathbf{f}_1, \mathbf{f}_2 >_{\mathcal{H}} = \mathbf{f}_1^T \cdot \mathbf{f}_2$ if $\mathbf{f}_1, \mathbf{f}_2 \in \mathcal{H}$, thus (37) can be rewritten as

$$\begin{aligned} &< \mathbf{g}, \mathbf{h} >_{\mathbb{H}} = < \Re(\mathbf{g}), \Re(\mathbf{h}) >_{\mathcal{H}} + < \Im(\mathbf{g}), \Im(\mathbf{h}) >_{\mathcal{H}} + \imath[< \Im(\mathbf{g}), \Re(\mathbf{h}) >_{\mathcal{H}} - < \Re(\mathbf{g}), \Im(\mathbf{h}) >_{\mathcal{H}}] = \\ &\Re(\mathbf{g})^T \cdot \Re(\mathbf{h}) + \Im(\mathbf{g})^T \cdot \Im(\mathbf{h}) + \imath[\Im(\mathbf{g})^T \cdot \Re(\mathbf{h}) - \Re(\mathbf{g})^T \cdot \Im(\mathbf{h})] = \mathbf{g} \cdot \mathbf{h}^* \end{aligned} \tag{38}$$

Therefore $< \mathbf{m}, \mathbf{l} >_{\mathbb{H}} = \mathbf{m} \cdot \mathbf{l}$. □

Therefore we have $< \mathbf{h}_i, \mathbf{s}^* >_{\mathbb{H}} = \mathbf{h}_i \cdot \mathbf{s}$, For sake of simplicity we define $\mathbf{s}^* = \mathbf{w}$. Based on the same principle in the real case presented in (II), the general regularized risk function of large MIMO detection in complex RKHS $\mathbb{H}$ defined by $< \mathbf{m}, \mathbf{l} >_{\mathbb{H}} = \mathbf{m}^T \mathbf{l}$ can be formulated as

$$\min_{\mathbf{w},\xi_i^r,\hat{\xi}_i^r,\xi_i^i,\hat{\xi}_i^i} \quad \frac{1}{2}||\mathbf{w}||_{\mathbb{H}}^2 + C\sum_{i=1}^{N_r}[(R(\xi_i^r) + R(\hat{\xi}_i^r) + R(\xi_i^i) + R(\hat{\xi}_i^i))]$$

$$s.t. \begin{cases} \Re(y_i - < \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}}) \leq \epsilon + \xi_i^r, i = 1, 2, \ldots, N_r \\ \Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - y_i) \leq \epsilon + \hat{\xi}_i^r, i = 1, 2, \ldots, N_r \\ \Im(y_i - < \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}}) \leq \epsilon + \xi_i^i, i = 1, 2, \ldots, N_r \\ \Im(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - y_i) \leq \epsilon + \hat{\xi}_i^i, i = 1, 2, \ldots, N_r \\ \xi_i^r, \hat{\xi}_i^r, \xi_i^i, \hat{\xi}_i^i \geq 0, i = 1, 2, \ldots, N_r \end{cases} \tag{39}$$

Where $\Re(\cdot)$ and $\Im(\cdot)$ denote real and imaginary part of a complex variable and inequality restrictions are set to real and imaginary part of the regression function separately. Let $\mathbf{K} = \mathbf{H}\mathbf{H}^H$ denote the kernel function, $\mathbf{K} = \Re(\mathbf{K}) + \imath\Im(\mathbf{K})$, where $\Re(\mathbf{K})$ and $\Im(\mathbf{K})$ denote matrices of corresponding real and imaginary parts. Similar to the Lagrange duality rational in section II-C, the Lagrange function associated with (39) is

$$\begin{aligned} L = {} & \frac{1}{2}||\mathbf{w}||_{\mathbb{H}}^2 + C\sum_{i=1}^{N_r}[(R(\xi_i^r) + R(\hat{\xi}_i^r) + R(\xi_i^i) + R(\hat{\xi}_i^i))] - \sum_{i=1}^{N_r}(\eta_i\xi_i^r + \hat{\eta}_i\hat{\xi}_i^r + \tau_i\xi_i^i \\ & + \hat{\tau}_i\hat{\xi}_i^i) + \sum_{i=1}^{N_r}\alpha_i(\Re(y_i) - \Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}}) - \epsilon - \xi_i^r) + \sum_{i=1}^{N_r}\hat{\alpha}_i(\Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}}) - \Re(y_i) - \epsilon - \hat{\xi}_i^r) \\ & + \sum_{i=1}^{N_r}\beta_i(\Im(y_i) - \Im(< \mathbf{h}_k, \mathbf{w} >_{\mathbb{H}}) - \epsilon - \xi_i^i) + \sum_{i=1}^{N_r}\hat{\beta}_i(\Im(< \mathbf{h}_k, \mathbf{w} >_{\mathbb{H}}) - \Im(y_i) - \epsilon - \hat{\xi}_i^i) \\ & s.t. \begin{cases} \eta_i, \hat{\eta}_i, \tau_i, \hat{\tau}_i, \alpha_i, \hat{\alpha}_i, \beta_i, \hat{\beta}_i \geq 0, i = 1, 2, \ldots, N_r \\ \xi_i^r, \hat{\xi}_i^r, \xi_i^i, \hat{\xi}_i^i \geq 0, i = 1, 2, \ldots, N_r \end{cases} \end{aligned} \tag{40}$$

where $\eta_i, \hat{\eta}_i, \tau_i, \hat{\tau}_i, \alpha_i, \hat{\alpha}_i, \beta_i, \hat{\beta}_i$ are Lagrange multipliers. In order to calculate the partial derivations of L with respect to $\mathbf{w}$ which is defined in complex domain, the Wirtinger's calculus is exploited [39] , Therefore we have

$$\begin{cases} \frac{\partial L}{\partial \mathbf{w}^*} = \frac{1}{2}\mathbf{w} - \frac{1}{2}\sum_{i=1}^{N_r}\alpha_i\mathbf{h}_i + \frac{1}{2}\sum_{i=1}^{N_r}\hat{\alpha}_i\mathbf{h}_i + \frac{\imath}{2}(\sum_{i=1}^{N_r}\beta_i\mathbf{h}_i - \sum_{i=1}^{N_r}\hat{\beta}_i\mathbf{h}_i) = 0 \\ \Rightarrow \mathbf{w} = \sum_{i=1}^{N_r}(\alpha_i - \hat{\alpha}_i)\mathbf{h}_i - \imath\sum_{i=1}^{N_r}(\beta_i - \hat{\beta}_i)\mathbf{h}_i \\ \frac{\partial L}{\partial \xi_i^r} = CR^{'}(\xi_i^r) - \eta_i - \alpha_i = 0 \Rightarrow \eta_i = CR^{'}(\xi_i^r) - \alpha_i \\ \frac{\partial L}{\partial \hat{\xi}_i^r} = CR^{'}(\hat{\xi}_i^r) - \hat{\eta}_i - \hat{\alpha}_i = 0 \Rightarrow \hat{\eta}_i = CR^{'}(\hat{\xi}_i^r) - \hat{\alpha}_i \\ \frac{\partial L}{\partial \xi_i^i} = CR^{'}(\xi_i^i) - \tau_i - \beta_i = 0 \Rightarrow \tau_i = CR^{'}(\xi_i^i) - \beta_i \\ \frac{\partial L}{\partial \hat{\xi}_i^i} = CR^{'}(\hat{\xi}_i^i) - \hat{\tau}_i - \hat{\beta}_i = 0 \Rightarrow \hat{\tau}_i = CR^{'}(\hat{\xi}_i^i) - \hat{\beta}_i \end{cases} \tag{41}$$

Based on (41), we have

$$\begin{aligned} &< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} = \sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j) < \mathbf{h}_i, \mathbf{h}_j >_{\mathbb{H}} + i\sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j) < \mathbf{h}_i, \mathbf{h}_j >_{\mathbb{H}} \\ &= \sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j)\mathbf{h}_i\mathbf{h}_j^H + i\sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\mathbf{h}_i\mathbf{h}_j^H \\ &= \sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j)\Re(\mathbf{K})_{ij} - \sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\Im(\mathbf{K})_{ij} + i(\sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j)\Im(\mathbf{K})_{ij} + \\ &\sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\Re(\mathbf{K})_{ij}), \end{aligned} \tag{42}$$

$$\begin{aligned} ||\mathbf{w}||_{\mathbb{H}}^2 &= \sum_{i,j=1}^{N_r}(\alpha_i - \hat{\alpha}_i)(\alpha_i - \hat{\alpha}_i)\mathbf{h}_i\mathbf{h}_j^H + \sum_{i,j=1}^{N_r}(\beta_i - \hat{\beta}_i)(\beta_i - \hat{\beta}_i)\mathbf{h}_i\mathbf{h}_j^H \\ &+ i(\sum_{i,j=1}^{N_r}(\alpha_i - \hat{\alpha}_i)(\beta_j - \hat{\beta}_j)\mathbf{h}_i\mathbf{h}_j^H - \sum_{i,j=1}^{N_r}(\alpha_i - \hat{\alpha}_i)(\beta_j - \hat{\beta}_j)\mathbf{h}_j\mathbf{h}_i^H) \end{aligned} \tag{43}$$

Because $\mathbf{K}$ is Hermitian, thus $\mathbf{K}_{ij} = \mathbf{K}_{ji}^*$. If we have r_i, $r_j \in \mathbb{R}$, then

$$\sum_{i,j}^{L} r_i r_j \Im(\mathbf{K})_{ij} = -\sum_{i,j}^{L} r_i r_j \Im(\mathbf{K})_{ji} = -\sum_{i,j}^{L} r_i r_j \Im(\mathbf{K})_{ij}, \tag{44}$$

Therefore

$$\sum_{i,j}^{l} r_i r_j \Im(\mathbf{K})_{ij} = 0, \tag{45}$$

Based on (45), (43) can be changed to

$$||\mathbf{w}||_{\mathbb{H}}^2 = \sum_{i,j=1}^{N_r} (\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)\Re(\mathbf{K})_{ij} + \sum_{i,j=1}^{N_r} (\beta_i - \hat{\beta}_i)(\beta_j - \hat{\beta}_j)\Re(\mathbf{K})_{ij} - 2\sum_{i,j=1}^{N_r} (\alpha_i - \hat{\alpha}_i)(\beta_j - \hat{\beta}_j)\Im(\mathbf{K})_{ij}. \tag{46}$$

