Abstract—Cloud radio access network (Cloud-RAN) is a revolutionary architecture that provides a cost-effective way to improve both the network capacity and energy efficiency by shifting the baseband signal processing to a single baseband unit (BBU) pool, which enables centralized signal processing. However, in order to exploit the performance gains of full cooperation, full channel state information (CSI) is often required, which will incur excessive signaling overhead and degrade the network performance. To resolve the CSI challenge for Cloud-RAN, we propose a novel CSI acquisition method, called compressive CSI acquisition. This new method can effectively reduce the CSI signaling overhead by obtaining instantaneous coefficients of only a subset of all the channel links. As a result, the BBU pool will obtain mixed CSI consisting of instantaneous values of some links and statistical CSI for the others. We then propose a stochastic coordinated beamforming (SCB) framework to deal with the uncertainty in the available mixed CSI. The SCB problem turns out to be a joint chance constrained program (JCCP) and is known to be highly intractable. In contrast to all the previous algorithms for JCCP that can only find feasible but sub-optimal solutions, we propose a novel stochastic DC (difference-of-convex) programming algorithm with optimality guarantee. To reduce the computational complexity, we also propose two low-complexity algorithms using the scenario approach and the Bernstein approximation method for larger-sized networks. Simulation results will show that the proposed compressive CSI acquisition method can reduce the CSI overhead significantly, and the proposed SCB algorithms provide performances close to the full CSI case.

Index Terms—Cloud-RAN, compressive CSI acquisition, stochastic coordinated beamforming, joint chance constrained program, stochastic DC programming.

I. INTRODUCTION

Cloud Radio Access Network (Cloud-RAN) [1], [2] has recently been proposed as a revolutionary architecture for future cellular networks to meet the exponential growth of mobile data traffic. In conventional cellular networks, the network capacity is frozen in separate places. Cloud-RAN, however, is able to release such frozen capacity into a resource pool that can then be used to more flexibly deal with the network dynamics, while also enabling features such as automation and elasticity. Specifically, in Cloud-RAN, all the baseband signal processing will be shifted to a central location in a datacenter, called the baseband unit (BBU) pool, while conventional powerful base stations will be replaced by low-power low-cost remote radio heads (RRHs). These RRHs are connected to the BBU pool through high-bandwidth and low-latency transport links, as shown in Fig. 1. Such an approach has significant cost advantages and can reduce both the capital expenditure (CAPEX) and operational expenditure (OPEX). It is flexible enough to handle dynamic traffic, and can significantly improve both the spectrum efficiency and energy efficiency, thanks to the centralized signal processing and resource allocation.

In order to fully exploit the benefits of full cooperation in Cloud-RAN, full channel state information (CSI) is often required. However, as the BBU pool can typically support hundreds of RRHs, obtaining full CSI in Cloud-RAN will deplete the radio resources, which can be regarded as the “curse of dimensionality” of Cloud-RAN. In particular, Lozano et al. [3] showed that the full cooperation gain is limited by the overhead of the orthogonal pilot-assisted channel estimation for uplink transmission in large-scale cooperative cellular networks. Huh et al. [4] quantified the downlink training overhead for large-scale network MIMO, which is regarded as the system overhead bottleneck even if the uplink feedback overhead is ignored. Therefore, the development of novel and effective CSI acquisition methods is critical for the practical implementation of the fully cooperative Cloud-RAN.

Although there have been numerous research efforts on CSI acquisition, there is still a lack of systematic approach for CSI overhead reduction. For limited feedback wireless systems [5], most works are focused on the uplink feedback overhead reduction while assuming that full instantaneous CSI is available at each receiver [6]. For cooperative cellular networks, local CSI [7] and distributed CSI [8] based schemes were proposed to reduce CSI signaling overhead among the backhaul links. However, this will not necessarily reduce the CSI acquisition overhead. Maddah-Ali and Tse [9] proposed an alternative scheme requiring only outdated transmitter-side CSI (CSIT). Unfortunately, the proposed scheme does not scale well for large cooperative MIMO systems because the precoding block length in terms of time slots grows very fast with the number of antennas [10]. In [11], Jafar proposed a topological interference management scheme without CSIT except the topology information (i.e., which channel coefficients are strong), but this scheme still cannot reduce the downlink training overhead, since it requires each receiver to know all the channel coefficients’ realizations.

CSI acquisition is also critical for non-cooperative multi-user wireless systems. For the massive MIMO system [12], in which the base stations (BSs) are not required to cooperate, the CSI acquisition overhead is proportional to the number of mobile users in each cell for the time-division duplex (TDD) mode. However, the pilot contamination during the training phase has been well recognized as a main challenge in massive MIMO systems and it has not been solved satis-
factorily [13]. Adhikary et al. [10] proposed a CSI overhead reduction method exploiting the structure of the correlation of the aggregated channel matrix to reduce the “effective” CSI dimensionality for the frequency-division duplex (FDD) massive MIMO system based on a linear equal-spaced transmit antenna topology model. This is however, a restrictive assumption.

In contrast to all the previous works, in this paper, we propose a novel CSI acquisition method, called compressive CSI acquisition, which can systematically reduce both the pilot training overhead and uplink feedback overhead. Specifically, it is achieved by exploiting the sparsity of the large-scale fading coefficients and determining, before the training phase, the channel coefficients needed to obtain their instantaneous values. As a result, the BBU pool will obtain the mixed CSI, including a subset of instantaneous CSI and statistical CSI for the other channel coefficients.

To deal with the uncertainty in the mixed CSI, we then propose a novel stochastic coordinated beamforming (SCB) framework to minimize the total transmit power while guaranteeing the quality-of-service (QoS) requirements. In contrast to the approach of robust beamforming design, e.g., [14], which tries to guarantee the worst-case performance with CSI uncertainty, the proposed SCB framework will guarantee the system performance with a given outage probability. This new approach is motivated by the fact that most wireless systems can tolerate the occasional outages in the QoS requirements [15]. In spite of the distinct advantages and insights of applying the SCB framework to handle the CSI uncertainty, it falls into a joint chance constrained program (JCCP) [16] which is known to be highly intractable [17]. All the available algorithms (e.g., the scenario approach [18]–[20] and the Bernstein approximation method [15], [21], [22]) can only find feasible solutions without any optimality guarantee.

In contrast, in this paper, we propose a novel stochastic DC programming algorithm, which can find the globally optimal solution if the SCB problem is convex and find a locally optimal solution if the problem is non-convex. The main idea of the algorithm is to approximate the system probabilistic QoS constraint as a DC constraint, producing an equivalent DC programming algorithm, which can find the globally optimal solution if the problem is non-convex. As the solutions of the available conservative algorithms (i.e., the scenario approach and the Bernstein approximation method) will be used as the initial point for the proposed stochastic DC programming, better performance is guaranteed. The proposed stochastic DC programming algorithm can be regarded as the first attempt to guarantee the optimality of the solutions of JCCP.

To reduce the computational complexity, we develop another algorithm based on the scenario approach for medium-sized networks. In addition, we extend the Bernstein approximation method [15], [22], which was widely used for individual chance constraints, to the SCB problem with a joint chance constraint for large-scale networks.

We shall show that the proposed compressive CSI acquisition method can significantly reduce the CSI overhead, while providing performance close to the full CSI case, thanks to SCB. With different computational complexities and performances, the three proposed algorithms can serve as the toolkit to solve the SCB problem for Cloud-RAN with different network sizes.

B. Organization

The remainder of the paper is organized as follows. Section II presents the system model and problem formulations, followed by the problem analysis. In Section III, the stochastic DC programming algorithm is developed. Section IV presents two low-complexity algorithms, namely, the scenario approach and the Bernstein approximation method. Simulation results will be presented in Section V. Finally, conclusions and discussions are presented in Section VI.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a Cloud-RAN with L remote radio heads (RRHs), where the l-th RRH is equipped with $N_l$ antennas, and there are $K$ single-antenna mobile users (MUs), as shown in Fig. 1. In this architecture, all the base band units (BBUs) are moved into a single BBU pool, creating a set of shared processing resources, and enabling efficient interference management and mobility management. All the RRHs are connected to the BBU pool through high-capacity and low-latency transport links. It is also assumed that all the user data are available at the BBU pool.

