Sequential Convex Programming for Full-Duplex Single-User MIMO Systems

Sean Huberman and Tho Le-Ngoc
Department of Electrical and Computer Engineering, McGill University, 3480 University Street, Montreal, Quebec, Canada, H3A 2A7
Email: sean.huberman@mail.mcgill.ca ; tho.le-ngoc@mcgill.ca

Abstract—This paper proposes two Sequential Convex Programming (SCP) algorithms, namely Difference of Convex functions (DC)-based and Sequential Convex Approximations for Matrix-variable Programming (SCAMP), for solving the non-convex matrix-variable sum-rate maximization problem in Full-Duplex (FD) Single-User Multiple-Input-Multiple-Output (SU-MIMO) systems. The two proposed algorithms result in different approximations of the objective function and hence, depending on the environment, one may be favorable than the other. Numerical results show that SCP can significantly increase the sum-rate over existing techniques for the SU-MIMO scenario. In particular, for the SU-MIMO scenario, the DC-based algorithm outperforms the SCAMP.

Index Terms—Full duplex, MIMO, sequential convex programming, non-convex optimization.

I. INTRODUCTION

Existing wireless communication systems operate using Half-Duplex (HD) transmission. For HD systems, nodes communicate in separate time or frequency slots to avoid interference. One candidate approach for increasing the spectral efficiency of wireless networks is Full-Duplex (FD) transmission by allowing simultaneous transmission/reception on the same frequency.

FD systems can potentially have a capacity “double” that of their HD counterparts. However, FD transmission incurs additional interference, known as self-interference, from the transmitter to the receiver of the same node. Typically, the self-interference is significantly larger than the received signal strength, which can prevent the potential “double” sum-rate gains of FD transmission, and, hence, calls for effective self-interference management. The FD approach lends itself well towards the concept of small cells (e.g., pico- or femto-cells) since for shorter ranges, the received Signal-to-self-Interference Ratio (SIR) is larger.

Most existing literature focuses on self-interference suppression for FD relays or Single-User Multiple-Input-Multiple-Output (SU-MIMO) networks (e.g., [1]–[4]). Various techniques for mitigating the effects of self-interference for FD relays, including time-domain cancellation and three spatial suppression techniques were presented in [1]. Most proposed techniques only focus on the suppression of self-interference without regards for the effect on the forward channel.

A Multi-User MIMO (MU-MIMO) system where BSs operate in FD-mode while MSs operate in HD-mode was investigated in [5]. The joint precoding scheme in [5] had the form of a Difference of Convex functions (DC), where a first-order approximation to the objective function was applied in order to convexify the optimization problem. It was assumed in [5] that the downlink and uplink users are geographically separated, and hence, the inter-user-interference was ignored. This assumption is equivalent to assuming that Node 2 in Fig. 1 experiences no self-interference.

In our previous work [6], we proposed a Self-Interference Pricing (SIP) based approach to replace the direct non-convex optimization problem by four pricing-based sub-problems to consistently provide a favorable trade-off between forward channel maximization and self-interference suppression.

The FD SU-MIMO sum-rate maximization problem formulation leads to a non-convex optimization problem for which finding solutions is difficult. Sequential Convex Programming (SCP) [7] is a well-known practical approach to solving non-convex optimization problems by constructing and solving a sequence of convex optimization problems.

In this paper, we propose two SCP-based algorithms for solving the non-convex sum-rate maximization problem directly. The first algorithm takes advantage of the DC structure of the non-convex optimization problem by directly looking at the objective function decomposition. The form of the original objective function naturally lends itself to this choice of DC decomposition. This approach is similar to the joint precoding scheme presented in [5], except that it considers self-interference at both nodes.

The second algorithm, called Sequential Convex Approximations for Matrix-variable Programming (SCAMP), is developed for solving general non-convex matrix-variable optimization problems with logarithmic objective functions. It approximates the objective function by first applying a lower-bound and then applying an upper-bound on one of the terms in the lower-bound. The result is a non-convex approximation of the original non-convex objective function, which can be more easily convexified. First-order approximations are applied to the non-convex terms of the approximation function in order to ensure that the approximate objective function satisfies the Disciplined Convex Programming (DCP) ruleset [8].

The SCAMP and DC-based approaches result in different objective function approximations. Hence, depending on the environment, one approach may be more favorable than the other. In this paper, we apply the SCAMP and DC-based approaches to the FD SU-MIMO system and compare their...
performance with respect to the SIP and optimized HD approaches.