Apply (41), (42), (45) and (46) to (40), the final form of Lagrange duality can be obtained

$$\begin{aligned}
&L = \frac{1}{2}[\sum_{i,j=1}^{N_r} (\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)\Re(\mathbf{K})_{ij} + \sum_{i,j=1}^{N_r} (\beta_i - \hat{\beta}_i)(\beta_j - \hat{\beta}_j)\Re(\mathbf{K})_{ij} - 2\sum_{i,j=1}^{N_r} (\alpha_i - \hat{\alpha}_i)(\beta_j - \hat{\beta}_j)\Im(\mathbf{K})_{ij}] + \\
&C\sum_{i=1}^{N_r}[R(\xi_i^r) + R(\hat{\xi}_i^r) + R(\xi_i^i) + R(\hat{\xi}_i^i)] - \sum_{i=1}^{N_r}((CR^{'}(\xi_i^r) - \alpha_i)\xi_i^r + (CR^{'}(\hat{\xi}_i^r) - \hat{\alpha}_i)\hat{\xi}_i^r + (CR^{'} - \beta_i)\xi_i^i \\
&+(CR^{'}(\hat{\xi}_i) - \hat{\beta}_i)\hat{\xi}_i^i) + \sum_{i=1}^{N_r}\alpha_i(\Re(y_i) - (\sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j)\Re(\mathbf{K})_{ij} - \sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\Im(\mathbf{K})_{ij}) - \epsilon - \xi_i^r) + \sum_{i=1}^{N_r}\hat{\alpha}_i \\
&((\sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j)\Re(\mathbf{K})_{ij} - \sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\Im(\mathbf{K})_{ij}) - \Re(y_i) - \epsilon - \hat{\xi}_i^r) + \sum_{i=1}^{N_r}\beta_i(\Im(y_i) - (\sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j) \\
&\Im(\mathbf{K})_{ij} + \sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\Re(\mathbf{K})_{ij}) - \epsilon - \xi_i^i) + \sum_{i=1}^{N_r}\hat{\beta}_i((\sum_{j=1}^{N_r}(\alpha_j - \hat{\alpha}_j)\Im(\mathbf{K})_{ij} + \sum_{j=1}^{N_r}(\beta_j - \hat{\beta}_j)\Re(\mathbf{K})_{ij}) - \Im(y_i) - \\
&\epsilon - \hat{\xi}_i^i)
\end{aligned} \tag{47}$$

$$\theta = -\frac{1}{2}[\sum_{i,j=1}^{N_r}(\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)\Re(\mathbf{K})_{ij} + \sum_{i,j=1}^{N_r}(\beta_i - \hat{\beta}_i)(\beta_j - \hat{\beta}_j)\Re(\mathbf{K})_{ij}] + C\sum_{i=1}^{N_r}[R(\xi_i^r) - \xi_i^r R^{'}(\xi_i^r)$$

$$+R(\hat{\xi}_i^r) - \hat{\xi}_i^r R^{'}(\hat{\xi}_i^r) + R(\xi_i^i) - \xi_i R^{'}(\xi_i^i) + R(\hat{\xi}_i^i) - \hat{\xi}_i^i R^{'}(\hat{\xi}_i^i)] + \sum_{i=1}^{N_r}[\Re(y_i)(\alpha_i - \hat{\alpha}_i) + \Im(y_i)(\beta_i - \hat{\beta}_i)] -$$

$$\epsilon\sum_{i=1}^{N_r}(\alpha_i + \hat{\alpha}_i + \beta_i + \hat{\beta}_i) \tag{48}$$

define $\tilde{R}(u) = R(u) - uR^{'}(u)$. Similar to (25), The dual optimization problem of complex MIMO system is

$$\begin{aligned}
\max_{\alpha_i,\hat{\alpha}_i,\beta_i,\hat{\beta}_i} \quad & \theta = -\frac{1}{2}[\sum_{i,j}^{N_r}(\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)\mathbf{K}_{ij}^r + \sum_{i,j}^{N_r}(\beta_i - \hat{\beta}_i)(\beta_j - \hat{\beta}_j)\mathbf{K}_{ij}^r] \\
& -\sum_{i}^{N_r}(\alpha_i + \hat{\alpha}_i + \beta + \hat{\beta}_i)\epsilon + [\sum_{i=1}^{N_r}(\alpha_i - \hat{\alpha}_i)Re(y_i) + \sum_{i=1}^{N_r}(\beta_i - \hat{\beta}_i)Im(y_i)] \\
& +C\sum_{i}^{N_r}(\tilde{R}(\xi_i^r) + \tilde{R}(\hat{\xi}_i^r) + \tilde{R}(\xi_i^i) + \tilde{R}(\hat{\xi}_i^i)) \\
& \begin{cases}
0 \le \alpha_i(\hat{\alpha}_i) \le C\tilde{R}(\xi_i^r)(\tilde{R}(\hat{\xi}_i^r)), i = 1, 2, \ldots, L \\
0 \le \beta_i(\hat{\beta}_i) \le C\tilde{R}(\xi_i^i)(\tilde{R}(\hat{\xi}_i^i)), i = 1, 2, \ldots, L \\
\xi_i^r(\hat{\xi}_i^r) \ge 0, i = 1, 2, \ldots, L \\
\xi_i^i(\hat{\xi}_i^i) \ge 0, i = 1, 2, \ldots, L
\end{cases}
\end{aligned} \tag{49}$$

Notice in (49), there is no correlation term between $\alpha_i, \hat{\alpha}_i$ and $\beta_i, \hat{\beta}_i$, therefore, (49) can be divided into two independent regression tasks,

$$\max_{\alpha_i,\hat{\alpha}_i} \quad \theta^r = -\frac{1}{2}\sum_{i,j}^{N_r}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)\Re(\mathbf{K})_{ij} - \sum_{i=1}^{N_r}(\alpha_i+\hat{\alpha}_i)\epsilon + \sum_{i=1}^{N_r}(\alpha_i-\hat{\alpha}_i)Re(y_i) + C\sum_{i=1}^{N_r}(\tilde{R}(\xi_i^r)$$

$$+\tilde{R}(\hat{\xi}_i^r))$$

$$\begin{cases} 0 \leq \alpha_i(\hat{\alpha}_i) \leq C\tilde{R}(\xi_i^r)(\tilde{R}(\hat{\xi}_i^r)), i=1,2,\ldots,L \\ \xi_i^r(\hat{\xi}_i^r) \geq 0, i=1,2,\ldots,L \end{cases} \tag{50}$$

$$\max_{\beta_i,\hat{\beta}_i} \quad \theta^i = -\frac{1}{2}\sum_{i,j}^{N_r}(\beta_i-\hat{\beta}_i)(\beta_j-\hat{\beta}_j)\Re(\mathbf{K})_{ij} - \sum_{i=1}^{N_r}(\beta_i+\hat{\beta}_i)\epsilon + \sum_{i=1}^{N_r}(\beta_i-\hat{\beta}_i)Im(y_i) + C\sum_{i=1}^{N_r}(\tilde{R}(\xi_i^i)$$

$$+\tilde{R}(\hat{\xi}_i^i))$$

$$\begin{cases} 0 \leq \beta_i(\hat{\beta}_i) \leq C\tilde{R}(\xi_i^i)(\tilde{R}(\hat{\xi}_i^i)), i=1,2,\ldots,L \\ \xi_i^i(\hat{\xi}_i^i) \geq 0, i=1,2,\ldots,L \end{cases} \tag{51}$$

Let $\mathbf{a}=[\alpha_1,\alpha_2,\ldots,\alpha_{N_r}]^T$, $\hat{\mathbf{a}}=[\hat{\alpha}_1,\hat{\alpha}_2,\ldots,\hat{\alpha}_{N_r}]^T$, $\mathbf{b}=[\beta_1,\beta_2,\ldots,\beta_{N_r}]^T$,$\hat{\mathbf{b}}=[\hat{\beta}_1,\hat{\beta}_2,\ldots,\hat{\beta}_{N_r}]^T$, $\Re(\mathbf{y}) = [\Re(y_1),\Re(y_2),\ldots,\Re(y_{N_r})]^T$, $\Im(\mathbf{y}) = [\Im(y_1),\Im(y_2),\ldots,\Im(y_{N_r})]^T$, $\mathbf{R}^r = [\tilde{R}(\xi_1^r)+\tilde{R}(\hat{\xi}_1^r),\tilde{R}(\xi_2^r)+\tilde{R}(\hat{\xi}_2^r),\ldots,\tilde{R}(\xi_{N_r}^r)+\tilde{R}(\hat{\xi}_{N_r}^r)]^T$, $\mathbf{R}^i = [\tilde{R}(\xi_1^i)+\tilde{R}(\hat{\xi}_1^i),\tilde{R}(\xi_2^i)+\tilde{R}(\hat{\xi}_2^i),\ldots,\tilde{R}(\xi_{N_r}^i)+\tilde{R}(\hat{\xi}_{N_r}^i)]^T$, $\mathbf{e}=[1,1,\ldots,1]^T\in\mathbb{R}^{N_r}$. The alternate form can be written as

$$\max_{\mathbf{a},\hat{\mathbf{a}}} \quad \theta^r = -\frac{1}{2}(\mathbf{a}-\hat{\mathbf{a}})^T\Re(\mathbf{K})(\mathbf{a}-\hat{\mathbf{a}}) + \Re(\mathbf{y})^T(\mathbf{a}-\hat{\mathbf{a}}) - \epsilon(\mathbf{e}^T(a+\hat{\mathbf{a}})) + C(\mathbf{e}^T\mathbf{R}^r)$$

$$\begin{cases} 0 \leq \alpha_i(\hat{\alpha}_i) \leq C\tilde{R}(\xi_i^r)(\tilde{R}(\hat{\xi}_i^r)), i=1,2,\ldots,N_r \\ \xi_i^r(\hat{\xi}_i^r) \geq 0, i=1,2,\ldots,N_r \end{cases} \tag{52}$$

$$\max_{\mathbf{b},\hat{\mathbf{b}}} \quad \theta^i = -\frac{1}{2}(\mathbf{b}-\hat{\mathbf{b}})^T\Re(\mathbf{K})(\mathbf{b}-\hat{\mathbf{b}}) + \Im(\mathbf{y})^T(\mathbf{b}-\hat{\mathbf{b}}) - \epsilon(\mathbf{e}^T(\mathbf{b}+\hat{\mathbf{b}})) + C(\mathbf{e}^T\mathbf{R}^i)$$

$$\begin{cases} 0 \leq \beta_i(\hat{\beta}_i) \leq C\tilde{R}(\xi_i^i)(\tilde{R}(\hat{\xi}_i^i)), i = 1, 2, \ldots, N_r \\ \xi_i^i(\hat{\xi}_i^i) \geq 0, i = 1, 2, \ldots, N_r \end{cases} \tag{53}$$

Observe that solving (52) and (53) are equivalent to solving two independent real Support vector regression task (dual channel), where only the real part of the kernel matrix is required for each channel. This can provide computational advantages for solving the dual optimization problems.

