CHAPTER 5

Low Rank Channel Estimation in FDD Mode

5.1 Introduction

In TDD mode, CSI acquired in the uplink may not be accurate for the downlink due to the calibration error of radio frequency chains and limited coherence time. FDD systems can provide more efficient communications with low latency. In FDD systems, CSI is obtained at every user by sending the pilot signal from BS and estimate the channel information with the help of pilot signal. The obtained CSI is fed back to the BS for precoding the user data.

The number of orthogonal pilots required for downlink channel estimation is proportional to the number of BS antennas, while the number of orthogonal pilots required for uplink channel estimation is proportional to the number of scheduled users. To estimate the downlink channel, the pilot overhead is in the order of a number of BS antennas which is prohibitively large in Massive MIMO system. Further, the estimated CSI by the user is feedback to the BS over the uplink channel. Hence, the overall overhead for uplink is high. Therefore, it is of importance to explore channel estimation in the downlink than that in the uplink, which can facilitate massive MIMO to be backward compatible with current FDD dominated cellular networks. Hence, it is necessary to explore channel estimation method for massive MIMO based on FDD mode with reduced overhead.

In this chapter, the channel is modeled for downlink and uplink FDD mode transmission is discussed in Section 5.2. In downlink propagation model, rich scattering is considered at the user side and most clusters are around BS. All users in the cell are accessible to cluster at BS leads to same steering matrix which introduces correlation among the users. Hence, the high dimensional downlink channel matrix is likely to approximate as low rank, where as in uplink, rich
scattering at user side approximates the channel as high dimensional i.i.d matrix. The channel estimation method for downlink is carried out at BS is presented in section 5.3. In Section 5.4, the convergence results for SVP-G, SVP-H, SVP-H are compared with the proposed WNN method based on the mean square error at different SNR levels are presented.

5.2 System and Channel Model

Consider the downlink FDD massive MIMO system with $M$ transmit antenna at BS, serving $K$ single receiver antenna user as shown in Fig.5.1. The BS transmits pilot $\phi_t \in \mathbb{C}^{M \times 1}$ at the $t^{th}$ channel use ($t = 1, 2, ...L$). The received pilot signal at the $k^{th}$ user is $y_k \in \mathbb{C}^{1 \times L}$ during $L$ channel use can be expressed as

$$y_k = h_k \Phi + n_k$$

(5.1)

where $\Phi = [\phi_1, \phi_2, ...\phi_L]$ is a $M \times L$ training matrix constructed from the transmitted pilots during $T$ channel use. $n_k \in \mathbb{C}^{1 \times L}$ represents the i.i.d additive white Gaussian noise with elements having zero mean and variance $\sigma_n^2$. The channel
vector \( h_k \in \mathbb{C}^{1 \times M} \) between the BS and the \( k^{th} \) user is given by

\[
h_k = \sum_{p=1}^{P} g_{k,p} a(\theta_p)
\]

(5.2)

where \( P \) is the number of scatterers or number of resolvable physical paths, \( \theta_p \) is the Angle of Departure (AoD) of the \( p^{th} \) path. For uniform linear antenna array the steering vector is defined as

\[
a(\theta_p) = [1, e^{-j2\pi \frac{D}{\lambda} \cos(\theta_p)}, \cdots, e^{-j2\pi \frac{D}{\lambda}(M-1)\cos(\theta_p)}]
\]

(5.3)

where \( D \) and \( \lambda \) denote the antenna spacing at the BS and carrier wavelength respectively.

In channel model, rich scattering is considered at the user side and most clusters are around BS. The clusters that are present around the BS are accessible to all users introduce correlation among the users, even when the users are geographically apart. Hence, the channel vectors associated with different users have the same steering vectors. Thus, the downlink channel matrix is given as

\[
H = GA
\]

(5.4)

where \( G \in \mathbb{C}^{K \times P} \) is the path gain matrix and \( A = [a(\theta_1)^T, a(\theta_2)^T, \cdots, a(\theta_P)^T] \in \mathbb{C}^{P \times M} \). Therefore, \( \text{rank}(H) \leq \min\{P, K, M\} \). Usually, \( M \) and \( K \) are large for massive MIMO system and the number of scatterers is assumed relatively small then the \( \text{rank}(H) \leq \min\{P\} \). Therefore high dimensional downlink channel matrix is approximated as a low rank channel.