The remainder of this paper is organized as follows: Section II describes the FD SU-MIMO system model. Section III presents the proposed DC-based and SCAMP algorithms applied to the FD SU-MIMO scenario. Section IV provides illustrative results, and Section V provides concluding remarks.

Notation: In this work, non-bold variables denote scalars, lower-case bold variables denote vectors, and upper-case bold variables denote matrices. \( E \{ \cdot \} \) refers to the expected value operation. \( A^\dagger \) refers to the conjugate transpose of matrix \( A \). \( \text{Tr} \{ A \} \) refers to the trace of matrix \( A \). \( |A| \) refers to the determinant of matrix \( A \). \( I_d \) and \( O_d \) refer to the \( d \times d \) identity and all-zero matrices, respectively. Finally, \( A \succ 0 \) implies that \( A \) is a positive semi-definite matrix.

II. SYSTEM MODEL

The FD SU-MIMO system model is shown in Fig. 1. Let each node be equipped with \( N_T \) transmit and \( N_R \) receive antennas, respectively. Let \( H_j \in \mathbb{C}^{N_R \times N_T} \) be the channel of the \( j \)-th node to the \( i \)-th node (\( i \neq j \)). Similarly, let \( G_i \in \mathbb{C}^{N_R \times N_T} \) be the self-interference channel matrix for the \( i \)-th node. Let \( x_j \in \mathbb{C}^{N_T \times 1} \) be the vector of transmitted signals for the \( i \)-th node. Let \( V_j \in \mathbb{C}^{N_T \times N_R} \) be the precoding matrix for the \( j \)-th node. Then, the transmission equation for the signal received at the \( i \)-th node, can be written as:

\[
y_i = H_j V_j x_j + G_i V_i x_i + z_i,
\]

where \( z_i \in \mathbb{C}^{N_T \times 1} \) is the noise at the \( i \)-th node. The first term represents the intended signals, while the second term represents the self-interference incurred by operating in FD mode.

The self-interference channels, \( G_i \ (i = 1, 2) \), are assumed to be estimated, while the forward channels, \( H_i \ (i = 1, 2) \), are assumed to be known perfectly, in order to easily compare with the HD case. More specifically, it is assumed that:

\[
G_i = \hat{G}_i + \Delta G_i,
\]

where \( G_i \) is the true channel matrix, \( \hat{G}_i \) is the estimated channel matrix, and \( \Delta G_i \) is the channel estimation error matrix, with zero mean and variance \( \sigma_{\Delta G}^2 \).

Let the covariance matrices of the direct and self-interference signals at the \( i \)-th receiver be:

\[
C_{i,j} = H_j V_j S_j V_j^H H_j^H,
\]

\[
C_{i,i} = G_i V_i S_i V_i^H G_i^H.
\]

where \( S_i = E \{ x_i x_i^H \} \). The achievable rate at the \( i \)-th node is:

\[
R_i = \log_2 \left| I_{N_R} + \left( \Sigma_i + C_{i,i} \right)^{-1} C_{i,j} \right|
\]

where \( \Sigma_i = E \{ z_i z_i^H \} \). Let the estimated achievable rate at the \( i \)-th node be:

\[
\hat{R}_i = \log_2 \left| I_{N_R} + \left( \hat{\Sigma}_i + \hat{C}_{i,i} \right)^{-1} C_{i,j} \right|
\]

where \( \hat{C}_{i,i} = \hat{G}_i V_i S_i V_i^H G_i^H \).

III. SEQUENTIAL CONVEX PROGRAMMING

This section describes the two proposed SCP algorithms: DC-based and SCAMP. The sum-rate maximization problem can be written as:

\[
\max_{Q_1, Q_2} \hat{R}_1 + \hat{R}_2
\]

subject to:

\[
\text{Tr} \{ Q_i \} \leq P_{\text{max},i}, \quad i = 1, 2,
\]

\[
Q_i \succ 0, \quad i = 1, 2,
\]

where \( Q_i = V_i S_i V_i^H \) and the positive semi-definite constraints ensure the resulting covariance matrices are feasible. Optimization problem (2) is non-convex and hence, difficult to solve directly. In this paper, we apply SCP to approximate the solution of the non-convex optimization problem. Once the optimal covariance matrix is solved for, the corresponding precoding matrices can be recovered using the Cholesky decomposition. In particular, \( Q_i = L_i L_i^H \) and hence, the precoding matrices can be computed as:

\[
V_i = L_i S_i^{-1/2},
\]

A. DC-Based Algorithm

The DC-based algorithm represents the non-convex objective function as a DC (i.e., \( f = g - h \), where \( g \) and \( h \) are convex) and applies a first-order approximation to \( h \) to make the objective function convex. First, the maximization problem (2) is re-written as a minimization problem with objective function: \( f = -\hat{R}_1 - \hat{R}_2 \).