IV. Work Set Selection and Solver

In (49) we have a quadratic optimization problem, The traditional optimization algorithms such as Newton, Quasi Newton can not be directly applied to this problem, because the sparseness of kernel matrix $\mathbf{K}$ can not be guaranteed, so that a prohibitive storage may be required when dealing with a large data set.

Decomposition methods are a set of efficient algorithms that can help to ease this difficulty. Decomposition methods work iteratively, and based on choosing a subset of variables S (named work set) to optimize in each iteration step, while keeping the rest variables N fixed. Sequential Minimal Optimization (SMO) is an extreme case of decomposition method, where the work set size is 2, and an analytic quadratic programming (QP) step instead of numerical QP step can be performed in each iteration.

Because (50) and (51) have the same structure, in this section we discuss the real part only.

By dividing the variables into work set S and fixed set N, we can divide vector $\mathbf{a}$ into two sub vectors $[\mathbf{a}_S, \mathbf{a}_N]$. thereafter $\mathbf{a}_S$ denotes a vector that consists of the components of $\mathbf{a}$ that belongs to set S, the same modifications are applied to $\hat{\mathbf{a}}$ and $\mathbf{y}$. Thus (52) can be rewritten as:

$$\max_{\mathbf{a}_S,\hat{\mathbf{a}}_S,\mathbf{a}_N,\hat{\mathbf{a}}_N} \theta = -\frac{1}{2}[(\mathbf{a}_S - \hat{\mathbf{a}}_S)^T, (\mathbf{a}_N - \hat{\mathbf{a}}_N)^T] \begin{bmatrix} \Re(\mathbf{K})_{SS} & \Re(\mathbf{K})_{SN} \\ \Re(\mathbf{K})_{NS} & \Re(\mathbf{K})_{NN} \end{bmatrix} \begin{bmatrix} (\mathbf{a}_S - \hat{\mathbf{a}}_S) \\ (\mathbf{a}_N - \hat{\mathbf{a}}_N) \end{bmatrix} + [\Re(\mathbf{y}_S)^T, \Re(\mathbf{y}_N)^T] \begin{bmatrix} (\mathbf{a}_S - \hat{\mathbf{a}}_S) \\ (\mathbf{a}_N - \hat{\mathbf{a}}_N) \end{bmatrix} -$$

$$\epsilon[\mathbf{e}_S^T, \mathbf{e}_N^T] \begin{bmatrix} (\mathbf{a}_S + \hat{\mathbf{a}}_S) \\ (\mathbf{a}_N + \hat{\mathbf{a}}_N) \end{bmatrix} + C(\mathbf{e}^T, \mathbf{R}^r)$$

$$\begin{cases} 0 \leq [\mathbf{a}]_i([\hat{\mathbf{a}}]_i) \leq C\tilde{R}(\xi_i^r)(\tilde{R}(\hat{\xi}_i^r)), i = 1, 2, \ldots, N_r \\ \xi_i^r(\hat{\xi}_i^r) \geq 0, i = 1, 2, \ldots, N_r \end{cases} \tag{54}$$

Where $\begin{bmatrix} \Re(\mathbf{K})_{SS} & \Re(\mathbf{K})_{SN} \\ \Re(\mathbf{K})_{NS} & \Re(\mathbf{K})_{NN} \end{bmatrix}$ is a permutation of $\Re(\mathbf{K})$, where $\Re(\mathbf{K})_{SN}$ denotes the sub matrix that consist of rows corresponding to set S and columns corresponding to N, Notice that $\Re(\mathbf{K})_{SN} = \Re(\mathbf{K})_{NS}$, therefore, (54) can be rewritten as

$$\max_{\mathbf{a},\hat{\mathbf{a}}} \quad \theta^r = -\frac{1}{2}[(\mathbf{a}_S - \hat{\mathbf{a}}_S)^T \Re(\mathbf{K})_{SS}(\mathbf{a}_S - \hat{\mathbf{a}}_S) + 2(\mathbf{a}_N - \hat{\mathbf{a}}_N)^T \Re(\mathbf{K})_{NS}(\mathbf{a}_S - \hat{\mathbf{a}}_S)] + \Re(\mathbf{y}_S^T)(\mathbf{a}_S - \hat{\mathbf{a}}_S) -$$

$$\epsilon(\mathbf{e}^T(\mathbf{a}_S + \hat{\mathbf{a}}_S)) - \frac{1}{2}(\mathbf{a}_N - \hat{\mathbf{a}}_N)^T \Re(\mathbf{K})_{NN}(\mathbf{a}_N - \hat{\mathbf{a}}_N) + \Re(\mathbf{y}_N^T)(\mathbf{a}_N - \hat{\mathbf{a}}_N) - \epsilon(\mathbf{e}^T(\mathbf{a}_N + \hat{\mathbf{a}}_N))$$

$$+C(\mathbf{e}^T(\mathbf{R}^r),$$

$$\begin{cases} 0 \leq [\mathbf{a}]_i([\hat{\mathbf{a}}]_i) \leq C\tilde{R}(\xi_i^r)(\tilde{R}(\hat{\xi}_i^r)), i = 1, 2, \ldots, N_r \\ \xi_i^r(\hat{\xi}_i^r) \geq 0, i = 1, 2, \ldots, N_r \end{cases} \tag{55}$$

In each iteration, in (55), $\mathbf{a}_N$ is fixed and only the sub QP problem that related to $\mathbf{a}_S$ is solved i.e

$$\max_{\mathbf{a}_S,\hat{\mathbf{a}}_S} \quad \theta^r_S = -\frac{1}{2}[(\mathbf{a}_S-\hat{\mathbf{a}}_S)^T\Re(\mathbf{K})_{SS}(\mathbf{a}_S-\hat{\mathbf{a}}_S)] + [Re(\mathbf{y}^T_S) - (\mathbf{a}_N-\hat{\mathbf{a}}_N)^T\mathbf{K}^r_{NS}](\mathbf{a}_S-\hat{\mathbf{a}}_S) -$$

$$\epsilon < \mathbf{e}^T, (\mathbf{a}_S+\hat{\mathbf{a}}_S) >,$$

$$\begin{cases} 0 \leq [\mathbf{a}_S]_i([\hat{\mathbf{a}}_S]_i) \leq C\tilde{R}(\xi^r_i)(\tilde{R}(\hat{\xi}^r_i)), i \in S \\ \xi^r_i(\hat{\xi}^r_i) \geq 0, i \in S \end{cases} \tag{56}$$

A proper work set selection strategy is required so that speed and performance requirement can be guaranteed. One idea is to perform update to dual variables that violate the complementary KKT conditions, which, based on (18) in II-C, can be formulated as

$$\begin{cases} (C\tilde{R}(\xi^r_i) - \alpha_i)\xi^r_i = 0, i = 1, 2, \ldots, L \\ (C\tilde{R}(\hat{\xi}^r_i) - \hat{\alpha}_i)\hat{\xi}^r_i = 0, i = 1, 2, \ldots, L \\ \alpha_i(\Re(\mathbf{y}_i) - < \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - \epsilon - \xi^r_i) = 0, i = 1, 2, \ldots, L \\ \hat{\alpha}_i(-\mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - \Re(\mathbf{y}_i) - \epsilon - \hat{\xi}^r_i) = 0, i = 1, 2, \ldots, L \end{cases} \tag{57}$$

According to Osuna's theorem [35], the final convergence can be guaranteed. In SMO algorithm, heuristic methods are used to find a work set of size two in order to accelerate the decomposition process [34]. The heuristic method first searches among the non-bound variables (that is $0 < \alpha_i < C\tilde{R}(\xi^r_i)$ and $0 < \hat{\alpha}_i < C\tilde{R}(\hat{\xi}_i)$), which are more likely to violate the complementary KKT conditions, Then searching the whole dual variable set. The second dual variable that can maximize optimization step size is chosen. An approximate step size is used as evaluator for sake of reducing computational cost.