### 5.3 Downlink Channel Estimation

In conventional FDD system, the channel vector for each user \( h_k \) \( (k = 1, 2, \cdots, K) \) is estimated individually and then the estimated CSI is fed back to the BS. In this thesis we have assumed, instead of estimating the channel vector at the user side, the observed pilot signal by each user is fed back to the BS. The joint MIMO channel estimation of all user is done at the BS. The pilot observation of all user
is expressed as

$$Y = H\Phi + N$$  \hspace{1cm} (5.5)$$

where $Y \in \mathbb{C}^{K \times L}$, $H = [h_1^T, h_2^T, \ldots, h_K^T]^T \in \mathbb{C}^{K \times M}$ is the downlink channel to be recovered and $N = [n_1^T, n_2^T, \ldots, n_K^T]^T \in \mathbb{C}^{K \times L}$ is the downlink noise matrix.

The pilot signal $W$ which is fed back to the BS by all users is given as

$$W = QY + Z$$  \hspace{1cm} (5.6)$$

where $Q \in \mathbb{C}^{M \times K}$ is the uplink channel matrix which is modelled as Rayleigh fading matrix whose entries i.i.d random variable with zero-mean and $\sigma^2$ variance. $Z \in \mathbb{C}^{M \times L}$ is the uplink noise matrix whose entries follows $\mathcal{CN}(0, \sigma^2_z)$.

To recover the downlink channel matrix at BS, firstly $Y$ has to be estimated. $Y$ matrix is estimated using LS estimation by assuming, uplink channel matrix $Q$ is known. The estimate $\hat{Y}$ is given as

$$\hat{Y} = (Q^HQ)^{-1}Q^HW$$  \hspace{1cm} (5.7)$$

Further, the estimation of downlink channel matrix at BS can be formulated as a rank minimization problem:

$$\min_{H} \text{rank}(H) \quad \text{s.t.} \quad \hat{Y} = H\Phi$$  \hspace{1cm} (5.8)$$

This rank minimization problem is nonconvex and NP hard. The above problem can be reformulated, when rank of the matrix is known as [69]

$$\min_{H} J(h) = ||\hat{Y} - \Psi h||^2_2 \quad \text{s.t.} \quad \text{rank}(H) \leq r$$  \hspace{1cm} (5.9)$$

The solution to the minimization problem is obtained iteratively using Singular Value Projection (SVP) algorithm. In SVP algorithm, channel matrix can also be iteratively updated using Newton’s method called SVP-N and the search direction is $\nabla^2 J(h)^{-1}\nabla J(h)$. The optimal step size $\lambda^i$ is chosen by minimizing the cost function $J$.

$$\lambda^i_N \equiv \min_t \{J(h^{i-1} + t\nabla^2 J(h)^{-1}\nabla J(h))\}$$  \hspace{1cm} (5.10)$$
Taking the derivative of the cost function and equating to zero the optimal step size $\lambda_N^i$ obtained is

$$\lambda_N^i = t = \frac{\nabla J(h^{i-1})^T (2\Psi^H\Psi)^{-1} \nabla J(h^{i-1})}{((2\Psi^H\Psi)^{-1} \nabla J(h^{i-1}))^T (2\Psi^H\Psi)(2\Psi^H\Psi)^{-1} \nabla J(h^{i-1})} \quad (5.11)$$

simplifying the equation

$$\lambda_N^i = -1 \quad (5.12)$$

The channel update matrix at $i^{th}$ iteration is given by

$$h(i + 1) = h(i) + \lambda_N^i \nabla^2 J(h)^{-1} \nabla J(h) \quad (5.13)$$

substituting the optimal step size and newton search direction, the above equation simplifies to

$$h = (\Psi^H\Psi)^{-1} \Psi^H \hat{y} \quad (5.14)$$

Hence, with the Newton search direction, the channel updating equation converges in one iteration. To get the low rank solution, the updated channel matrix is projected on to the low-rank matrix constraint set. The projection of the matrix to the low-rank matrix is done using SVD. Therefore, the SVP-N algorithm gives the low rank solution in two steps.