\( f \) is a non-convex function but can be written as in a DC form. In order to write \( f \) in DC form, \( -\hat{R}_1 \) and \( -\hat{R}_2 \) are separately written in DC form.

\[
-\hat{R}_i = g_i - h_i,
\]

where \( g_i \) and \( h_i \) are given by:

\[
g_i = -\log_2 \left| \Sigma_i + G_i Q_i G_i^H + H_j Q_j H_j^H \right|,
\]

\[
h_i = -\log_2 \left| \Sigma_i + \hat{G}_i Q_i \hat{G}_i^H \right|.
\]

Next, the first-order approximation [9, p. 69], (6), is applied to (5) to produce an affine term, \( h_i \).

\[
\log_2 |A + X| \geq \log_2 |A + X_0| + \frac{1}{\ln(2)} \text{Tr} \left( (A + X_0)^{-1} (X - X_0) \right).
\]

Note that this formula can be easily derived using the concept of matrix differentials [10], [11] and the fact that \( \text{vec}(A)^\dagger \text{vec}(B) = \text{Tr} \{ A^\dagger B \} \).
Hence, the affine approximation to (5) is given by:
\[
\tilde{h}_i = -\frac{1}{\ln(2)} \text{Tr} \left[ \left( \Sigma_i + \tilde{G}_i Q_i^{(k)} \tilde{G}_i^\dagger \right)^{-1} \tilde{G}_i \left( Q_i - Q_i^{(k)} \right) \tilde{G}_i^\dagger \right]
- \log_2 \left| \Sigma_i + \tilde{G}_i Q_i^{(k)} \tilde{G}_i^\dagger \right|, 
\]
(7)
where \(Q_i^{(k)}\) refers to the transmit covariance matrix of the \(i\)-th node associated with the \(k\)-th iteration.

Therefore, the objective function \(f_{dc}\) given by:
\[
f_{dc} = g_1 + g_2 - \tilde{h}_1 - \tilde{h}_2
\]
(8)
is a convex function. Hence, \(f_{dc}\) is a convex approximation to the original objective function \(f = -\tilde{R}_1 - \tilde{R}_2\). As such, the non-convex optimization problem (2) can be locally approximated by the convex optimization problem (9).

\[
\min_{Q_1, Q_2} g_1 + g_2 - \tilde{h}_1 - \tilde{h}_2
\]
subject to: \(\text{Tr} \{ Q_i \} \leq P_{\text{max},i}, \ i = 1, 2\), \(Q_i \succeq 0, \ i = 1, 2\).
(9)

The resulting Algorithm 1 iteratively updates the objective function approximation, \(f_{dc}\), and solves the convex approximation until convergence. Note that the convex optimization problem can be solved using cvx, a package for solving disciplined convex programs in Matlab [12], [13].

Algorithm 1: DC-based Algorithm.

Randomly initialize \(Q_1^{(0)}, Q_2^{(0)}\);
Initialize \(k = 0\);
repeat
\[
\text{Update } \tilde{h}_1, \tilde{h}_2 \text{ using (7), using } Q_i^{(k)}, Q_i^{(k)}; \\
\text{Update } f_{dc} \text{ using (8);} \\
\text{Solve (9) for } Q_1^*, Q_2^*; \\
\text{k = k + 1;} \\
\text{Update } Q_i^{(k)} = Q_i^*, i = 1, 2; \\
\text{until } f_{dc} \text{ converges;}
\]
Apply Cholesky decomposition: \(Q_i^* = L_i L_i^\dagger, \ i = 1, 2\);
Solve for \(V_i\) using (3), \(i = 1, 2\).