Another idea for work set selection is to choose the dual variables whose update can provide the maximum improvement to dual objective function. The improvement of the sub dual objective function is

$$\bigtriangledown\theta_S^r = \theta_S^r((\mathbf{a}_S + \Delta_S), (\hat{\mathbf{a}}_S + \hat{\Delta}_S)) - \theta_S^r(\mathbf{a}_S, \hat{\mathbf{a}}_S), \tag{58}$$

where $\Delta_S = \mathbf{a}_S^{new} - \mathbf{a}_S$, $\hat{\Delta}_S = \hat{\mathbf{a}}_S^{new} - \hat{\mathbf{a}}_S$, $\mathbf{a}_S^{new}$ and $\hat{\mathbf{a}}_S^{new}$ are the updates of $\mathbf{a}_S$ and $\hat{\mathbf{a}}_S$. $\bigtriangledown\theta_S^r$ in (58), based on (56), can be written as

$$\begin{aligned}&\bigtriangledown\theta_S^r = -\frac{1}{2}[(\Delta_S - \hat{\Delta}_S)^T \Re(\mathbf{K})_{SS}(\Delta_S - \hat{\Delta}_S) + 2(\mathbf{a}_S - \hat{\mathbf{a}}_S)^T \Re(\mathbf{K})_{SS}(\Delta_S - \hat{\Delta}_S)] + [\Re(\mathbf{y}_S^T) - (\mathbf{a}_N - \hat{\mathbf{a}}_N)^T \\ &\Re(\mathbf{K})_{NS}](\Delta_S - \hat{\Delta}_S) - \epsilon\mathbf{e}_S^T(\Delta_S + \hat{\Delta}_S) = -\frac{1}{2}(\Delta_S - \hat{\Delta}_S)^T \Re(\mathbf{K})_{SS}(\Delta_S - \hat{\Delta}_S) + \{\Re(\mathbf{y}_S^T) - [(\mathbf{a}_S - \hat{\mathbf{a}}_S)^T \\ &\Re(\mathbf{K})_{SS} + (\mathbf{a}_N - \hat{\mathbf{a}}_N)^T \Re(\mathbf{K})_{NS}]\}(\Delta_S - \hat{\Delta}_S) - \epsilon\mathbf{e}_S^T(\Delta_S + \hat{\Delta}_S),\end{aligned} \tag{59}$$

where we have

$$\begin{aligned}&(\mathbf{a}_S - \hat{\mathbf{a}}_S)^T \Re(\mathbf{K})_{SS} + (\mathbf{a}_N - \hat{\mathbf{a}}_N)^T \Re(\mathbf{K})_{NS} = [(\mathbf{a}_S - \hat{\mathbf{a}}_S)^T, (\mathbf{a}_N - \hat{\mathbf{a}}_N)^T] \begin{bmatrix} \Re(\mathbf{K})_{SS} \\ \Re(\mathbf{K})_{NS} \end{bmatrix} = \\ &(\mathbf{a} - \hat{\mathbf{a}})^T \Re(\mathbf{K})_S,\end{aligned} \tag{60}$$

where $\Re(\mathbf{K})_S \in \mathbb{R}^{N_r \times S}$ denotes the matrix constructed by all the columns corresponding to the work set S. Therefore, (59) can be rewritten as

$$\begin{aligned}&\bigtriangledown\theta_S^r = -\frac{1}{2}(\Delta_S - \hat{\Delta}_S)^T \Re(\mathbf{K})_{SS}(\Delta_S - \hat{\Delta}_S) + [\Re(\mathbf{y}_S^T) - (\mathbf{a} - \hat{\mathbf{a}})^T \Re(\mathbf{K})_S](\Delta_S - \hat{\Delta}_S) - \epsilon\mathbf{e}_S^T \\ &(\Delta_S + \hat{\Delta}_S),\end{aligned} \tag{61}$$

define intermediate variable vector $\Phi \in \mathbb{C}^{N_r}$, $\Re(\Phi) = \Re(\mathbf{y}) - \Re(\mathbf{K})(\mathbf{a} - \hat{\mathbf{a}})$ and $\Im(\Phi) = \Im(\mathbf{y}) - \Re(\mathbf{K})(\mathbf{b} - \hat{\mathbf{b}})$, let Φ_S denote the vector that consists of the components in Φ that corresponding to S. Thus (61) can be rewritten as

$$\bigtriangledown\theta^r_S = -\frac{1}{2}(\Delta_S - \hat{\Delta})^T_S\Re(\mathbf{K})_{SS}(\Delta - \hat{\Delta})_S + (\Re(\Phi)_S)^T(\Delta - \hat{\Delta})_S - \epsilon\mathbf{e}^T_S$$
$$(\Delta + \hat{\Delta})_S \tag{62}$$

Because the offset term is omitted in Large MIMO regression model, therefore different from SMO type algorithms, there is no linear equation constraint, which is inducted by offset b, as shown in (22). Therefore it is possible to update only one variable pair in each iteration. However, more efficient work set selection strategies based on maximum dual objective gain selection without considering offset term, is proposed in [36]. A work set of size two is updated in each iteration. The computational cost is reduced while maintaining the comparable performance with that with offset. In CSVR large MIMO detector, the work set selection strategy in [36] is exploited. Basically, in each iteration this strategy uses sequential single variable searchings, whose searching time is $O(n)$, where n is the number of data set, to determine a work set of size two. On the one hand, this strategy can find a work set of size two, whose update approximately maximize the gain in the dual objective function. On the other hand, the brute-force searching for the work size of two, whose update can maximize the gain in the dual objective function, requires $O(n^2)$ searching time. The former strategy requires as few iterations as the latter one, since in each iteration, the latter strategy is more expensive, the former one can enjoy significantly faster running speed.

A. Single Direction Solver

Recall the KKT complementary conditions

$$(C\tilde{R}(\xi_i^r) - \alpha_i)\xi_i^r = 0, i = 1, 2, \ldots, N_r \tag{63}$$

$$(C\tilde{R}(\hat{\xi}_i^r) - \hat{\alpha}_i)\hat{\xi}_i^r = 0, i = 1, 2, \ldots, N_r \tag{64}$$

$$\alpha_i(\Re(\mathbf{y}_i) - < \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - \epsilon - \xi_i^r) = 0, i = 1, 2, \ldots, N_r \tag{65}$$

$$\hat{\alpha}_i(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - \Re(\mathbf{y}_i) - \epsilon - \hat{\xi}_i^r) = 0, i = 1, 2, \ldots, N_r \tag{66}$$

Recall the discussions of slack variables by (3) and (4) in section II-C. we have $\xi_i^r \hat{\xi}_i^r = 0$. similar to the definitions in (30), we have

$$\xi_i^r(\hat{\xi}_i^r) = |\Re(\mathbf{y}_i) - \Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}})|_\epsilon, \tag{67}$$

therefore in (65) and (66), there is at least one of $(\Re(\mathbf{y}_i) - < \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - \epsilon - \xi_i^r)$ and $(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} - \Re(\mathbf{y}_i) - \epsilon - \hat{\xi}_i^r)$ can be non zero, therefore in order to satisfy equalities (65) and (66), at least one of α_i and $\hat{\alpha}_i$ need to be zero, that is $\alpha_i \hat{\alpha}_i = 0$.

Hence we can substitute $\lambda_i = \alpha_i - \hat{\alpha}_i$ and $|\lambda_i| = \alpha_i + \hat{\alpha}_i$, therefore the update unit is changed to the single optimization variable λ_i rather than the pair α_i and $\hat{\alpha}_i$.

We first introduce 1-D work set selection strategy in which single optimization variable whose update can maximize the gain of dual objective function is chosen as the work set in each iteration. Define $\Lambda = [\lambda_1, \lambda_2, \ldots, \lambda_{N_r}]$, $\sigma_i = \lambda_i^{new} - \lambda_i$, $\Sigma = [\sigma_1, \sigma_2, \ldots, \sigma_{N_r}]$. Substitute $\mathbf{a}, \hat{\mathbf{a}}$ with Λ and using Σ. Let Λ_S and Σ_S denote the vector that consist of the components in Λ and

Σ that belong to work set S, the sub dual objective function in (56) and its gain (62)

$$\max_{\Lambda_S} \quad \theta_S^r = -\frac{1}{2}[\Lambda_S^T \Re(\mathbf{K})_{SS}\Lambda_S] + [\Re(\mathbf{y})_S^T - \Lambda_N^T \Re(\mathbf{K})_{NS}]\Lambda_S - \epsilon < \mathbf{e}_S^T, |\Lambda_S| >, \tag{68}$$

$$\bigtriangledown \theta_S^r = -\frac{1}{2}\Sigma_S^T \Re(\mathbf{K})_{SS}\Sigma_S + (\Phi_S^r)^T \Sigma_S - \epsilon < \mathbf{e}_S^T, |\Lambda_S^{new}| - |\Lambda_S| >, \tag{69}$$

In 1-D solver, $|S| = 1$, based on (69), the sub dual objective function corresponding to λ_m, $m = 1, 2, \ldots, N_r$ can be written as

$$\max_{\lambda_m} \quad \theta_m^r = -\frac{1}{2}(\lambda_m^{new})^2 \Re(\mathbf{K})_{mm} + [\Re(\mathbf{y}_m) - \sum_{j \neq m}^{N_r} \Re(\mathbf{K})_{mj}\lambda_j]\lambda_m^{new} - \epsilon(|\lambda_m^{new}|), \tag{70}$$

Based on the definition of Φ in (62), $\Re(\Phi)_i = \Re(\mathbf{y}_i) - \sum_{j=1}^{N_r} \lambda_j^r \Re(\mathbf{K})_{ij}$, similarly, as to dual variable λ_i^i, $\Re(\Phi)_i^i = Im(y_i) - \sum_{j=1}^{N_r} \lambda_j^i \Re(\mathbf{K})_{ij}$. Thereafter, for sake of brevity, we use λ_i instead of λ_i^r. Take the partial derivative of θ_m^r respect to $\tilde{\lambda}_m^{new}$, we have

$$\begin{aligned}
&\frac{\partial \theta_m^r}{\partial \tilde{\lambda}_m^{new}} = -\tilde{\lambda}_m^{new} \Re(\mathbf{K})_{mm} + \Re(\mathbf{y}_m) - \sum_{j \neq m}^{N_r} \lambda_j \Re(\mathbf{K})_{mj} - \epsilon(sgn(\tilde{\lambda}_m^{new})) = \\
&-\tilde{\lambda}_m^{new} \Re(\mathbf{K})_{mm} + \Re(\Phi)_m + \lambda_m \Re(\mathbf{K})_{mm} - \epsilon(sgn(\tilde{\lambda}_m^{new})) \\
&\Rightarrow \tilde{\lambda}_m^{new} = \lambda_m + \frac{\Re(\Phi)_m - \epsilon(sgn(\tilde{\lambda}_m^{new}))}{\Re(\mathbf{K})_{mm}},
\end{aligned} \tag{71}$$

The update of λ_m is completed by clipping

$$\lambda_m^{new} = [\tilde{\lambda}_m^{new}]_{-CR'(\xi_m)}^{CR'(\xi_m)} \tag{72}$$

where $[]_a^b$ denotes clipping function

$$[x]_a^b = \begin{cases} a \quad if \quad x \leq a \\ x \quad if \quad a < x < b \\ b \quad if \quad x \geq b \end{cases} \tag{73}$$

based on (69), The maximal gain of objective function with respect to the mth optimization variable λ_m is

$$\begin{aligned} \bigtriangledown\theta_m^r &= \theta_m^r(\lambda_m + \sigma_m) - \theta_m^r(\lambda_m) \\ &= -\frac{1}{2}\sigma_m^2 \Re(\mathbf{K})_{mm} + \Re(\Phi)_m \sigma_m - \epsilon(|\lambda_m^{new}| - |\lambda_m|) \\ &= \sigma_m[-\frac{1}{2}\sigma_m \Re(\mathbf{K})_{mm} + \Re(\Phi)_m] - \epsilon(|\lambda_m^{new}| - |\lambda_m|), \end{aligned} \tag{74}$$

In 1-D searching procedure, the optimization variable whose update can achieve the maximum gain of sub dual objective function is chosen, assume the kth optimization variable is chosen, based on the this principle

$$k = \arg \max_{(m=1,2,..,N_r)} \bigtriangledown\theta_m^r. \tag{75}$$

Notice in (71), $sgn(\tilde{\lambda}_m^{new})$ is unknown before λ_m is updated, therefore two conditions $sgn(\tilde{\lambda}_m^{new}) = 1$ and $sgn(\tilde{\lambda}_m^{new}) = -1$ are considered. The final decision of $sgn(\tilde{\lambda}_m^{new})$ is made by comparing the corresponding gains of the sub objective functions in (74).