\underline{Algorithm} : Channel Estimator using SVP-N algorithm

1: Input $M$, $K$, $L$, $\Phi$, $\hat{Y}$, $\alpha$, $r$

2: \hspace{1cm} $\Psi = \Phi^T \otimes I_M$

3: \hspace{1cm} $h \leftarrow (\Psi^H\Psi)^{-1} \Psi^H \hat{y}$

4: \hspace{1cm} $H = \text{unvec}(h)$

5: \hspace{1cm} $[U \Sigma V] = SVD(H)$

6: \hspace{1cm} $H \leftarrow U(:, 1 : r) \Sigma(1 : r, 1 : r)V(:, 1 : r)^H$
**Complexity Order:** The computational complexity lies in calculation of SVD of the $M \times K$ matrix of rank $r$ is $O(M^2r)$ and matrix-vector multiplication in step has a complexity of $O((ML)(MK))$. The total complexity of the SVP-N algorithm is $O(M^2r + (ML)(MK))$

In SVP-N, SVD operation is used only once and hence the error variance will be more. In SVP algorithm, channel matrix can also be updated using Gradient decent method (SVP-G) i.e search direction is the gradient of the cost function $\nabla J(h) = 2\Psi^H(\Psi h - \hat{y})$. The optimal step size $\lambda^i_G$ is chosen to minimize the cost function $J$ is given as

$$\lambda^i_G = \min_t \{ J(h^{i-1} + t\nabla J(h^i)) \} \quad (5.15)$$

Solving the equation, the optimal step size $\lambda^i_G$ obtained is

$$\lambda^i_G = t = -\frac{\nabla J(h^{i-1})^T \nabla J(h^{i-1})}{\nabla J(h^{i-1})^T (2\Psi^T\Psi) \nabla J(h^{i-1})} \quad (5.16)$$

Therefore, the SVP-G algorithm for the channel estimation problem consists of two steps: (i) channel updating matrix (ii) SVD operation to obtain the low-rank solution. These two steps solved iteratively are shown below:

---

**Algorithm**: Channel Estimator using SVP-G algorithm

1: **Input** $M, K, L, \Phi, \hat{Y}, \alpha, r$

2: **Initialization**: $h(1) = 0, \Psi = \Phi^T \otimes I_M, i = 1.$

3: **repeat**

4: $h(i + 1) \leftarrow h(i) + 2\lambda^i_G \Psi^H(\Psi h(i) - \hat{y})$

5: $H(i+1) = \text{unvec}(h(i+1))$

6: $[U\Sigma V] = SV D(H(i + 1))$

7: $H(i + 1) \leftarrow U(:, 1 : r)\Sigma(1 : r, 1 : r)V(:, 1 : r)^H$
8: \( h(i+1) = \text{vec}(H(i+1)) \)
9: \( i = i+1 \)
10: until maximum number of iteration reached

**Complexity Order:**

The computational complexity lies in calculation of SVD of the \( M \times K \) matrix of rank \( r \) is \( \mathcal{O}(M^2r) \) and matrix-vector multiplication in step has a complexity of \( \mathcal{O}((ML)(MK)) \). The total complexity of the SVP-G algorithm is \( \mathcal{O}(\text{iter}(M^2r + (ML)(MK))) \).