The propagation channel from the l-th RRH to the k-th MU is denoted as $h_{kl} \in \mathbb{C}^{N_l}$, $1 \leq k \leq K$, $1 \leq l \leq L$. We focus on the downlink transmission, for which the joint signal processing is more challenging. Denote $\mathcal{K} \triangleq \{1, \ldots, K\}$ as the set of the mobile users. The received signal $y_k \in \mathbb{C}$ at MU
The feasible set of conic programming (SOCP) problem, which is convex and constraints:

$$
\text{minimize} \quad \sum_{l=1}^{L} h_{lk}^H v_k s_k + \sum_{i \neq k}^{L} \sum_{l=1}^{L} h_{lk}^H v_i s_i + n_k, \forall k \in K,
$$

where \( s_k \) is the encoded information symbol for MU \( k \) with \( \mathbb{E}[|s_k|^2] = 1 \), \( v_k \) is the transmit beamforming vector from the \( l \)-th RRH to the \( k \)-th MU, and \( n_k \sim \mathcal{CN}(0, \sigma_k^2) \) is the additive Gaussian noise at MU \( k \). We assume that \( s_k \)'s and \( n_k \)'s are mutually independent and all the users apply single user detection. The corresponding signal-to-interference-plus-noise ratio (SINR) for MU \( k \) is given by

$$
\Gamma_k(v, h_k) = \frac{|h_k^T v_k|^2}{\sum_{i \neq k} |h_i^T v_k|^2 + \sigma_k^2}, \forall k \in K.
$$

where \( h_k \triangleq [h_{1k}^T, h_{2k}^T, \ldots, h_{Lk}^T]^T = [h_{kn}]_{1 \leq n \leq N} \in \mathbb{C}^N \) with \( N = \sum_{l=1}^{L} N_l \), \( v_k \triangleq [v_{1k}^T, v_{2k}^T, \ldots, v_{Lk}^T]^T \in \mathbb{C}^N \) and \( v \triangleq [v_k]_{k=1}^{K} \in \mathbb{C}^{NK} \). The beamforming vectors \( v_k \)'s are designed to minimize the total transmit power while satisfying the QoS requirements for all the MUs. The beamformer design problem can be formulated as

$$
\mathcal{P}_{\text{Full}} : \text{minimize} \quad \sum_{l=1}^{L} \sum_{k=1}^{K} ||v_{lk}||^2
$$

subject to \( \Gamma_k(v, h_k) \geq \gamma_k, \forall k \in K \),

where \( \gamma_k \) is the target SINR for MU \( k \), and the convex set \( \mathcal{V} \) is the feasible set of \( v_{lk} \)'s that satisfy the per-RRH power constraints:

$$
\mathcal{V} \triangleq \left\{ v_{lk} \in \mathbb{C}^{N_l} : \sum_{k=1}^{K} ||v_{lk}||^2 \leq P_l, \forall l, k \right\},
$$

with \( P_l \) as the maximum transmit power of the RRH \( l \).

The problem \( \mathcal{P}_{\text{Full}} \) can be reformulated as a second order conic programming (SOCP) problem, which is convex and can be solved efficiently (e.g., via the interior-point method). Please refer to [2] for details. Such coordinated beamforming can significantly improve the network energy efficiency. However, solving problem \( \mathcal{P}_{\text{Full}} \) requires full CSI available at the BBU pool, which is impractical for Cloud-RAN given its large system size (e.g., both \( L \) and \( K \) may be larger than 100). In the next subsection, we will propose a novel CSI acquisition method, called compressive CSI acquisition, to reduce the CSI signaling overhead.

### A. Compressive CSI Acquisition

In this subsection, we will present the proposed compressive CSI acquisition method. The main idea is to determine the most “relevant” channel links that are critical for performance before the training phase, and then only the coefficients of these links will be obtained during CSI training.

1) CSI Overhead Reduction: We will first quantify the CSI overhead for both training and feedback phases. The discussion will be general, as we do not make any assumption on the duplexing mode. With compressive CSI acquisition, only part of the channel coefficients will be obtained. For user \( k \), define a set \( \Omega_k \) of size \( D_k \) \((0 \leq D_k \leq N)\) such that the channel coefficients \( h_{kn} \) will be obtained during CSI training if and only if \((k, n) \in \Omega_k \). Given \( \Omega_k \), \( \forall k \in K \), and assuming that orthogonal pilot symbols are used for downlink training, then the training overhead is proportional to \( \max_{1 \leq k \leq K} D_k \). This is justified by modeling the orthogonal pilot allocation problem as a graph coloring problem on an unweighted bipartite graph \( G = (\mathcal{N}, \mathcal{K}, \mathcal{E}) \), where \( \mathcal{N} \triangleq \{1, 2, \ldots, N\} \) is the set of transmit antennas, \( \mathcal{K} \) is the set of MUs. In this case, an edge \( e \in \mathcal{E} \) exists if \((n, k) \in \Omega \) where \( \Omega \triangleq \Omega_1 \cup \cdots \cup \Omega_K \), \( n \in \mathcal{N} \) and \( k \in \mathcal{K} \). By the Vizing’s theorem [24], the minimum number of colors assigned to the edges of a bipartite graph so that no two adjacent edges have the same color (corresponding to no mutual interference for pilot training) is its maximum degree. Therefore, the required number of orthogonal pilots is \( \max_{1 \leq k \leq K} D_k \) \( \in \mathcal{K} \), which quantifies the training overhead. For the uplink feedback overhead, which is needed for the FDD system, given \( \Omega_k \), \( \forall k \in K \), in order to guarantee a constant CSI distortion \( d \), the total CSI feedback bits should scale as \( \mathcal{O}((\sum_{k=1}^{K} D_k \log(1/d)) \) [6]. Therefore, \( \sum_{k=1}^{K} D_k \) is a good indicator for the feedback overhead.

From the above discussion, the CSI overhead is controlled by the sizes of \( \Omega_k \)'s. We propose to determine the sets \( \Omega_k \)'s before the pilot training phase, so that the CSI overhead can be effectively controlled, especially compared to the channel coherence time. This approach is fundamentally different from the conventional limited feedback wireless systems, which require knowledge of all the channel coefficients before the feedback. We will refer to this method as compressive CSI acquisition, as it is similar to “compressive sensing” [25], where useful information can be extracted with much fewer samples than obtaining complete data. This “compression” idea will be critical for the design of large-scale wireless cooperative networks, as there is no way to collect all the side information before actual processing. We should rather try to directly extract the relevant information so that efficient communication can be achieved.
2) CSI Selection Rule: Given the size constraints for the sets $\Omega_k$’s (i.e., $|\Omega_k| = D_k, \forall k \in K$), how to determine the indices of each set is a combinatorial optimization problem, which is intractable in general. In this paper, we propose a practical CSI selection rule by exploiting the sparsity of the large-scaling fading coefficients. Denote the support of the channel vector $h_k = [h_{kn}]_{1 \leq n \leq N} \in \mathbb{C}^N$ as $|\{n \in \mathbb{Z}^N : |h_{kn}| \geq \lambda\}|$, where $\lambda > 0$ is a pre-chosen threshold. Due to path loss and large-scale fading, the size of the support of each channel vector can be much smaller than $N$, which is the number of coefficients if full CSI is to be obtained. This property was exploited in [11] to measure the partial connectivity of the channel links for topological interference management, where the receivers will compare powers of the estimated channel links with a pre-chosen threshold to determine which channel links are strong. However, the approach in [11] requires that each receiver obtains all the instantaneous channel coefficients to measure the channel sparsity, which cannot reduce the downlink training overhead.

By Chebyshev’s inequality, $\Pr\{|h_{kn}| \geq \lambda\} \leq \frac{\theta_{kn}}{\lambda}$ with $\theta_{kn} = \sqrt{\mathbb{E}[|h_{kn}|^2]}$ representing the large-scale fading coefficient of the channel link $h_{kn}$, the following support of the large-scale fading coefficient vector $\theta_k = [\theta_{kn}]_{1 \leq n \leq N}$ can be regarded as a good estimate of the sparsity of the channel coefficients,

$$||\theta_k||_{t_0}(\lambda) \triangleq |\{n \in \mathbb{Z}^N : |h_{kn}| \geq \lambda\}|,$$

where $\lambda$ is a pre-chosen parameter. A similar idea on exploiting the sparsity of the large-scaling fading coefficients was presented in [10] under a linear equal-spaced transmit antenna topology model.

Based on the above discussion, we propose in this paper, the following sparsity based CSI selection rule to determine the sets $\Omega_k$’s.