B. Double Direction Solver

Although the omission of offset in the CSVR-MIMO detector makes 1-D solver possible, however, recent work in machine learning field shows training SVM without offset by 2-D solver with special work set selection strategies has more rapid training speed while the comparable performance is retained [36]. The optimal 2-D solver uses the same principle as 1-D solver, $|S| = 2$, assume the work set consists of the mth and nth optimization variables, that are $\Lambda_S = [\lambda_m, \lambda_n]$. Based on (69), the sub dual objective function can be written as

$$\begin{aligned}
&\max_{\lambda_m, \lambda_n} \quad \theta^r_{m,n} = -\frac{1}{2}[(\tilde{\lambda}^{new}_m)^2 \Re(\mathbf{K})_{mm} + (\tilde{\lambda}^{new}_n)^2 \Re(\mathbf{K})_{nn} + 2\tilde{\lambda}^{new}_m \tilde{\lambda}^{new}_n \Re(\mathbf{K})_{mn}] - \\
&\tilde{\lambda}^{new}_m \sum_{j \neq m,n}^{N_r} \lambda_j \Re(\mathbf{K})_{mj} - \tilde{\lambda}^{new}_n \sum_{j \neq m,n}^{N_r} \lambda_j \Re(\mathbf{K})_{nj} + \Re(\mathbf{y}_m)\tilde{\lambda}^{new}_1 + \Re(\mathbf{y}_n)\tilde{\lambda}^2_n \\
&-\epsilon(|\lambda^{new}_m| + |\tilde{\lambda}^{new}_n|), \qquad (76)
\end{aligned}$$

Based on (76), the partial derivatives of $\theta^r_{m,n}$ with respect to $\tilde{\lambda}^{new}_m$ and $\tilde{\lambda}^{new}_n$ are

$$\begin{aligned}
&\frac{\partial \theta^r_{m,n}}{\partial \tilde{\lambda}^{new}_m} = -\tilde{\lambda}^{new}_m \Re(\mathbf{K})_{mm} - \tilde{\lambda}^{new}_n \Re(\mathbf{K})_{mn} - \sum_{j \neq m,n}^{N_r} \lambda_j \Re(\mathbf{K})_{mj} + \Re(\mathbf{y}_m) - \epsilon sgn(\tilde{\lambda}^{new}_m) = \\
&-\tilde{\lambda}^{new}_m \Re(\mathbf{K})_{mm} - \tilde{\lambda}^{new}_n \Re(\mathbf{K})_{mn} + \Re(\Phi)_m + \lambda_m \Re \mathbf{K}_{mm} + \lambda_n \Re(\mathbf{K})_{mn} - \epsilon sgn(\tilde{\lambda}^{new}_m) = 0 (77) \\
&\frac{\partial \theta^r_{m,n}}{\partial \tilde{\lambda}^{new}_n} = -\tilde{\lambda}^{new}_n \Re(\mathbf{K})_{nn} - \tilde{\lambda}^{new}_m \Re(\mathbf{K})_{mn} - \sum_{j \neq m,n}^{N_r} \lambda_j \Re(\mathbf{K})_{nj} + \Re(\mathbf{y}_n) - \epsilon sgn(\tilde{\lambda}^{new}_n) = \\
&-\tilde{\lambda}^{new}_n \Re(\mathbf{K})_{nn} - \tilde{\lambda}^{new}_m \Re(\mathbf{K})_{mn} + \Re(\Phi)_n + \lambda_m \Re(\mathbf{K})_{mn} + \lambda_n \Re(\mathbf{K})_{nn} - \epsilon sgn(\tilde{\lambda}^{new}_n) = 0 \ (78)
\end{aligned}$$

where $\frac{\partial |x|}{\partial x} = sgn(x)$ denotes the sign of x. Based on (77) and (78) we have

$$(\tilde{\lambda}^{new}_m - \lambda_n)\Re(\mathbf{K})_{mm} = \Re(\Phi)_m - \epsilon sgn(\tilde{\lambda}^{new}_m) - (\tilde{\lambda}^{new}_n - \lambda_n)\Re(\mathbf{K})_{mn} \qquad (79)$$

$$(\tilde{\lambda}^{new}_n - \lambda_n)\Re(\mathbf{K})_{nn} = \Re(\Phi)_n - \epsilon sgn(\tilde{\lambda}^{new}_n) - (\tilde{\lambda}^{new}_m - \lambda_m)\Re(\mathbf{K})_{mn} \qquad (80)$$

hence based on (79) and (80), the update formula of $\tilde{\lambda}_m^{new}$ and $\tilde{\lambda}_n^{new}$ are

$$\tilde{\lambda}_m^{new} = \lambda_m + \frac{\Re(\Phi)_m \Re(\mathbf{K})_{nn} - \Re(\Phi)_n \Re(\mathbf{K})_{mn} - \epsilon[sgn(\tilde{\lambda}_m^{new})\Re(\mathbf{K})_{nn} - sgn(\tilde{\lambda}_n^{new})\Re(\mathbf{K})_{mn}]}{\Re(\mathbf{K})_{mm}\Re(\mathbf{K})_{nn} - (\Re(\mathbf{K})_{mn})^2} \tag{81}$$

$$\tilde{\lambda}_n^{new} = \lambda_n + \frac{\Re(\Phi)_n \Re(\mathbf{K})_{mm} - \Re(\Phi)_m \Re(\mathbf{K})_{mn} - \epsilon[sgn(\tilde{\lambda}_n^{new})\Re(\mathbf{K})_{mm} - sgn(\tilde{\lambda}_m^{new})\Re(\mathbf{K})_{mn}]}{\Re(\mathbf{K})_{mm}\Re(\mathbf{K})_{nn} - (\Re(\mathbf{K})_{mn})^2} \tag{82}$$

Then the updated optimization variables are clipped by constraint

$$\begin{aligned} &\lambda_i^{new} = [\tilde{\lambda}_i^{new}]_{-CR'(\xi_i)}^{CR'(\xi_i)}, \\ &i = m, n \end{aligned} \tag{83}$$

Based on (62), the gain of 2-D solver objective function can be written as

$$\begin{aligned} &\bigtriangledown\theta_{mn}^r = -\frac{1}{2}[\sigma_m^2\Re(\mathbf{K})_{mm} + \sigma_n^2\Re(\mathbf{K})_{nn} + 2\sigma_m\sigma_n\Re(\mathbf{K})_{mn}] + \Re(\Phi)_m\sigma_m + \Re(\Phi)_n\sigma_n \\ &-\epsilon(|\lambda_m^{new}| - |\lambda_m| + |\lambda_n^{new}| - |\lambda_n|), \end{aligned} \tag{84}$$

assume the ith and jth optimization variables are chosen, the optimization variables in 2-D solver have the same update rule as that of 1-D solver, that is

$$[i, j] = \arg \max_{m,n=1,2,\ldots,N_r} \bigtriangledown\theta_{m,n}^r. \tag{85}$$

modify (87), we have

$$\bigtriangledown\theta^r_{mn} = \sigma_m[-\frac{1}{2}\sigma_m\Re(\mathbf{K})_{mm} + \Re(\Phi)_m] - \epsilon(|\lambda^{new}_m| - |\lambda_m|) + \sigma_n[-\frac{1}{2}\sigma_n\Re(\mathbf{K})_{nn} + \Re(\Phi)_n] - \epsilon(|\lambda^{new}_n| - |\lambda_n|)$$
$$-\sigma_m\sigma_n\Re(\mathbf{K})_{mn}, \tag{86}$$

Similar to single direction solver, notice in (81) and (82), $sgn(\tilde{\lambda}^{new}_m)$ and $sgn(\tilde{\lambda}^{new}_n)$ are unknown before λ_m and λ_n are updated, therefore four conditions are considered

$$sgn(\tilde{\lambda}^{new}_m) = 1, sgn(\tilde{\lambda}^{new}_n) = 1$$
$$sgn(\tilde{\lambda}^{new}_m) = 1, sgn(\tilde{\lambda}^{new}_n) = -1$$
$$sgn(\tilde{\lambda}^{new}_m) = -1, sgn(\tilde{\lambda}^{new}_n) = 1$$
$$sgn(\tilde{\lambda}^{new}_m) = -1, sgn(\tilde{\lambda}^{new}_n) = -1$$

The final decisions of $sgn(\tilde{\lambda}^{new}_m)$ and $sgn(\tilde{\lambda}^{new}_n)$ are made by comparing the corresponding possible gains of the sub objective functions in (87).