B. SCAMP Algorithm

First, the objective function is approximated by combining the first-order approximations of the functions \(\log_2 |A + X|\) given by (6) and \(\log_2 |X|\), given by:
\[
\log_2 |X| \geq \log_2 |X_0| + \frac{1}{\ln(2)} \text{Tr} \left[ X_0^{-1} (X - X_0) \right].
\]
(10)
The inverse of a sum of two matrices can be written as [14]:
\[
(A + X_0)^{-1} = X_0^{-1} - X_0^{-1} (I + AX_0^{-1})^{-1} AX_0^{-1}.
\]
(11)
Hence, using (11), (6) can be re-written as:
\[
\log_2 |A + X| \geq \log_2 |A + X_0| + \frac{\text{Tr} \left[ X_0^{-1} (X - X_0) \right]}{\ln(2)} - \frac{\text{Tr} \left[ X_0^{-1} (I + AX_0^{-1})^{-1} AX_0^{-1} (X - X_0) \right]}{\ln(2)}.
\]
(12)
Combining (10) and (12) gives:
\[
\log_2 |A + X| \approx \log_2 |X| + \frac{1}{\ln(2)} \text{Tr} [\Phi X] + \beta,
\]
(13)
where \(\Phi\) and \(\beta\) are given by:
\[
\Phi = X_0^{-1} (I + AX_0^{-1})^{-1} AX_0^{-1}, \\
\beta = \log_2 |A + X_0| + \frac{\text{Tr} \left[ X_0^{-1} (I + AX_0^{-1})^{-1} A \right]}{\ln(2)} - \log_2 |X_0|.
\]
(14)
(15)
Applying (13) to \(\tilde{R}_i\) gives the following equation:
\[
-\tilde{R}_i \approx \eta_i + \nu_i + \frac{t_i}{\ln(2)} - \beta_i,
\]
(16)
where \(\beta_i\) is defined as in (15) with \(A = I_{N_R}\) and \(X_0 = \left( \Sigma_i + \tilde{G}_i Q_i^{(k)} \tilde{G}_i^\dagger \right)^{-1} H_j Q_j^{(k)} H_j^\dagger\) and where \(Q_i^{(k)}\) is defined as in Section III-A. As well, \(\eta_i, \nu_i,\) and \(t_i\) are defined as follows:
\[
\eta_i = -\log_2 \left| H_j Q_j H_j^\dagger \right|,
\]
(17)
\[
\nu_i = \log_2 \left| \Sigma_i + \tilde{G}_i Q_i \tilde{G}_i^\dagger \right|,
\]
(18)
\[
t_i = \text{Tr} \left[ \Phi_i \left( \Sigma_i + \tilde{G}_i Q_i \tilde{G}_i^\dagger \right)^{-1} H_j Q_j H_j^\dagger \right],
\]
(19)
where \(\Phi_i\) is given by (14) with \(A = I_{N_R}\) and \(X_0 = \left( \Sigma_i + \tilde{G}_i Q_i^{(k)} \tilde{G}_i^\dagger \right)^{-1} H_j Q_j^{(k)} H_j^\dagger\).

In order to ensure that \(-\tilde{R}_i\) satisfies the DCP ruleset [8], \(\nu_i\) and \(t_i\) were replaced by their respective first-order approximations given by (6) and (20), respectively.
\[
\text{Tr} \left[ AX^{-1} Y \right] \approx \text{Tr} \left[ AX_0^{-1} Y_0 \right] + \text{Tr} \left[ (AX_0^{-1})^\dagger (Y - Y_0) \right] - \text{Tr} \left[ (X_0^{-1} Y_0 AX_0^{-1})^\dagger (X - X_0) \right],
\]
(20)
which can be computed using the concept of matrix differentials [10], [11].

The approximations \(\tilde{\nu}_i\) and \(\tilde{t}_i\) are given by:
\[
\tilde{\nu}_i = \log_2 \left| Y_0^\dagger \right| + \frac{1}{\ln(2)} \text{Tr} \left[ (Y_0^\dagger)^{-1} G_i (Q_i - Q_i^{(k)}) \tilde{G}_i^\dagger \right],
\]
(21)
\[
\tilde{t}_i = -\text{Tr} \left[ (Y_0^\dagger)^{-1} H_j Q_j^{(k)} H_j^\dagger \Phi_i (Y_0^\dagger)^{-1} (G_i (Q_i - Q_i^{(k)}) \tilde{G}_i^\dagger) \right] + \text{Tr} \left[ (\Phi_i (Y_0^\dagger)^{-1})^\dagger H_j (Q_j - Q_j^{(k)}) H_j^\dagger \right] + \text{Tr} \left[ \Phi_i (Y_0^\dagger)^{-1} H_j Q_j^{(k)} H_j^\dagger \right],
\]
(22)
where \(Y_0^\dagger = \Sigma_i + \tilde{G}_i Q_i^{(k)} \tilde{G}_i^\dagger\). Therefore, the objective function, \(f_{scamp}\), is a convex function and is given by:
\[
f_{scamp} = \sum_{i=1}^{2} \left\{ \eta_i + \tilde{\nu}_i + \frac{\tilde{t}_i}{\ln(2)} - \beta_i \right\}.
\]
(23)