**Sparsity Based CSI Selection Rule:** Given the CSI overhead constraints $|\Omega_k| = D_k, \forall k \in K$, rearranging the entries of the vector $\theta_k = [\theta_{kn}]_{1 \leq n \leq N}$ with decreased magnitudes $|\theta_{k(1)}| \geq |\theta_{k(2)}| \geq \cdots \geq |\theta_{k(N)}|$, then the set $\Omega_k$ is determined by including the indices of the $D_k$ largest entries of the vector $\theta_k$.

**Remark 1:** The proposed sparsity based CSI selection rule is easy to implement. It is possible to improve performance by developing more sophisticated selection rules. For example, a different selection rule based on statisticalCSI was proposed in [20], which, however, does not have any performance guarantee and is with higher implementation complexity. A full investigation on this aspect will be left to our future work, while, in this paper, we focus on stochastic coordinated beamforming to handle mixed CSI.

### B. Stochastic Coordinated Beamforming with Mixed CSI

With compressive CSI acquisition, the BBU pool will obtain mixed CSI, i.e., with instantaneous channel coefficients for links indexed in the set $\Omega$ and statistical CSI for the other channel links. The uncertainty in the available CSI brings a new technical challenge for the system design. To guarantee performance, we will impose a probabilistic QoS constraint, specified as follows

$$\Pr\{\Gamma_k(v, \xi) \geq \gamma_k, \forall k\} \geq 1 - \epsilon,$$  \tag{6}

where $\xi_k \triangleq [h_{kn}]_{(k,n) \in \Omega_k}$ denotes the vector of the channel coefficients between all the RRHs and the MU $k$ with only statistical information, and $0 < \epsilon < 1$ indicates that the system should guarantee the QoS requirements for all the MUs simultaneously with probability of at least $1 - \epsilon$. The probability is calculated over all the random vectors $\xi_k$’s. The stochastic coordinated beamforming (SCB) is thus formulated to minimize the total transmit power while satisfying the system chance constraint (6):

$$\mathcal{P}_{\text{SCB}}: \minimize_{v \in \mathcal{V}} \sum_{i=1}^{L} \sum_{k=1}^{K} \|v_{ik}\|^2$$

subject to $\Pr\{\Gamma_k(v, \xi_k) \geq \gamma_k, \forall k\} \geq 1 - \epsilon,$ \tag{7}

which is a joint chance constrained program (JCCP) [16], [17] and is known to be intractable in general.

**Remark 2:** An alternative way to deal with the uncertainty in mixed CSI is to impose the following individual probabilistic QoS constraints:

$$\Pr\{\Gamma_k(v, \xi_k) \geq \gamma_k, \forall k\} \geq 1 - \epsilon_k, \forall k \in K,$$ \tag{8}

where $0 < \epsilon_k < 1$ is the maximum tolerated outage probability for MU $k$. However, when $\epsilon_1 + \cdots + \epsilon_K \leq \epsilon$, by the union bound, the feasible set formed by the individual probabilistic QoS constraint (8) is a subset of the feasible set formed by the joint probabilistic QoS constraint (6) (Please refer to Section IV-B for details). Furthermore, when $\epsilon = 0$ in (6), the probabilistic QoS constraint (6) becomes a robust QoS constraint (i.e., the instantaneous QoS requirements should be satisfied for all the realizations of the random vector $\xi$), yielding a robust design. By allowing some outages in the QoS requirements, the feasible set formed by the constraint (6) will be larger than the robust case with $\epsilon = 0$, and thus it will provide better performance. Therefore, the proposed SCB framework provides higher flexibility in handling different QoS constraints.

**Remark 3:** In the mixed CSI model, we assume the instantaneous coefficients of $h_{kl}$’s are perfectly known at the BBU pool if $(k,l) \in \Omega_k$. However, the proposed SCB framework can be easily extended to other types of uncertainties in the available CSI, e.g., with estimation errors in the obtained channel coefficients. The key assumption of the SCB framework is that the BBU pool knows the distribution of the CSI with uncertainties.

1) Problem Analysis: There are two major challenges in solving $\mathcal{P}_{\text{SCB}}$. Firstly, the chance (or probabilistic) constraint (6) has no closed-form expression in general and is difficult to evaluate. Secondly, the convexity of the feasible set formed by the probabilistic constraint is difficult to verify. The general idea to handle such a constraint is to seek a safe and tractable approximation. “Safe” means that the feasible set formed by the approximated constraint is a subset of the original feasible set, while “tractable” means that the optimization problem
over the approximated feasible set should be computationally efficient (e.g., relaxed to a convex program).

A natural way to form a computationally tractable approximation is the scenario approach [18]. Specifically, the chance constraint (6) can be approximated by the following K.J sampling constraints:

\[ \Gamma_k(v, \xi^j_k) \geq \gamma_k, k \in K, 1 \leq j \leq J, \]  

(9)

where \( \xi^j_k = [\xi^j_k]^1_{1 \leq k \leq K}, 1 \leq j \leq J \) is a sample of J independent realizations of the random vector \( \xi \). The SCB problem \( \mathcal{P}_{SCB} \) thus can be approximated by a convex program based on the constraints (9). This approach can find a feasible solution with a high probability, which will be presented in Section IV-A. An alternative way is to derive an analytical upper bound for the chance constraint based on the Bernstein type inequality [21], [22], resulting in a deterministic convex optimization problem, which will be presented in Section IV-B. The Bernstein approximation based approach can thus find a feasible but suboptimal solution.

Although the above methods have the advantage of computational efficiency due to the convex approximation, the common drawback of all these algorithms is the conservativeness due to the “safe” approximation. Furthermore, it is also difficult to quantify the qualities of the solutions generated by the algorithms. This motivates us to seek a novel approach to find a more reliable solution to the problem \( \mathcal{P}_{SCB} \). In this paper, we will propose a stochastic DC programming algorithm to find the globally optimal solution to \( \mathcal{P}_{SCB} \) if the problem is convex and a locally optimal solution if it is non-convex, which can be regarded as the first attempt to guarantee the optimality of the solutions of the JCCP.

III. STOCHASTIC DC PROGRAMMING ALGORITHM

In this section, we propose a stochastic DC programming algorithm to solve the problem \( \mathcal{P}_{SCB} \), which will be served as a performance benchmark. We will first propose a DC programming reformulation for the problem \( \mathcal{P}_{SCB} \), which will then be solved by stochastic successive convex optimization. As this algorithm will use the solution either from the scenario approach or the Bernstein approximation method as the initial point, it will guarantee to outperform both of them.

A. DC Programming Reformulation for the SCB Problem

The main challenge of the SCB problem \( \mathcal{P}_{SCB} \) is the intractable chance constraint. In order to overcome the difficulty, we will propose a DC programming reformulation that is different from all the previous conservative approximation methods. We first propose a DC approximation to the chance constraint (6). Specifically, the QoS constraints \( \Gamma_k(v, \xi^j_k) \geq \gamma_k \) can be rewritten as the following DC constraints, (Please refer to Appendix A for the preliminaries on the DC functions and the corresponding DC program.)

\[ d_k(v, \xi^j_k) \triangleq c_{k,1}(v-k, \xi^j_k) - c_{k,2}(v, \xi^j_k) \leq 0, k \in K. \]  

(10)

where \( v-k \leq [v_k]_{k \neq k} \), and both \( c_{k,1}(v-k, \xi^j_k) \) and \( c_{k,2}(v, \xi^j_k) \) are convex quadratic functions in \( v \). Therefore, \( d_k(v, \xi^j_k) \)'s are DC functions in \( v \). Then, the chance constraint (6) can be rewritten as \( f(v) < \epsilon \), with \( f(v) \) given by

\[
f(v) \triangleq 1 - \Pr \left\{ \Gamma_k(v, \xi^j_k) \geq \gamma_k, \forall k \right\} \\
= \Pr \left\{ \max_{1 \leq k \leq K} d_k(v, \xi^j_k) > 0 \right\} \\
= E \left[ 1_{(0, +\infty)} \left( \max_{1 \leq k \leq K} d_k(v, \xi^j_k) \right) \right], \tag{11}
\]

where \( 1_A(z) \) is an indicator of set \( A \). That is, \( 1_A(z) = 1 \) if \( z \in A \) and \( 1_A(z) = 0 \), otherwise. The indicator function makes \( f(v) \) non-convex in general.