Recall the gain of dual objective function of 1-D solver in (74), we obtain

$$\bigtriangledown\theta^r_{ij} = \bigtriangledown\theta^r_i + \bigtriangledown\theta^r_j - \sigma_i\sigma_j\Re(\mathbf{K})_{ij}, \tag{87}$$

where $\bigtriangledown\theta^r_i$, $\bigtriangledown\theta^r_j$ denote gains of 1-D solver with mth and nth dual variable pairs are chosen.

C. Approximation of Optimal Double Direction Solver based on Single Direction Solver

From (87), we can observe that the gain of 2-D solver is a summation of the gain of 2 independent 1-D solver and a correlation term $\sigma_i\sigma_j\Re(\mathbf{K})_{ij}$.

Obviously optimal 2-D work set $[i, j]$ can be determined by brute-force searching manner, which requires $O(n^2)$ searching times. Based on (87), we can approximate optimal 2-D searching strategy by the combinations of optimal 1-D searching approach, we will prove in the large MIMO systems, when N_t is sufficient large, this approximation is very effective. Here we propose two kinds of 1-D approximate searching strategy:

1) One-shot 1-D Searching: do one round 1-D searching and calculate all the 1-D gain based on (74), then choose the two optimization variables indexed by i and j, whose update can achieve the first and the second largest gain of the sub dual objective functions.

2) Sequential 1-D Searching: do two rounds 1-D searchings, in the first round find optimization variable indexed by i whose update can achieve the maximal 1-D gain of the sub dual objective functions, then update the ith optimization variable. In the second round, find jth optimization variable whose update can achieve the maximal 1-D gain of the sub dual objective functions.

The effectiveness of 1-D approximation strategies are majorly determined by the ratio $\frac{\sigma_i \sigma_j \Re(\mathbf{K})_{ij}}{\triangledown\theta_i^r + \triangledown\theta_j^r}$. Hence we provide theoretical analyses based on the view of channel hardening phenomenon. It can be proved that in the large MIMO systems, the correlation term is ignorable comparing to 1-D gains of the sub objective function. Prior to the theoretical analyse, we first investigate some mathematical properties of channel hardening (to be completed).

V. Stopping Criteria

As we have explained in section II-C, the upper bound of Lagrangian dual objective function is determined by primal objective function, further more the optimal of primal and dual objective function is found if and only if the equality holds, that is

$$\theta(\lambda^r, \lambda^i) = f(\mathbf{w}, \xi) \tag{88}$$

$$\frac{1}{2}||\mathbf{w}||^2_{\mathbb{H}} + C\sum_{i=1}^{N_r}[R(\xi_i^r) + R(\hat{\xi}_i^r) + R(\xi_i^i) + R(\hat{\xi}_i^i)], \tag{89}$$

(49) can be rewritten as follow by substituting $\lambda^r = \alpha - \hat{\alpha}$, $|\lambda^r| = \alpha + \hat{\alpha}$ and $\lambda^i = \beta - \hat{\beta}$, $|\lambda^i| = \beta + \hat{\beta}$

$$\begin{aligned}\theta(\lambda^r, \lambda^i) = &-\frac{1}{2} < (\lambda^r)^T, \mathbf{K}^r\lambda^r > -\frac{1}{2} < (\lambda^i)^T, \mathbf{K}^r\lambda^i > + < Re(\mathbf{y})^T, \lambda^r > + < Im(\mathbf{y})^T, \lambda^i > \\ &-\epsilon < \mathbf{e}^T, (|\lambda^r| + |\lambda^i|) > +C\sum_{i=1}^{N_r}[\tilde{R}(\xi_i^r) + \tilde{R}(\hat{\xi}_i^r) + \tilde{R}(\xi_i^i) + \tilde{R}(\hat{\xi}_i^i)],\end{aligned} \tag{90}$$

Similarly, (43) can be formulated as

$$||\mathbf{W}||^2_{\mathbb{H}} =< (\lambda^r)^T, \mathbf{K}^r\lambda^r > + < (\lambda^i)^T, \mathbf{K}^r\lambda^i > -2 < \lambda^r, \mathbf{K}^i\lambda^i >, \tag{91}$$

hence, duality gap can be formulated as

$$\begin{aligned}G(\lambda^r, \lambda^i) = f(\mathbf{w}, \xi) - \theta(\lambda^r, \lambda^i) =&< (\lambda^r)^T, \mathbf{K}^r\lambda^r > + < (\lambda^i)^T, \mathbf{K}^r\lambda^i > - < Re(\mathbf{y})^T, \lambda^r > - < Im(\mathbf{y})^T, \lambda^i > \\ -\epsilon < \mathbf{e}^T, (|\lambda^r| + |\lambda^i|) > +C\sum_{i=1}^{N_r}&[\xi_i^r R^{'}(\xi_i^r) + \hat{\xi}_i^r R^{'}(\hat{\xi}_i^r) + \xi_i^i R^{'}(\xi_i^i) + \hat{\xi}_i^i R^{'}(\hat{\xi}_i^i)] - 2 < \lambda^r, \mathbf{K}^i\lambda^i > .\end{aligned} \tag{92}$$

As we explained in section II-B, the choice of risk function is determined by distribution of noise, as to Gaussian noise, the risk function is

$$R(u) = \frac{1}{2}u^2, \tag{93}$$

hence

$$\tilde{R}(u) = R(u) - uR'(u) = -\frac{1}{2}u^2, \tag{94}$$

In ϵ-SVR, the objective to exploit slack variables ξ_i^r and $\hat{\xi}_i^r$ is to compensate the influences from the outliers that exceed the ϵ-tube which are caused by noise. Therefore in ϵ-SVR, ξ_i^r and $\hat{\xi}^r$ are defined as

$$\xi_i^r = \max(0, \Re(\mathbf{y}_i) - \Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}}) - \epsilon) \tag{95}$$

$$\hat{\xi}_i^r = \max(0, \Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}}) - \Re(\mathbf{y}_i) - \epsilon) \tag{96}$$

Because the distance between the estimations $\Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}})$ and the observations $\Re(\mathbf{y}_i)$ can only exceeds the ϵ-tube in one direction, therefore there is at most one of ξ_i^r and $\hat{\xi}_i^r$ can be non zero. That is $\xi_i^r \hat{\xi}_i^r = 0$. Therefore the risk function can be rewritten as

$$R(\xi_i^r) + R(\hat{\xi}_i^r) = \frac{1}{2}|Re(\mathbf{y}_i) - Re(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}})|_\epsilon^2 \tag{97}$$

$$R(\xi_i^i) + R(\hat{\xi}_i^i) = \frac{1}{2}|Im(\mathbf{y}_i) - Im(< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}})|_\epsilon^2 \tag{98}$$

where $|\cdot|_\epsilon$ denotes ϵ insensitive function as mentioned in section II-B. Recall (42),

$$< \mathbf{h}_i, \mathbf{w} >_{\mathbb{H}} = \sum_{j=1}^{N_r} \lambda_j^r \Re(\mathbf{K})_{ij} - \sum_{j=1}^{N_r} \lambda_j^i \Im(\mathbf{K})_{ij} + i(\sum_{j=1}^{N_r} \lambda_j^r \Im(\mathbf{K})_{ij} + \sum_{j=1}^{N_r} \lambda_j^i \Re(\mathbf{K})_{ij}), \tag{99}$$

we obtain

$$Re(\mathbf{y}_i) - Re(< \mathbf{h}_i, \mathbf{W} >_{\mathbb{H}}) = Re(\mathbf{y}_i) - \sum_{j=1}^{N_r} \lambda_j^r \mathbf{K}_{ij}^r + \sum_{j=1}^{N_r} \lambda_j^i \mathbf{K}_{ij}^i \tag{100}$$

$$Im(\mathbf{y}_i) - Im(< \mathbf{h}_i, \mathbf{W} >_{\mathbb{H}}) = Im(\mathbf{y}_i) - \sum_{j=1}^{N_r} \lambda_j^i \mathbf{K}_{ij}^r - \sum_{j=1}^{N_r} \lambda_j^r \mathbf{K}_{ij}^i \tag{101}$$

Two intermediate variables Φ and Ψ are defined

$$\Phi^r = Re(\mathbf{y}) - \mathbf{K}^r \lambda^r ; \Phi^i = Im(\mathbf{y}) - \mathbf{K}^r \lambda^i \tag{102}$$

$$\Psi^r = \mathbf{K}^i \lambda^i ; \Psi^i = -\mathbf{K}^i \lambda^r \tag{103}$$

Therefore based on (98)-(103), duality gap in (92) can be rewritten as

$$G(\lambda^r, \lambda^i) =< (\lambda^r)^T, \mathbf{K}^r \lambda^r > + < (\lambda^i)^T, \mathbf{K}^r \lambda^i > - < Re(\mathbf{y})^T, \lambda^r > - < Im(\mathbf{y})^T, \lambda^i >$$

$$+\epsilon < \mathbf{e}^T, (|\lambda^r| + |\lambda^i|) > +C \sum_{i=1}^{N_r} [(|\Phi_i^r + \Psi_i^r|)_\epsilon^2 + (|\Phi_i^i + \Psi_i^i|)_\epsilon^2] - 2 < \lambda^r, \mathbf{K}^i \lambda^i > . \tag{104}$$

Based on objective function in (90), (104) can be rewritten as

$$G = (\lambda_r, \lambda_i) =< Re(\mathbf{y})^T, \lambda^r > + < Im(\mathbf{y})^T, \lambda^i > -\epsilon < \mathbf{e}^T, (|\lambda^r| + |\lambda^i|) > -2\theta(\lambda_i, \lambda_j)$$

$$-2 < \lambda^r, \mathbf{K}^i \lambda^i > . \tag{105}$$

The duality gap between primal problem and dual problem is used to evaluate how close a solution is to global minimum. In our scenario, duality gap is employed as stopping criteria. Therefore to make stopping criteria more effective to monitor if algorithm convergent, we monitor the ratio by a value of tolerance (usually this tolerance is set to 10^{-3}).

$$\frac{G}{G+\theta} \tag{106}$$

A. Update Φ, Ψ and G

In each iteration Φ, Ψ and G are updated partially based on 2 updated optimization variables. Here we give the pseudo code to update Φ, Ψ and G.