The SVP-G algorithm takes a longer time to converge but gives minimum error variance compared to SVP-N. Hence in [69], the authors combined the advantage of SVP-G and SVP-N and proposed SVP-Hybrid (SVP-H) algorithm. In SVP-H, SVP-N is used in the first iteration to have fast convergence and SVP-G is used in the rest of the iteration to have minimum error variance compared to SVP-N. SVP-H algorithm for the channel estimation problem is given below:

**Algorithm :** Channel Estimator using SVP-H algorithm

1: Input \( M, K, L, \Phi, \hat{Y}, \alpha, r \)
2: Initialization: \( H(1) = \text{rand}(K,M), h(1) = \text{vec}(H(1)) \)
\( Hq(1) = \text{SVD}_r(H(1)), hq(1) = \text{vec}(Hq(1)), i = 1. \)
3: repeat
4: if \( i = 1 \)
5: \( \lambda(i) = \lambda_N(i), d(i) = d_N(i) \)
6: else
7: \( \lambda(i) = \lambda_G(i), d(i) = d_G(i) \)
8: end
9: \( h(i + 1) \leftarrow hq(i) + \lambda(i)d(i) \)
10: \( H(i+1) = \text{unvec}(h(i+1)) \)
11: \([U\Sigma V] = SVD(H(i+1))\)
12: \( H(i + 1) \leftarrow U(:, 1 : r)\Sigma(1 : r, 1 : r)V(:, 1 : r)^H \)
13: \( hq(i+1) = \text{vec}(H(i+1)) \)
14: \( i = i + 1 \)
15: until maximum number of iteration reached

**Complexity Order:**

The computational complexity lies in calculation of SVD of the \( M \times K \) matrix of rank \( r \) is \( \mathcal{O}(M^2r) \) and matrix-vector multiplication in step has a complexity of \( \mathcal{O}((ML)(MK)) \). The total complexity of the SVP-H algorithm is \( \mathcal{O}(\text{iter}(M^2r + (ML)(MK))) \).

In the algorithm, \( d_N \) and \( d_G \) are the search direction for Newton and gradient method. In all these algorithms the singular value of the estimated channel matrix is equal to the singular value of the original channel matrix plus the singular value of the noise matrix. Hence at lower SNR, the error variance will be more compared to the error variance at higher SNR. Therefore, SVP-H gives minimum error variance only at high SNR. To overcome the above issue, that is to maintain minimum variance at all SNR, IWSVT algorithm is used and the corresponding optimization problem is

\[
\min_{\mathbf{H}} \|\mathbf{H}\|_{w,*} \quad \text{s.t.} \quad \hat{\mathbf{y}} = \mathbf{Ψh} \quad (5.17)
\]

In order to speed up the convergence, FIWSVT algorithm is used for non-orthogonal training sequence. The proposed algorithm for the channel estimation problem is given below:
**Algorithm**: Channel Estimator using FIWSVT algorithm

1. **Input** $M, K, L, \Phi, \hat{Y}, \lambda, \alpha, r$

2. **Initialization**: $H_d(1) = 0, H(1) = 0, \Psi = \Phi^T \otimes I_M, W_i = I, t_1 = 0$, $i = 1$.

3. **repeat**
   4. $A \leftarrow H_d(i) + \frac{1}{\alpha} vec^{-1}_{M,K}(\Psi^H vec(\hat{Y} - H_d(i)\Phi))$
   5. $[U \Sigma V] = SVD(A)$
   6. Thresholding: $\Sigma_i = \text{Diag}(\sigma_i - \lambda w_i)$
   7. $H(i) \leftarrow U(:, 1 : r) \Sigma_i (1 : r, 1 : r) V(:, 1 : r)^H$
   8. $t_{i+1} = \frac{1 + \sqrt{1 + 4t_i^2}}{2}$
   9. $H_d(i + 1) = H(i) + (\frac{t_{i+1}}{t_i})(H(i) - H(i - 1))$
   10. $i \leftarrow i + 1$
   11. Update $W_i$
   12. **until** condition satisfied or maximum number of iteration reached

13. **Output**: $H_d$

**Complexity Order:**
The computational complexity lies in calculation of SVD of the $M \times K$ matrix of rank $r$ is $O(M^2 r)$ and matrix-vector multiplication in step has a complexity of $O((ML)(MK))$. The total complexity of the FIWSVT algorithm is $O(\text{iter}(M^2 r + (ML)(MK)))$. 