The conventional approach to deal with the non-convex indicator function is to approximate it by a convex function, yielding a conservative convex approximation. For example, using \( \exp(z) \geq 1_{(0, +\infty)}(z) \) will yield the Bernstein approximation [21]. Applying \( [\nu + z + \nu] \geq 1_{(0, +\infty)}(z), \nu > 0 \) will obtain a conditional value at risk (CVaR) type approximation, which is known as the tightest convex approximation [21]. Although these approximations enjoy the advantage of being convex, all of them are conservative and will lose optimality for the solution of the original problem. More specifically, only the feasibility of the solutions can be guaranteed with these approximations. Instead of seeking the convex approximation for the chance constraint, in this paper, we propose to use the following non-convex function [17] to approximate the indicator function \( 1_{(0, +\infty)}(z) \) in (11):

\[
\psi(z, \nu) = \frac{1}{\nu}([\nu + z]^+ - z^+), \nu > 0,
\]  

(12)

which is a DC function [26] in \( z \). Although the DC function is not convex, it does have many advantages. In particular, Hong et al. [17] proposed to use this DC function to approximate the chance constraint assuming that \( d_k(v, \xi^j_k), \forall k \in K \), are convex. However, we cannot directly extend their results for our problem, since the functions \( d_k(v, \xi^j_k) \)'s in (10) are non-convex. Fortunately, we can still adopt the DC function \( \psi(z, \nu) \) in (12) to approximate the chance constraint based on the following lemma.

Lemma 1 (DC Approximation for the Chance Constraint): The non-convex function \( f(v) \) in (11) can be approximated by the following DC function

\[
\hat{f}(v, \nu) = E \left[ \psi \left( \max_{1 \leq k \leq K} d_k(v, \xi^j_k), \nu \right) \right] \\
= \frac{1}{\nu} u(v, \nu) - u(v, 0), \nu > 0,
\]  

(13)

where

\[
u(v, \nu) = E \left[ \max_{1 \leq k \leq K + 1} s_k(v, \xi^j_k, \nu) \right] \tag{14}
\]

is a convex function and the convex quadratic functions \( s_k(v, \xi^j_k, \nu) \) are given by

\[
s_k(v, \xi^j_k, \nu) = \nu + c_{k,1}(v-k, \xi^j_k) + \sum_{i \neq k} c_{i,2}(v, \xi^j_i), \tag{15}
\]

for \( 1 \leq k \leq K \), and \( s_{K+1}(v, \xi^j_k, \nu) = \sum_{i=1}^{K} c_{i,2}(v, \xi^j_i) \) is a convex quadratic function too.

Proof: Please refer to Appendix B for details.
Based on the DC approximation function \( \hat{f}(v, \nu) \), we propose to solve the following problem to approximate the original SCB problem \( \mathcal{P}_{SCB} \):

\[
\mathcal{P}_{DC} : \begin{aligned}
    & \text{minimize} \quad \sum_{l=1}^{L} \sum_{k=1}^{K} \|v_{lk}\|^2 \\
    & \text{subject to} \quad \inf_{\nu > 0} \hat{f}(v, \nu) < \epsilon,
\end{aligned}
\]

where \( \inf_{\nu > 0} \hat{f}(v, \nu) \) is the most accurate approximation function to \( f(v) \). Program \( \mathcal{P}_{DC} \) is a DC program with the convex set \( V \), the convex objective function, and the DC constraint function [26]. One major advantage of the DC approximation \( \mathcal{P}_{DC} \) is the equivalence to the original problem \( \mathcal{P}_{SCB} \). That is, the DC approximation will not lose any optimality of the solution of the SCB problem \( \mathcal{P}_{SCB} \), as stated in the following theorem.

**Theorem 1 (DC Programming Reformulation):** The DC programming problem \( \mathcal{P}_{DC} \) in (16) is equivalent to the original SCB problem \( \mathcal{P}_{SCB} \).

**Proof:** Please refer to Appendix C for details.

As the constraint in \( \mathcal{P}_{DC} \) is an optimization problem, it is difficult to be solved directly. To circumvent this difficulty and by observing that \( \hat{f}(v, \nu) \) is nondecreasing in \( \nu \) for \( \nu > 0 \), as indicated in (56), we propose to solve the following \( \kappa \)-approximation problem

\[
\mathcal{P}_{DC}(\kappa) : \begin{aligned}
    & \text{minimize} \quad \sum_{l=1}^{L} \sum_{k=1}^{K} \|v_{lk}\|^2 \\
    & \text{subject to} \quad u(v, \kappa) - u(v, 0) < \kappa \epsilon,
\end{aligned}
\]

for any fixed parameter \( \kappa > 0 \) to approximate the original problem \( \mathcal{P}_{DC} \), following a similar approach in (17). Define the deviation of a given set \( A_1 \) from another set \( A_2 \) as [16]

\[
\mathbb{D}(A_1, A_2) = \sup_{x_1 \in A_1} \left( \inf_{x_2 \in A_2} \|x_1 - x_2\| \right),
\]

then we have the following theorem indicating that the optimal solution and the optimal value of \( \mathcal{P}_{DC}(\kappa) \) converge to those of the original SCB problem \( \mathcal{P}_{SCB} \) when \( \kappa \to 0 \).

**Theorem 2 (Optimality of the \( \kappa \)-Approximation):** Denote the set of the optimal solutions and optimal values of problems \( \mathcal{P}_{DC}(\kappa) \) and \( \mathcal{P}_{SCB} \) as \( S^*(\kappa) \), \( V^*(\kappa) \), \( S^* \) and \( V^* \), respectively, then we have \( \lim_{\kappa \to 0} \mathbb{D}(S^*(\kappa), S^*) = 0 \).

**Proof:** The proof follows [17, Theorem 2].

Thus, we can focus on solving the program \( \mathcal{P}_{DC}(\kappa) \), which will provide a good approximation to the original program \( \mathcal{P}_{SCB} \) when \( \kappa \) is small enough. Although \( \mathcal{P}_{DC}(\kappa) \) is still a non-convex DC program, it has the algorithmic advantage, as will be presented in the next subsection.

### B. Successive Convex Approximation Algorithm

In this subsection, we will present a successive convex approximation algorithm [17, 23] to solve the non-convex \( \kappa \)-approximation program \( \mathcal{P}_{DC}(\kappa) \). The main idea is to upper bound the non-convex DC constraint function in \( \mathcal{P}_{DC}(\kappa) \) by a convex function at each iteration. Specifically, at the \( j \)-th iteration, given the vector \( v[j] \in V \), for the convex function \( u(v, 0) \), we have

\[
u(v, 0) \geq u(v[j], 0) + 2\langle \nabla_v u(v[j], 0), v - v[j] \rangle,
\]

where \( (a, b) \triangleq \Im(a^\dagger b) \) for any \( a, b \in \mathbb{C} \) and the complex gradient \( \nabla_v u(v, \kappa) \) is given as follows.

**Lemma 2:** For any given \( \kappa > 0 \), the complex gradient of \( u(v, \kappa) \) with respect to \( v^* \) (the complex conjugate of \( v \)) is given by

\[
\nabla_v u(v, \kappa) = \mathbb{E}[\nabla_v s_k(v, \xi, \kappa)],
\]

where \( k^* = \arg \max_{1 \leq k \leq K} s_k(v, \xi, \kappa) \), and \( \nabla_v s_k(v, \xi, \kappa) = [\nu_{k,i}]_{1 \leq i \leq K} \) with \( \nu_{k,i} \in \mathbb{C}^N \) given by

\[
\nu_{k,i} = \left\{ \begin{array}{ll}
    \langle h_k h_k^\dagger + \frac{1}{\gamma_i} h_i h_i^\dagger \rangle v_i, & \text{if } i \neq k, \ 1 \leq k \leq K, \\
    0, & \text{otherwise,}
\end{array} \right.
\]

and \( \nabla_v s_{K+1}(v, \xi, \kappa) = [\nu_{K+1,i}]_{1 \leq i \leq K} \) with \( \nu_{K+1,i} = \frac{1}{\gamma_i} h_i h_i^\dagger v_i, \forall i \).

**Proof:** Please refer to Appendix D for details.

Therefore, the non-convex DC constraint function in \( \mathcal{P}_{DC}(\kappa) \) can be upper bounded by the following convex function:

\[
l(v, v[j]) \triangleq u(v, \kappa) - u(v[j], 0) - 2\langle \nabla_v u(v[j], 0), v - v[j] \rangle.
\]

Based on the convex approximation (21) to the DC constraint in \( \mathcal{P}_{DC}(\kappa) \), we will then solve the following stochastic convex optimization problem at the \( j \)-th iteration:

\[
\mathcal{P}_{DC}(v[j], \kappa) : \begin{aligned}
    & \text{minimize} \quad \sum_{l=1}^{L} \sum_{k=1}^{K} \|v_{lk}\|^2 \\
    & \text{subject to} \quad l(v, v[j]) < \kappa \epsilon.
\end{aligned}
\]

Define \( F_0 \) as the feasible set of the program \( \mathcal{P}_{SCB} \). The proposed stochastic DC programming algorithm to the SCB problem \( \mathcal{P}_{SCB} \) is presented as Algorithm 1 in the following table.