Based on the definition of Φ and Ψ in (102) and (103), we have the following procedure to update Φ and Ψ in real channel and imaginary channel, assume the optimization coordinate updated in each channel are 1 and 2.

```
procedure 1. UPDATE $\Phi^r$ AND $\Psi^i$ IN REAL CHANNEL
    for $i = 1 : N_r$ do
        $\Phi_i^r = \Phi_i^r - \sigma_1^r \mathbf{K}_{i1}^r - \sigma_2^r \mathbf{K}_{i2}^r$
        $\Psi_i^i = \Psi_i^i - \sigma_1^r \mathbf{K}_{i1}^i - \sigma_2^r \mathbf{K}_{i2}^i$
    end for
end procedure
```

```
procedure 2. UPDATE $\Phi^i$ AND $\Psi^r$ IN IMAGINARY CHANNEL
    for $i = 1 : N_r$ do
        $\Phi_i^i = \Phi_i^i - \sigma_1^i \mathbf{K}_{i1}^r - \sigma_2^i \mathbf{K}_{i2}^r$
        $\Psi_i^r = \Psi_i^r + \sigma_1^i \mathbf{K}_{i1}^i + \sigma_2^i \mathbf{K}_{i2}^i$
    end for
end procedure
```

Then the risk function term in (104) is updated as following

The pseudo code to update duality gap G based on (104) is shown as follow assume the coordinate updated in real channel is i and j, in imaginary channel is m and f.

```
procedure 3. UPDATE RISK FUNCTION IN REAL CHANNEL($\chi^r$)
    $\chi^r = 0$                                        ▷ initial risk term
    for $i = 1 : N_r$ do
        if $|\Phi_i^r + \Psi_i^r| > \epsilon$ then
            $\chi^r += (|\Phi_i^r + \Psi_i^r| - \epsilon)^2$
        end if
    end for
end procedure
```

```
procedure 4. UPDATE RISK FUNCTION IN IMAGINARY CHANNEL($\chi^i$)
    $\chi^i = 0$                                        ▷ initial risk term
    for $i = 1 : N_r$ do
        if $|\Phi_i^i + \Psi_i^i| > \epsilon$ then
            $\chi^i += (|\Phi_i^i + \Psi_i^i| - \epsilon)^2$
        end if
    end for
end procedure
```

Pseudo code for sequential single searching 2-D solver is shown as following

The following is the pseudo code of complex support vector detector (CSVD) is shown in Appendix A

```
procedure 5. UPDATE $G$
    $G+ = Re(\mathbf{y}_1)\sigma_i^r + Re(\mathbf{y}_2)\sigma_j^r$
    $G+ = Im(\mathbf{y}_1)\sigma_m^i + Re(\mathbf{y}_2)\sigma_f^i$
    $G- = \epsilon(|\lambda_i^r + \sigma_i^r| - |\lambda_i^r| + |\lambda_j^r + \sigma_j^r| - |\lambda_j^r|)$
    $G- = \epsilon(|\lambda_m^i + \sigma_m^i| - |\lambda_m^i| + |\lambda_f^i + \sigma_f^i| - |\lambda_f^i|)$
    $G- = 2(\bigtriangledown\theta_{i,j,m,f}(\sigma_i^r, \sigma_j^r, \sigma_m^i, \sigma_f^i))$    ▷ Update sub objective function based on (87)
    $G+ = C(\chi^r + \chi^i)^{new} - (\chi^r + \chi^i)$ ▷ Update risk function term based on Procedure 3 and Procedure 4
    $G- = \sigma_i^r\sigma_m^i\mathbf{K}_{im}^i + \sigma_i^r\sigma_f^i\mathbf{K}_{if}^i + \sigma_j^r\sigma_m^i\mathbf{K}_{jm}^i + \sigma_j^r\sigma_f^i\mathbf{K}_{jf}^i + \sigma_m^i\sum_{k=1}^{N_r}\lambda_k^r\mathbf{K}_{km}^i + \sigma_f^i\sum_{k=1}^{N_r}\lambda_k^r\mathbf{K}_{kf}^i - \sigma_i^r\sum_{k=1}^{N_r}\lambda_k^i\mathbf{K}_{ki}^i - \sigma_j^r\sum_{k=1}^{N_r}\lambda_k^i\mathbf{K}_{kj}^i$    ▷ Update $< \lambda^r, \mathbf{K}^i\lambda^i >$
end procedure
```

```
procedure 6. SEQUENTIAL SINGLE SEARCHING 2-D SOLVER WITHOUT DAMPING
    Step 1. Search for two optimization variables based on single direction solver
    for $i = 1 : N_r$ do
        calculate $\bigtriangledown\theta_i^r(\bigtriangledown\theta_i^i)$        ▷ Based on single direction solver IV-A
    end for
    choose the dual variable with first and the second largest gain of sub objective function, denoted as 1st and 2nd
    Step 2. Update 1st and 2nd optimization variables based on double direction solver
    update $\lambda_{1st}^r(\lambda_{1st}^i)$ and $\lambda_{2nd}^r(\lambda_{2nd}^i)$        ▷ Based on double direction solver IV-B
    update $\Phi^r(\Phi^i)$ and $\Psi^r(\Psi^i)$ by Procedure 1 and Procedure 2
end procedure
```

```
procedure 7. SEQUENTIAL SINGLE SEARCHING 2-D SOLVER WITH DAMPING
    Step 1. Search for two optimization variables based on single direction solver
    for $i = 1 : N_r$ do                                        ▷ First round searching
        calculate $\bigtriangledown\theta_i^r(\bigtriangledown\theta_i^i)$        ▷ Based on single direction solver IV-A
    end for
    choose the optimization variable with the largest gain of objective function as $1st_1$
    update $\Phi^r(\Phi^i)$ and $\Psi^r(\Psi^i)$ with respect to $1st_1$
    for $i = 1 : N_r$ do                                        ▷ Second round searching
        calculate $\bigtriangledown\theta_i^r(\bigtriangledown\theta_i^i)$        ▷ Based on single direction solver IV-A
    end for
    choose the optimization variable with the largest gain of objective function as $1st_2$
    Step 2. Update $1st_1$ and $1st_2$ optimization variables based on double direction solver
    update $\lambda_{1st_1}^r(\lambda_{1st_1}^i)$ and $\lambda_{1st_2}^r(\lambda_{1st_2}^i)$        ▷ Based on double direction solver IV-B
    update $\Phi^r(\Phi^i)$ and $\Psi^r(\Psi^i)$ by Procedure 1 and Procedure 2
end procedure
```

VI. COMPUTER SIMULATIONS

Computer simulation is launched to test the detection and run time performance of proposed dual channel complex support vector detection algorithm. For sake of brevity, the real case is tested first, all the experiments are made by C, compiled by gcc version 4.8.3 on 64 bit Fedora (release 19) Linux system. The experiment platform is a desktop computer with I5-4th generation CPU with quad processing cores, 3.2 GHz clock rate, 8 GB RAM.

For sake of brevity, we consider a real uncoded spatial multiplex large MIMO system to simulate one channel of the proposed dual channel complex support vector detection algorithm. with N_r received antennas and N_t transmitted antennas. The propagation channel matrix is constructed by channel gain components that are identically independent distributed (i.i.d) Gaussian random variables with zero mean and unit variance. transmitted symbols are mutually independent modulated by M PAM with normalized average energy $\frac{1}{N_t}$, transmitted over flat fading channel, the sample of noise is AWGN with zero mean and variance $\frac{1}{10^{SNR/10}}$, where SNR denotes the signal to noise ratio. We make experiment to low loading factor system 100×40 and full loading factor 100×100, with at least $1e^5$ channel realizations and at least 500 symbol errors accumulated. Fig.3 shows the symbol error rate (SER) performance, Table.I shows the average iteration time of real SVD for different SNR.

TABLE I
AVERAGE ITERATION TIME OF REAL SUPPORT VECTOR DETECTOR

Array Size	**SNR**								
	6	8	10	12	14	16	18	20	22
100×40	682	681	681	681	680	679	678	677	680
100×100	1925	1916	1903	1885	1862	1827	1782	1723	1654

APPENDIX A

PSEUDO CODE OF CSVD

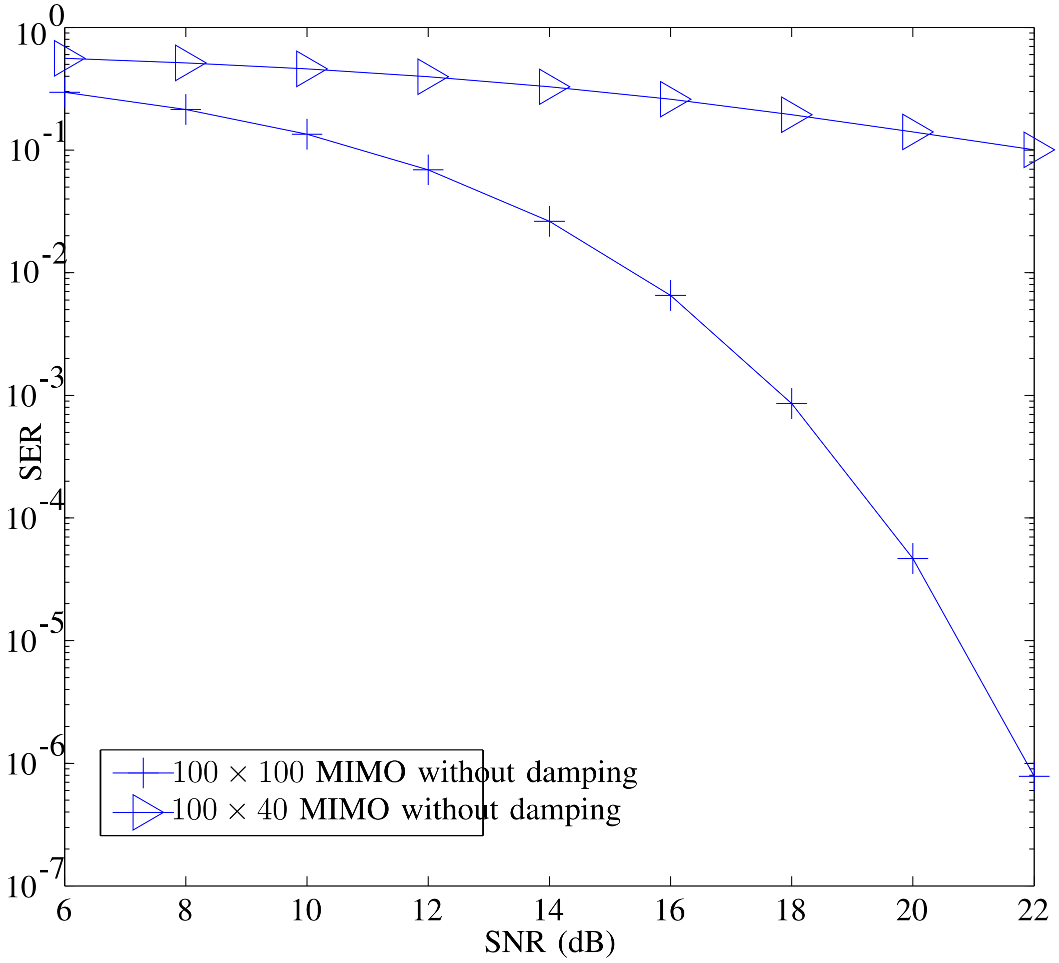

Fig. 3. SER performance of 100 × 100 and 100 × 40 MIMO system

References

[1] "IEEE Standard for Information technology– Telecommunications and information exchange between systemslocal and metropolitan area networks– Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz." *IEEE Std 802.11ac-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012)*, pp. 1–425, Dec 2013.