85
5.4 Simulation Results and Discussion

In this section, the WNN channel estimator for FDD system is evaluated based on the normalized MSE performance index for the nonorthogonal training sequence. The parameters of the single cell massive MIMO system for simulation is given in Table 5.1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of BS Antennas ($M$)</td>
<td>60</td>
</tr>
<tr>
<td>Number of users in a cell ($K$)</td>
<td>20</td>
</tr>
<tr>
<td>Number of scatterers ($P$)</td>
<td>10</td>
</tr>
<tr>
<td>Rank of the matrix ($r$)</td>
<td>6</td>
</tr>
<tr>
<td>Length of the training data ($L$)</td>
<td>70</td>
</tr>
<tr>
<td>Antenna Spacing ($D/\lambda$)</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Table 5.1: System Parameters

Figure 5.2: Normalized MSE Vs Number of iteration (SNRd=10 dB, SNRu=15 dB)
Fig. 5.2 and 5.3 show the convergence analysis of the algorithm SVP-N, SVP-G, SVP-H and FIWSVT algorithm for downlink with SNR (SNRd) fixed as 10 dB and uplink SNR (SNRu) is varied for 15 dB and 20 dB. It is observed from the figure that FIWSVT algorithm gives minimum NMSE value compared to all other variants of SVP algorithm. It is noted that FIWSVT algorithm reaches the steady state error faster than SVP-G. However, SVP-N and SVP-H algorithms converge faster with high estimation error. As uplink SNR increases, FIWSVT takes more iteration to reach steady state with minimum NMSE.

Fig. 5.4 and 5.5 shows the scenario, where the SNR of the uplink is varied for 10 dB and 20 dB by keeping downlink SNR as 15 dB. Similar trend is observed in NMSE performance for FIWSVT and SVP variants. SVP-G and FIWSVT provide minimum NMSE value compared to other two algorithms but takes more iteration to converge compared to SVP-N and SVP-H. As downlink SNR value increase, FIWSVT algorithm takes more number of iteration to converge as uplink SNR value increase which is shown in Fig. 5.6, 5.7 and 5.8.

The NMSE performance of FIWSVT and FISVT algorithms for different
Figure 5.4: Normalized MSE Vs Number of iteration (SNRd=15 dB, SNRu=10 dB)

Figure 5.5: Normalized MSE Vs Number of iteration (SNRd=15 dB, SNRu=20 dB)
Figure 5.6: Normalized MSE Vs Number of iteration (SNRd=25 dB,
SNRu=10 dB)

Figure 5.7: Normalized MSE Vs Number of iteration (SNRd=25 dB,
SNRu=15 dB)
Figure 5.8: Normalized MSE Vs Number of iteration (SNRd=25 dB, SNRu=30 dB)

Figure 5.9: Normalized MSE Vs Uplink SNR (downlink SNR =15 dB)
uplink SNR is simulated by fixing the downlink SNR as constant. Fig.5.9 and Fig.5.10 show NMSE versus uplink SNR for downlink SNR 15 dB and 25 dB respectively. From the response, FIWSVT shows minimum NMSE compared to FISVT algorithm for both downlink SNR.

5.5 Summary

In this chapter, downlink FDD channel is modeled as a low-rank channel by considering most of the clusters are around BS and rich scattering at the user side. Instead of estimating the downlink channel at the user side, the received pilot signal of the user is sent back to the BS and downlink channel matrix is estimated at BS under the assumption that uplink channel matrix is known. The received pilot signal of the users is estimated using LS method. The downlink channel estimation problem is studied for nonorthogonal training sequence using FIWSVT algorithm when the rank of the matrix is known. The convergence and NMSE of the FIWSVT algorithm are compared with SVP-G, SVP-N, and SVP-H. It is
shown through simulation, FIWSVT algorithm has minimum NMSE and faster convergence at low SNR compared to other algorithms whereas, at high SNR, it shows minimum NMSE same as SVP-H but with more number of iteration compared to SVP-H.