**Algorithm 1: Stochastic DC Programming Algorithm**

**Step 0:** Find the initial solution \( v[0] \in F_0 \) and set the iteration counter \( j = 0 \);

**Step 1:** If \( v[j] \) satisfies the termination criterion, go to End;

**Step 2:** Solve problem \( \mathcal{P}_{DC}(v[j], \kappa) \) and obtain the optimal solution \( v[j+1] \);

**Step 3:** Set \( j = j + 1 \) and go to Step 1;

**End.**

Based on Theorem 2 on the optimality of the \( \kappa \)-approximation, the convergence of the stochastic DC programming algorithm is presented in the following theorem, which reveals the main advantage compared with all the previous algorithms for the JCCP problem, i.e., it guarantees optimality as \( \kappa \to +\infty \).

**Theorem 3 (Convergence of Stochastic DC Programming):** Denote \( \{v[j]\} \) as the sequence generated by the stochastic DC programming algorithm. Suppose that the limit of the
sequence exists, i.e., \( \lim_{j \to +\infty} v^{[j]} = v^* \), which satisfies the Slater’s condition\(^1\), then \( v^* \) is the globally optimal solution of the SCB problem \( \mathcal{P}_{SCB} \) if it is convex; otherwise, \( v^* \) is a locally optimal solution, as \( \kappa \to 0 \).

**Proof:** Please refer to Appendix E for details.

In order to implement this algorithm, we need to address the following two issues. The first one is how to find a high-quality initial point; and the second one is how to solve the stochastic convex program \( \mathcal{P}_{DC}(v^{[j]}, \kappa) \) efficiently at each iteration. We will address them in the following two subsections, respectively.

1) **Initial Solution:** In principle, the stochastic DC programming algorithm can start from any feasible solution of the problem \( \mathcal{P}_{SCB} \). In this paper, we propose to use either the scenario approach or the Bernstein approximation method to generate a high-quality initial solution for the stochastic DC programming algorithm, which will be presented in Section IV.

2) **Sample Average Approximation Method for the Stochastic Convex Optimization Problem:** We propose to use the sample average approximation (SAA) based algorithm [16], [17] to solve the stochastic convex program \( \mathcal{P}_{DC}(v^{[j]}, \kappa) \) at the \( j \)-th iteration. Specifically, the SAA estimate of \( u(v, \kappa) \) is given by

\[
\bar{u}(v, \kappa) = \frac{1}{M} \sum_{m=1}^{M} \max_{1 \leq k \leq K+1} s_k(v, \xi^m, \kappa),
\]

where \( \xi^m(1 \leq m \leq M) \) is a sample of \( M \) independent realizations of the random vector \( \xi \). Similarly, the SAA estimate of the gradient \( \nabla v^* u(v, \kappa) \) is given by

\[
\nabla v^* u(v, \kappa) = \frac{1}{M} \sum_{m=1}^{M} \nabla v^* s_k^{*\kappa}(v, \xi^m, \kappa),
\]

where \( k^*_m = \arg \max_{1 \leq k \leq K+1} s_k(v, \xi^m, \kappa) \). Therefore, the SAA estimate of the convex function \( l(v, v^{[j]}) \) (21) is given by

\[
l(v, v^{[j]}) \triangleq \bar{u}(v, \kappa) - \bar{u}(v^{[j]}, 0) - 2\langle \nabla v^* u(v^{[j]}, 0), v - v^{[j]} \rangle,
\]

which is convex in \( v \). We will thus solve the following SAA based convex optimization problem

\[
\mathcal{P}_{DC}(v^{[j]}, \kappa, M) : \begin{array}{ll}
\text{minimize} & \sum_{l=1}^{L} \sum_{k=1}^{K} \| v_{lk} \|^2 \\
\text{subject to} & \bar{l}(v, v^{[j]}) < \kappa
\end{array}
\]

which can then be solved efficiently using the interior-point method [27], where \( x = [x_m]_{1 \leq m \leq M} \in \mathbb{R}^M \) is the collection of the slack variables.

The following theorem indicates that the SAA based program \( \mathcal{P}_{DC}(v^{[j]}, \kappa, M) \) for the stochastic convex optimization \( \mathcal{P}_{DC}(v^{[j]}, \kappa) \) will not lose any optimality in the asymptotic region.

**Theorem 4:** Denote the set of the optimal solutions and optimal values of problems \( \mathcal{P}_{DC}(v^{[j]}, \kappa) \) and \( \mathcal{P}_{DC}(v^{[j]}, \kappa, M) \) as \( S^*(v^{[j]}), V(v^{[j]}), S^*_M(v^{[j]}), V_M(v^{[j]}) \), respectively, then we have \( E(S^*_M(v^{[j]}), S^*(v^{[j]})) \to 0 \) and \( V_M(v^{[j]}) \to V^*(v^{[j]}) \) with probability one as the sample size increases, i.e., as \( M \to +\infty \).

**Proof:** Please refer to Appendix F for details.

Based on Theorems 1-4, we conclude that the proposed stochastic DC programming algorithm converges to the globally optimal solution of the SCB problem if it is convex and to a locally optimal solution if the problem is non-convex, in the asymptotic region, i.e., as \( \kappa \to 0 \) and \( M \to +\infty \).

**C. Complexity Analysis and Implementation Issues**

Denote by \( W \) the number of constraints of an SOCP problem, then the number of iterations of the interior-point method to solve the SOCP is upper bounded by \( O(\sqrt{W}) \) [28]. For the stochastic DC programming algorithm, at each iteration, we need to solve the convex QCQP program \( \mathcal{P}_{QCQP}^{[j]} \) (which is a subclass of SOCP [28]) using the interior-point method, with \( L + KM + 1 \) (\( M \) is the number of independent realizations of the random vector \( \xi \)) constraints. Therefore, the complexity of the stochastic DC programming algorithm at each iteration is approximated by \( O(\sqrt{L + KM + 1}) \).

The stochastic DC programming algorithm requires the distribution of the random vector \( \xi \) to generate the Monte Carlo samples [17]. This approach can be widely applied for any channel model, e.g., Raleigh fading or Nakagami fading channel models. As the computational complexity of the algorithm will be prohibitive for large-sized networks, its main application will be for a clustered deployment or to provide a performance benchmark for low-complexity algorithms.

**IV. LOW-COMPLEXITY ALGORITHMS FOR STOCHASTIC COORDINATED BEAMFORMING**

In this section, we propose two low-complexity algorithms (i.e., the scenario approach and the Bernstein approximation method) to solve the SCB problem \( \mathcal{P}_{SCB} \), aiming at finding high-quality initial solutions for the stochastic DC programming algorithm and also for practical implementation.
for large-sized networks. Among the three proposed SCB algorithms in this paper, the main difference is the way they deal with the chance constraint.

A. The Scenario Approach

For the scenario approach, we approximate the probabilistic QoS constraint (6) by multiple “sampling” QoS constraints. Specifically, denote $\xi_j^l \in [\xi_j^1, \xi_j^K], 1 \leq j \leq J$, as a sample of $J$ independent realizations of the random vector $\xi$, which can be generated from the Monte Carlo simulation. Then, given the $J$ realizations $H_j \triangleq \{\xi_j^l\}_{1 \leq j \leq J}$, the probabilistic QoS constraint (6) can be approximated by the following $KJ$ sampling QoS constraints:

$$\Gamma_k(\mathbf{v}, \xi_j^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J,$$

which indicate that the QoS requirements for all the $J$ channel realizations must be satisfied simultaneously. With the sampling constraints, we have the following scenario approximation problem:

$$\mathcal{P}_{SA}(H_j) : \min_{\mathbf{v} \in V} \sum_{k=1}^{L} \sum_{j=1}^{K} ||v_{jk}||^2$$

subject to $\Gamma_k(\mathbf{v}, \xi_j^l) \geq \gamma_k, \forall k, j.$

which can be reformulated as an SOCP problem. Define the beamformer $\mathbf{v}$, the probability of the QoS satisfiability:

$$H \triangleq \{\xi_j^l\}_{1 \leq j \leq J},$$

the beamformer probability of the QoS satisfiability:

$$\hat{H} \triangleq \{\xi_j^l\}_{1 \leq j \leq J},$$

the following equation:

$$\begin{align*}
&\text{Pr}\{H_k(\mathbf{v}, \xi_k^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J\} \geq 1 - \epsilon, \\
&\text{Pr}\{H_k(\mathbf{v}, \xi_k^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J\} \geq 1 - \epsilon, \\
&\text{Pr}\{H_k(\mathbf{v}, \xi_k^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J\} \geq 1 - \epsilon,
\end{align*}$$

where

$$\begin{align*}
&\text{Pr}\{H_k(\mathbf{v}, \xi_k^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J\} \\
&= \text{Pr}\{H_k(\mathbf{v}, \xi_k^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J\} \\
&= \text{Pr}\{H_k(\mathbf{v}, \xi_k^l) \geq \gamma_k, \forall k \in K, 1 \leq j \leq J\}.
\end{align*}$$