As such, the non-convex optimization problem (2) can be approximated by the following convex optimization problem.
\[
\min_{Q_1, Q_2} \sum_{i=1}^{2} \left\{ \eta_i + \tilde{\nu}_i + \frac{\tilde{t}_i}{\ln(2)} - \beta_i \right\}
\]
subject to: \(\text{Tr} \{ Q_i \} \leq P_{\text{max},i}, \ i = 1, 2\), \(Q_i \succeq 0, \ i = 1, 2\).
(24)
Note that optimization problems (24) and (9) differ only in the selection of the convex approximation (i.e., $f_{\text{DC}}$ and $f_{\text{SCAMP}}$).

The SCAMP algorithm, described in Algorithm 2, iteratively updates the objective function approximation, $f_{\text{SCAMP}}$, and solves the convex approximation until convergence. The convex optimization problem can also be solved using cvx, a package for solving disciplined convex programs in Matlab [12], [13].

**Algorithm 2: SCAMP Algorithm.**

Randomly initialize $Q_1^{(0)}$, $Q_2^{(0)}$;  
Initialize $k = 0$ ;
repeat
  Update $\tilde{\nu}_i$, $\tilde{t}_i$ ($i = 1, 2$) using (21) and (22), using $Q_1^{(k)}$, $Q_2^{(k)}$ ;
  Update $f_{\text{SCAMP}}$ using (23) ;
  Solve (24) for $Q_1^\star$, $Q_2^\star$ ;
  $k = k + 1$ ;
  Update $Q_i^{(k)} = Q_i^\star$, $i = 1, 2$ ;
until $f_{\text{SCAMP}}$ converges;
Apply Cholesky decomposition: $Q_i = L_i L_i^\top$, $i = 1, 2$ ;
Solve for $V_i$ using (3), $i = 1, 2$

### IV. ILLUSTRATIVE RESULTS

This section provides a performance comparison of the SCAMP, DC-based, SIP, and optimized HD. For all scenarios, the noise was normalized such that $E\{z_i z_i^\dagger\} = I_{N_R}$ ($i = 1, 2$). Hence, the forward channels, $H_i$ ($i = 1, 2$), were generated as zero-mean complex Gaussian random variables with a variance equal to the Signal-to-Noise-Ratio (SNR). Conversely, the estimated self-interference channels, $G_i$ ($i = 1, 2$), were generated as a zero-mean complex Gaussian random variable with a variance equal to the self-Interference-to-Noise Ratio (INR), where SNR/INR represents the Signal-to-self-Interference Ratio at each receiver (SIR). The SIR combines various factors including the distance between nodes and the amount of passive cancellation at each transceiver. The estimation error was generated as a zero-mean complex Gaussian random variable with a variance equal to $\sigma_{\text{err}}^2$. The simulations assume that $N_T = N_R = 4$ and that both nodes use an identical power (normalized to one).

Fig. 2 shows the FD sum-rate and FD-to-HD sum-rate ratio vs. SNR with SIR $= -10$ dB and $\sigma_{\text{err}}^2 = 1$. The results indicate that the SCP-based techniques can provide 40–65% improvements over optimized HD, even when $\sigma_{\text{err}}^2$ is quite large.

The results show that the DC-based algorithm provides the best sum-rate. However, it was observed that the performance of the DC-based algorithm is significantly more sensitive to $\sigma_{\text{err}}^2$ than the SCAMP algorithm. As such, the SCAMP approach may be more suitable for scenarios with more imperfect channel knowledge.

### V. CONCLUDING REMARKS

This paper presented two SCP algorithms for FD SU-MIMO systems. The structure of the problem naturally led itself to the DC-based algorithm. We introduced the SCAMP algorithm for solving general non-convex optimization problems with a logarithmic objective function. Illustrative results showed that the SCP approach provides substantial sum-rate improvements.
of both SIP and optimized HD. More specifically, sum-rate gains of 30–70\% were found in different scenarios. In particular, the DC-based algorithm generally offers better sum-rate than the SCAMP algorithm; however, the SCAMP approach is more robust to imperfect channel knowledge.

**ACKNOWLEDGMENTS**

The work presented in this paper is partly supported by the R&D Contract Advanced Transmission Multiple-Access Schemes for Future Radio Access from Huawei Technologies Canada.

**REFERENCES**