Algorithm 1 Dual Channel Complex Support Vector Detection Algorithm

```
procedure CSVD(y,H)
    Step 1. Initialization
    $\mathbf{K} = \mathbf{H}\mathbf{H}^H$    ▷ kernel matrix
    $\chi^r = 0$, $\chi^i = 0$    ▷ risk function
    for $i = 1 : N_r$ do    ▷ initialize $\lambda^r$, $\lambda^i$, $\Phi^r$, $\Phi^i$, $\Psi^r$, $\Psi^i$ and duality gap $G$
        $\lambda_i^r = 0, \lambda_i^i = 0$
        $\Phi_i^r = Re(y_i), \Phi_i^i = Im(y_i)$
        $\Psi_i^r = 0, \Psi_i^i = 0$
        if $|\Phi_i^r| > \epsilon$ then
            $\chi^r + = (|\Phi_i^r| - \epsilon)^2$
        end if
        if $|\Phi_i^i| > \epsilon$ then
            $\chi^i + = (|\Phi_i^i| - \epsilon)^2$
        end if
    end for
    $G = C(\chi^r + \chi^i)$    ▷ initialize duality gap
    $\theta = -0.5G$    ▷ initialize objective function
    Step 2. if $G > tol$, go to step 3, else go to Step 5
    Step 3.
    Sequentia single searching 2-D solver with or without damping    ▷ find two optimization variables to be updated
    Step 4. Procedure 5 update $G$
    Step 5.
    $\tilde{x} = (\lambda^r + i\lambda^i)^T\mathbf{H}$    ▷ reconstruct $\mathbf{x}$
    $\mathbf{x} = \mathbb{Q}(\tilde{x})$    ▷ $\mathbb{Q}(\cdot)$ denotes quantization operation based on symbol constellation
    go back to Step 2
    Step 6. Return $\mathbf{x}$
end procedure
```

[2] E. Dahlman, S. Parkvall, J. Skold, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*. Academic press, 2010.

[3] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, “Scaling up MIMO: Opportunities and challenges with very large arrays,” *Signal Processing Magazine, IEEE*, vol. 30, no. 1, pp. 40–60, 2013.

[4] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta, “Massive MIMO for next generation wireless systems,” *Communications Magazine, IEEE*, vol. 52, no. 2, pp. 186–195, 2014.

[5] P. W. Wolniansky, G. J. Foschini, G. Golden, R. Valenzuela *et al.*, “V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel,” in *Signals, Systems, and Electronics, 1998. ISSSE 98. 1998 URSI*

International Symposium on. IEEE, 1998, pp. 295–300.

[6] G. J. Foschini, G. D. Golden, R. Valenzuela, P. W. Wolniansky *et al.*, “Simplified processing for high spectral efficiency wireless communication employing multi-element arrays,” *Selected Areas in Communications, IEEE Journal on*, vol. 17, no. 11, pp. 1841–1852, 1999.

[7] J. Benesty, Y. Huang, and J. Chen, “A fast recursive algorithm for optimum sequential signal detection in a blast system,” *Signal Processing, IEEE Transactions on*, vol. 51, no. 7, pp. 1722–1730, 2003.

[8] M. O. Damen, H. El Gamal, and G. Caire, “On maximum-likelihood detection and the search for the closest lattice point,” *Information Theory, IEEE Transactions on*, vol. 49, no. 10, pp. 2389–2402, 2003.

[9] J. Jaldén and B. Otterste, “On the complexity of sphere decoding in digital communications,” *Signal Processing, IEEE Transactions on*, vol. 53, no. 4, pp. 1474–1484, 2005.

[10] L. G. Barbero and J. S. Thompson, “Fixing the complexity of the sphere decoder for MIMO detection,” *Wireless Communications, IEEE Transactions on*, vol. 7, no. 6, pp. 2131–2142, 2008.

[11] Z. Luo, M. Zhao, S. Liu, and Y. Liu, “Generalized parallel interference cancellation with near-optimal detection performance,” *Signal Processing, IEEE Transactions on*, vol. 56, no. 1, pp. 304–312, 2008.

[12] D. Radji and H. Leib, “Interference cancellation based detection for v-blast with diversity maximizing channel partition,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 3, no. 6, pp. 1000–1015, 2009.

[13] K. V. Vardhan, S. K. Mohammed, A. Chockalingam, and B. S. Rajan, “A low-complexity detector for large MIMO systems and multicarrier CDMA systems,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 3, pp. 473–485, 2008.

[14] P. Li and R. D. Murch, “Multiple output selection-LAS algorithm in large MIMO systems,” *Communications Letters, IEEE*, vol. 14, no. 5, pp. 399–401, 2010.

[15] N. Srinidhi, T. Datta, A. Chockalingam, and B. S. Rajan, “Layered tabu search algorithm for large-MIMO detection and a lower bound on ML performance,” *Communications, IEEE Transactions on*, vol. 59, no. 11, pp. 2955–2963, 2011.

[16] T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan, “Random-restart reactive tabu search algorithm for detection in large-mimo systems,” *Communications Letters, IEEE*, vol. 14, no. 12, pp. 1107–1109, 2010.

[17] P. Som, T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan, “Low-complexity detection in large-dimension MIMO-ISI channels using graphical models,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 8, pp. 1497–1511, 2011.

[18] P. Som, T. Datta, A. Chockalingam, and B. S. Rajan, “Improved large-MIMO detection based on damped belief propagation,” in *Information Theory Workshop (ITW), 2010 IEEE*. IEEE, 2010, pp. 1–5.

[19] T. L. Narasimhan and A. Chockalingam, “Channel hardening-exploiting message passing (CHEMP) receiver in large-scale MIMO systems,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 8, no. 5, pp. 847–860, 2014.

[20] J. Goldberger and A. Leshem, “MIMO detection for high-order QAM based on a Gaussian tree approximation,” *Information Theory, IEEE Transactions on*, vol. 57, no. 8, pp. 4973–4982, 2011.

[21] S. Mohammed, A. Chockalingam, and B. S. Rajan, “Low-complexity near-map decoding of large non-orthogonal stbcs using pda,” in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*. IEEE, 2009, pp. 1998–2002.

[22] T. Datta, N. A. Kumar, A. Chockalingam, and B. S. Rajan, “A novel monte-carlo-sampling-based receiver for large-scale uplink multiuser MIMO systems,” *Vehicular Technology, IEEE Transactions on*, vol. 62, no. 7, pp. 3019–3038, 2013.

[23] Q. Zhou and X. Ma, “Element-based lattice reduction algorithms for large MIMO detection,” *Selected Areas in Communications, IEEE Journal on*, vol. 31, no. 2, pp. 274–286, 2013.

[24] V. Vapnik, “Pattern recognition using generalized portrait method,” *Automation and remote control*, vol. 24, pp. 774–780, 1963.

[25] V. Vapnik and A. Chervonenkis, “A note on one class of perceptrons,” *Automation and remote control*, vol. 25, no. 1, 1964.

[26] B. Schölkopf and A. J. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[27] B. E. Boser, I. M. Guyon, and V. N. Vapnik, “A training algorithm for optimal margin classifiers,” in *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 1992, pp. 144–152.

[28] I. Guyon, B. Boser, and V. Vapnik, “Automatic capacity tuning of very large vc-dimension classifiers,” *Advances in neural information processing systems*, pp. 147–147, 1993.

[29] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

[30] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[31] B. Schölkopf, C. Burges, and V. Vapnik, “Incorporating invariances in support vector learning machines,” in *Artificial Neural NetworksICANN 96*. Springer, 1996, pp. 47–52.

[32] V. Vapnik, S. E. Golowich, and A. Smola, “Support vector method for function approximation, regression estimation, and signal processing,” in *Advances in Neural Information Processing Systems 9*. Citeseer, 1996.

[33] A. J. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[34] J. Platt *et al.*, “Fast training of support vector machines using sequential minimal optimization,” *Advances in kernel*

methodssupport vector learning, vol. 3, 1999.

[35] E. Osuna, R. Freund, and F. Girosi, "An improved training algorithm for support vector machines," in *Neural Networks for Signal Processing [1997] VII. Proceedings of the 1997 IEEE Workshop*. IEEE, 1997, pp. 276–285.

[36] I. Steinwart, D. Hush, and C. Scovel, "Training svms without offset," *The Journal of Machine Learning Research*, vol. 12, pp. 141–202, 2011.

[37] P. Bouboulis, S. Theodoridis, C. Mavroforakis, and L. Evaggelatou-Dalla, "Complex support vector machines for regression and quaternary classification."

[38] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[39] P. Bouboulis and S. Theodoridis, "Extension of Wirtinger's calculus to reproducing kernel Hilbert spaces and the complex kernel LMS," *Signal Processing, IEEE Transactions on*, vol. 59, no. 3, pp. 964–978, 2011.