Thus, by solving the following equation:

$$\beta = \exp \left\{ \frac{(J \epsilon - NK + 1)^2}{2J \epsilon} \right\},$$

we obtain the approximately minimum value for $J$ as

$$J^* = \frac{1}{\epsilon} \left[ (NK - 1) + \ln \frac{1}{\beta} + \sqrt{\ln^2 \frac{1}{\beta} + 2(NK - 1) \ln \frac{1}{\beta}} \right].$$

Therefore, we can conclude that with $J \geq J^*$ realizations for the scenario approximation problem $\mathcal{P}_{SA}(H_j)$, the solution $\hat{v}(H_j)$ will be feasible for $\mathcal{P}_{SCB}$ with probability at least $1 - \beta$.

1) Computational Complexity Analysis: For the scenario approach, we need to solve the SOCP program $\mathcal{P}_{SA}(H_j)$ with $L + KJ$ ($J = \mathcal{O}(NK/\epsilon)$ as the number of independent realizations of the random vector $\xi$) constraints. Using the interior-point method, the computational complexity is approximated as $\mathcal{O}(\sqrt{L + NK^2}/\epsilon)$. As the sample size will increase drastically for the large-scale networks according to (33), we thus suggest to implement this algorithm for a medium-sized network, e.g., with $L = 6 \sim 10$. The scenario approach only requires knowledge of the samples of the random vector $\xi$. Hence, it can also be used for any channel distribution model.

Compared to the stochastic DC programming algorithm, this algorithm still has several drawbacks. First, it can only find a feasible but suboptimal solution to the problem $\mathcal{P}_{SCB}$. Second, the performance of the scenario approach cannot be improved by generating more samples of the random vector $\xi$, even with enough computational resources [17]. Actually, it will lead to a worse performance when the sample size increases as more sampling QoS constraints are required to be satisfied.

B. The Bernstein Approximation Method

For the Bernstein approximation method, the probabilistic QoS constraint (6) will be upper bounded by a convex constraint in closed form. In order to derive an analytical upper bound for the chance constraint (6), we assume Rayleigh fading channels, i.e., $h_{kn} \sim \mathcal{CN}(0, \theta_{kn})$, and they are independent of each other [22]. We will first convert the joint QoS chance constraint (6) to a set of individual channel constraints. Let $\epsilon_k > 0, 1 \leq k \leq K$, such that $\sum_{k=1}^{K} \epsilon_k = \epsilon$, then by the union bound the system outage constraints (6) and (11) can be upper bounded by [21]

$$f(\mathbf{v}) = \text{Pr}\left\{ \bigcup_{k=1}^{K} \{\Gamma_k(\mathbf{v}, \xi_k^l) < \gamma_k\} \right\} \leq \sum_{k=1}^{K} \text{Pr}\{\Gamma_k(\mathbf{v}, \xi_k^l) < \gamma_k\} < \epsilon,$$

where

$$\text{Pr}\{\Gamma_k(\mathbf{v}, \xi_k^l) < \gamma_k\} \leq \epsilon_k, \forall k \in K.$$
Next, we propose to use the Bernstein approximation method to give upper bounds for the individual probabilistic QoS constraints (35).

We rewrite the channel vector between all the transmit antennas and MU $k$ as

$$\mathbf{h}_k = \mathbf{w}_k + \mathbf{A}_k \mathbf{g}_k, \forall k \in K,$$  

(36)

where $\mathbf{w}_k = [\omega_{kn}]_{1 \leq n \leq N} \in \mathbb{C}^N$ is a deterministic vector with $\omega_{kn} = h_{kn}$ if $(k, n) \in \Omega_k$ and $\omega_{kn} = 0$ otherwise; $\mathbf{g}_k \sim \mathcal{CN}(0, \mathbf{I}_N)$, and $\mathbf{A}_k = \text{diag}\{\lambda_1, \ldots, \lambda_N\} \in \mathbb{R}^{N \times N}$ with $\lambda_n = 0$, if $(k, n) \in \Omega_k$ and $\lambda_n = \theta_n$, otherwise. Define $\mathbf{B}_k \triangleq \sigma_n \mathbf{Q}_k - \sum_{i \neq k} \mathbf{Q}_i$, $k \in K$, where $\mathbf{Q}_k \triangleq \mathbf{v}_k \mathbf{v}_k^\mathsf{H} \in \mathbb{C}^{N \times N}$ with $\text{rank}(\mathbf{Q}_k) = 1$, then the QoS constraints $\Gamma_k(\mathbf{v}, \mathbf{\xi}_k) \geq \gamma_k$ can be rewritten as $\mathbf{h}_k^\mathsf{H} \mathbf{B}_k \mathbf{h}_k \geq \sigma_k^2$, which is equivalent to

$$\mathbf{g}_k^\mathsf{H} \mathbf{A}_k \mathbf{B}_k \mathbf{A}_k \mathbf{g}_k + 2\mathbf{r}(\mathbf{g}_k^\mathsf{H} \mathbf{A}_k \mathbf{B}_k \mathbf{w}_k) + \omega_k^\mathsf{H} \mathbf{B}_k \omega_k \geq \sigma_k^2,$$  

(37)

where $k \in K$. Based on the equivalent expressions for the QoS constraints, we can derive a closed-form upper bound for the chance constraint (6) based on the following lemma.

**Lemma 4 (Bernstein-Type Inequality [22]):** Given the vector $\mathbf{r} \in \mathbb{C}^N$, $\mathbf{g} \sim \mathcal{CN}(0, \mathbf{I}_N)$ and $\mathbf{A} \in \mathbb{R}^{N \times N}$, for the random variable $X = \mathbf{g}^\mathsf{H} \mathbf{A} \mathbf{g} + 2\mathbf{r}(\mathbf{g}^\mathsf{H} \mathbf{r})$, we have

$$\Pr\{X > T(\mathbf{A}, \mathbf{r}, \eta)\} \geq 1 - e^{-\eta},$$

(38)

where

$$T(\mathbf{A}, \mathbf{r}, \eta) = \text{Tr} (\mathbf{A}) - \sqrt{2\eta \sqrt{\|\mathbf{A}\|_2^2 + 2\|\mathbf{r}\|^2} - \eta \lambda^+(\mathbf{A})},$$

with $\lambda^+ \triangleq \max\{\lambda_{\text{max}}(-\mathbf{A}), 0\}$.

Based on this lemma and (34), a conservative approximation of the probabilistic QoS constraint (6) is given by

$$T(\mathbf{A}_k \mathbf{B}_k \mathbf{A}_k, \mathbf{A}_k \mathbf{B}_k \mathbf{w}_k, -\ln \epsilon_k) + \omega_k^\mathsf{H} \mathbf{B}_k \omega_k \geq \sigma_k^2, \forall k,$$  

(39)

where $\sum_{k=1}^K \epsilon_k = \epsilon$.

By dropping the rank-one constraints for $\mathbf{Q}_k$’s, we will solve the following convex semi-definite relaxation (SDR) programming problem to approximate the original SCB problem:

$$\mathcal{P}_{\text{SCB}} : \text{minimize} \sum_{k=1}^K \text{Tr}(\mathbf{Q}_k)$$

subject to

$$T(\mathbf{A}_k \mathbf{B}_k \mathbf{A}_k, \mathbf{A}_k \mathbf{B}_k \mathbf{w}_k, -\ln \epsilon_k) + \omega_k^\mathsf{H} \mathbf{B}_k \omega_k \geq \sigma_k^2, \forall k,$$

$$\sum_{k=1}^K \mathbf{C}_{lk} \mathbf{Q}_k \leq P_l, \forall l.$$  

(40)

where $\mathbf{C}_{lk} \in \mathbb{R}^{N \times N}$ is a diagonal matrix with the $l$-th diagonal block matrix as $\mathbf{I}_N$. Introducing the slack variables $x \triangleq [x_k]_{1 \leq k \leq K}$ and $y \triangleq [y_k]_{1 \leq k \leq K}$ and denote $\mathbf{Q} \triangleq [\mathbf{Q}_k]_{1 \leq k \leq K}$, we have the following equivalent convex conic optimization problem [22] for $\mathcal{P}_{\text{Bern}}$:

$$\mathcal{P}_{\text{CO}} : \text{minimize} \sum_{k=1}^K \text{Tr}(\mathbf{Q}_k)$$

subject to

$$\text{Tr}(\mathbf{A}_k \mathbf{B}_k \mathbf{A}_k) + \omega_k^\mathsf{H} \mathbf{B}_k \omega_k - x_k \sqrt{-2 \ln \epsilon_k + y_k \ln \epsilon_k} \geq \sigma_k^2, \forall k,$$

$$\sqrt{\|\mathbf{A}_k \mathbf{B}_k \mathbf{A}_k\|_F^2 + 2\|\mathbf{A}_k \mathbf{B}_k \omega_k\|^2} \leq x_k, \forall k,$$

$$y_k \mathbf{I}_N + \mathbf{A}_k \mathbf{B}_k \mathbf{A}_k \geq 0, y_k \geq 0, \forall k,$$

$$\sum_{k=1}^K \mathbf{C}_{lk} \mathbf{Q}_k \leq P_l, \forall l,$$  

(41)

which is a deterministic convex optimization problem and can be solved by the interior-point method [27] efficiently. Denote $\mathbf{Q}^\ast = [\mathbf{Q}_k^\ast]_{1 \leq k \leq K}$ as the optimal solution of the problem $\mathcal{P}_{\text{CO}}$. If $\mathbf{Q}_k^\ast$ is rank-one, we perform rank-one decomposition $\mathbf{Q}_k^\ast = \mathbf{v}_k^\ast(\mathbf{v}_k^\ast)^\mathsf{H}$ to obtain the rank-one solution $\mathbf{v}_k^\ast$. Otherwise, we employ the Gaussian randomization method [29] to obtain a rank-one approximate solution from $\mathbf{Q}^\ast$.

1) **Computational Complexity Analysis:** For the Bernstein approximation method, we need to solve the conic optimization problem $\mathcal{P}_{\text{CO}}$ with $L + 3K$ constraints. Using the interior-point method, the computational complexity is approximated as $O(\sqrt{L + 3K})$. With the low computational complexity, the Bernstein approximation method can be implemented for large-sized networks.

Compared to the DC programming algorithm, the Bernstein approximation method can only find a feasible but suboptimal solution, and the conservativeness of this method is difficult to quantify. Moreover, the Bernstein approximation method requires knowledge of the explicit distribution of the random vector $\mathbf{\xi}$. In particular, in order to derive a closed-form upper bound for the chance constraint in $\mathcal{P}_{\text{SCB}}$, this method can only be applied for special distribution models such as the Raleigh fading channel model. Therefore, this method is not robust against the distribution of the random vector $\mathbf{\xi}$.

**C. Algorithms for SCB**

We summarize the three algorithms for the SCB problem $\mathcal{P}_{\text{SCB}}$ in Table I. Note that, for the stochastic DC programming algorithm, it is hard to quantify the computational complexity theoretically, as it is not clear how to upper bound the iteration number of the algorithm for termination, though its convergence speed is very fast as shown in simulations. Therefore, we only list the complexity as $O(\sqrt{L + KM + 1})$ for each iteration of the stochastic DC programming algorithm in this Table. We can see a clear tradeoff between the achievable performance and the computational complexity, which makes different algorithms applicable for different application scenarios.

**V. Simulation Results**

In this section, we simulate the performance of the proposed CSI acquisition method with different stochastic coordinated
Assumptions
Globally or locally optimal solution
Medium-sized networks ($1 \leq N \leq 148.1 + 37.6 \log 14$)
Statistical CSI
Complexity
Small-sized networks ($1 \leq N \leq 16$)
Statistical CSI
Large-scale networks ($L \geq 8$) 
Raleigh Fading

**TABLE I**

**ALGORITHMS FOR THE STOCHASTIC COORDINATED BEAMFORMING PROBLEM**

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Assumptions</th>
<th>Complexity</th>
<th>Optimality</th>
<th>Application Scenarios</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stochastic DC Programming</td>
<td>Statistical CSI</td>
<td>$O(\sqrt{L + KM + 1})$</td>
<td>Globally or locally optimal solution</td>
<td>Small-sized networks ($L = 1 \sim 5$)</td>
</tr>
<tr>
<td>The Scenario Approach</td>
<td>Statistical CSI</td>
<td>$O(\sqrt{L + KN^2/\kappa})$</td>
<td>Feasible but suboptimal solution</td>
<td>Medium-sized networks ($L = 6 \sim 10$)</td>
</tr>
<tr>
<td>The Bernstein Method</td>
<td>Raleigh Fading</td>
<td>$O(\sqrt{L + 3K})$</td>
<td>Feasible but suboptimal solution</td>
<td>Large-scale networks ($L &gt; 10$)</td>
</tr>
</tbody>
</table>

**TABLE II**

**SIMULATION PARAMETERS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pathloss at distance $d_{kl}$ (km)</td>
<td>148.1 + 37.6 log$<em>2(d</em>{kl})$ dB</td>
</tr>
<tr>
<td>Standard deviation of log-norm shadowing $\sigma_s$</td>
<td>8 dB</td>
</tr>
<tr>
<td>Small-scale fading distribution $g_{kl}$</td>
<td>$\mbox{CN}(0, I)$</td>
</tr>
<tr>
<td>Noise power $\sigma_k^2$ (10 MHz bandwidth)</td>
<td>-102 dB</td>
</tr>
<tr>
<td>Maximum transmit power of RRH $P_1$</td>
<td>1 W</td>
</tr>
<tr>
<td>Transmit antenna power gain</td>
<td>9 dBi</td>
</tr>
</tbody>
</table>

beamforming algorithms. We consider the following channel model for the link between the $k$-th user and the $l$-th RRH:

$$h_{kl} = 10^{-L(d_{kl})/20} \sqrt{\varphi_{kl} s_{kl}} c_{kl}, \forall k, l,$$

(42)

where $L(d_{kl})$ is the path-loss at distance $d_{kl}$, as given in Table II, $s_{kl}$ is the shadowing coefficient, $\varphi_{kl}$ is the antenna gain and $c_{kl}$ is the small-scale fading coefficient. We use the standard cellular network parameters as shown in Table II. The maximum outage probability that the system can tolerate is set as $\epsilon = 0.1$ and set $\epsilon_k = \epsilon/K, \forall k \in K$ for the Bernstein approximation algorithm. The proposed stochastic DC programming algorithm will stop if the difference between the objective values of the two consecutive iterations is less than $10^{-4}$. We set $\kappa = 10^{-2}$. Define the CSI dimensionality compression ratio for MU $k$ as

$$\beta_k = \frac{|\Omega_k|}{N} = \frac{D_k}{N}, \forall k \in K,$$

(43)

which can be regarded as the first-order measurement of the CSI acquisition overhead for Cloud-RAN.

**A. Convergence of the Stochastic DC Programming**

Consider a network with $L = 5$ single-antenna RRHs and $K = 3$ single-antenna MUs uniformly and independently distributed in the square region $[-400, 400] \times [-400, 400]$ meters. The pre-defined CSI acquisition overhead constraints are $|\Omega_k| = D_k = 3, \forall k \in K$, so the CSI dimensionality compression ratio is $\beta_k = \frac{3}{5} = 60\%, \forall k \in K$. The QoS requirements are set as $\gamma_k = 3\mbox{dB}, \forall k \in K$. The sample size for the simulation approach is 300 and for the stochastic DC programming algorithm is 100. The convergence speed of the stochastic DC programming algorithm is shown in Fig. 2, with the initial point being either the solution of the scenario approach or the solution of the Bernstein approximation method. This figure shows that the convergence speed of the proposed stochastic DC programming is very fast for the simulated scenario. Furthermore, it will converge to the same value regardless of the initial solutions, but different initial solutions may affect the convergence speed. This figure also clearly demonstrates the performance gain of the stochastic DC programming algorithm over both the scenario approach and the Bernstein approximation method.

**B. Total Transmit Power versus Target SINR**

With the same network setting as for Fig. 2, Fig. 3 shows the total transmit power with different algorithms for different target SINRs. The solution of the scenario approach is used as the initial point for the stochastic DC programming algorithm. We generate 300 samples for the scenario approach and 100 samples for the stochastic DC programming for each simulation. The CSI dimensionality compression ratio is set as $\beta_k = \frac{3}{5} = 60\%, \forall k \in K$ and the target QoS requirements are set as $\gamma_k = 3\mbox{dB}, \forall k \in K$. We simulate 50 network realizations, each with one channel realization. This figure shows that the proposed stochastic DC programming algorithm with the highest computational complexity, always outperforms both the scenario approach and the Bernstein approximation method. The solutions of the Bernstein approximation method are more conservative than the scenario approach, resulting in the highest total transmit power, but it enjoys the lowest computational complexity. This figure

---

2 The corresponding channel coefficients for the realization can be found at http://ihome.ust.hk/~yshiach/publications.html
3 The small number of realizations in the simulation is mainly due to the high computational complexity of the stochastic DC programming algorithm.
for the scenario approach with the CSI dimensionality ratio 
\( \beta_k = \frac{\gamma}{\beta} = 87.5\%, \forall k \in K \), there is a negligible performance loss due to incomplete CSI, while the performance loss with 
\( \beta_k = \frac{\gamma}{\beta} = 62.5\%, \forall k \in K \), is still relatively small.

Note that the performance loss compared to the full CSI case is partially due to the sub-optimality of the algorithms as demonstrated in Fig. 2 and Fig. 3. Furthermore, as we did not explicitly consider the CSI overhead, the actual performance gap to the full CSI case will be even smaller, as the higher overhead to obtain full CSI will degrade its performance. For some extreme cases, e.g., with a very short channel coherence time, the compressive CSI acquisition supported by stochastic coordinated beamforming will be a better option compared to obtaining full CSI.

VI. CONCLUSIONS AND DISCUSSIONS

In this paper, we proposed a unified framework consisting of a novel compressive CSI acquisition method and stochastic coordinated downlink beamforming with mixed CSI for Cloud-RAN. Simulation results showed that the CSI acquisition overhead can be significantly reduced (about 40\% for the simulated settings) with the proposed approach, with performance close to the full CSI case. The proposed SCB framework demonstrated its effectiveness of dealing with the uncertainty in mixed CSI while guaranteeing probabilistic QoS. Through analysis and simulations, the proposed stochastic DC programming algorithm for stochastic coordinated beamforming was shown to converge to an optimal solution, while all the previous algorithms failed to attain this attractive property.

This initial investigation demonstrated the effectiveness of the proposed compressive CSI acquisition method in terms of CSI overhead reduction, the strength of the SCB framework in terms of transmit power minimization, and also the advantage of the stochastic DC programming algorithm in terms of achieving an optimal solution for the JCCP problem. A key lesson from this study is that the challenge of CSI overhead reduction in Cloud-RAN can be overcome by exploiting the unique properties within this new network architecture, i.e., the sparsity in the large-scale aggregated channel matrix. Further investigation will be needed to improve the computational efficiency for the stochastic DC programming algorithm, and to design low-complexity precoders with mixed CSI.

APPENDIX A

PRELIMINARIES ON DC PROGRAMMING

In order to make the paper self-contained, we shall introduce in the appendix some preliminaries on the DC (difference-of-convex) programming [26] as follows:

Definition 1: Let \( \mathcal{V} \) be a convex subset of \( \mathbb{C}^n \). A real-valued function \( f : \mathcal{V} \rightarrow \mathbb{R} \) is called a DC (difference-of-convex) function on \( \mathcal{V} \), if there exists two convex functions \( g, h : \mathcal{V} \rightarrow \mathbb{R} \) such that \( f \) can be expressed in the form

\[
 f(x) = g(x) - h(x). \tag{44}
\]

The following proposition gives some properties for the DC functions.
**Proposition 1:** Let \( f(x) \) and \( f_i(x), i = 1, \ldots, m, \) be DC functions, then the following functions are also DC functions:

1. \( \sum_{i=1}^{m} \lambda_i f_i(x) \) for \( \lambda_i \in \mathbb{R}, i = 1, \ldots, m; \)
2. \( f^+(x) = \max\{f(x), 0\}, f^-(x) = \min\{f(x), 0\}. \)

The general form of a DC programming problem is given by

\[
\begin{aligned}
&\text{minimize } f_0(x) \\
&\text{subject to } f_k(x) \leq 0, \forall k,
\end{aligned}
\]

where \( f_k(x), k = 0, 1, \ldots, K, \) are DC functions and \( V \) is a convex subset of \( \mathbb{C}^n. \)

**APPENDIX B**

**PROOF OF LEMMA 1**

For simplicity, we denote \( c_k(v) \triangleq c_k(v-k) \), \( c_{k2}(v) \triangleq c_{k2}(v-k), \) and \( d_k(v) \triangleq d_k(v). \) For any \( v \in V, \) \( d_k(v) = c_k(v-k), \) \( \forall k \in K, \) are DC functions on \( V, \) as both \( c_k(v) \) and \( c_{k2}(v) \) are convex functions of \( v. \) For any \( \nu > 0, \) we first prove that the following function

\[
\psi\left(\max_{1 \leq k \leq K} d_k(v), \nu\right) = \frac{1}{\nu} \left[\nu + \max_{1 \leq k \leq K} d_k(v)\right]^{-} - \left(\max_{1 \leq k \leq K} d_k(v)\right)^{+},
\]

is also a DC function. The function \( d_k(v) \) can be rewritten as

\[
d_k(v) = c_k(v) + \sum_{i \neq k} c_{i2}(v) - \sum_{i=1}^{K} c_{i2}(v).
\]

Therefore, the following function

\[
\max_{1 \leq k \leq K} d_k(v) = \\
\max_{1 \leq k \leq K} \left\{ c_k(v) + \sum_{i \neq k} c_{i2}(v) - \sum_{i=1}^{K} c_{i2}(v), c_{i1}(v, \xi) \right\}
\]

is a DC function, as both the functions \( c_{i1}(v, \xi) \) and \( c_{i2}(v, \xi) \) are convex in \( v. \) Furthermore, for any \( z_1, z_2 \in \mathbb{R} \) and \( z = z_1 - z_2, \) we have \( z = \max\{z_1, z_2\} - z_2. \) Therefore,

\[
\psi\left(\max_{1 \leq k \leq K} d_k(v), \nu\right) = \frac{1}{\nu} [m(v, \nu) - m(v, 0)],
\]

is a DC function of \( v, \) as

\[
m(v, \nu) = \max\{\nu + C_1(v, \xi), C_2(v, \xi)\},
\]

is a convex function of \( v. \) According to Proposition 1, \( \hat{f}(v, \nu) = E[\psi(\max_{1 \leq k \leq K} d_k(v, \xi), \nu)] \) is a DC function on \( V. \) Therefore, the proof is completed.

**APPENDIX C**

**PROOF OF THEOREM 1**

In order to prove Theorem 1, we need to prove the following equality:

\[
\inf_{\nu > 0} \hat{f}(v, \nu) = f(v).
\]

First, we need to prove the monotonicity of the function \( \hat{f}(v, \nu) \) in the variable \( \nu. \) According to (49) and (50), the function \( \hat{f}(v, \nu) \) can be rewritten as

\[
\hat{f}(v, \nu) = E[\pi(\nu, C_1(v, \xi), C_2(v, \xi))],
\]

where

\[
\pi(\nu, z_1, z_2) \triangleq \frac{1}{\nu} [\max\{\nu + z_1, z_2\} - \max\{z_1, z_2\}],
\]

for any \( z_1, z_2 \in \mathbb{R} \) and \( \nu > 0. \) Therefore, we only need to prove the monotonicity of the function \( \pi(\nu, z_1, z_2) \) in the variable \( \nu. \)

Define \( z \triangleq z_1 - z_2, \) then we have

\[
\pi(\nu, z_1, z_2) = \left(1 + \frac{1}{\nu}\right) 1_{(-\nu, 0)}(z) + 1_{(0, +\infty)}(z).
\]

For any \( \nu_1 > \nu_2 > 0 \) and any \( z_1, z_2 \in \mathbb{R}, \) we have

\[
\pi(\nu_1, z_1, z_2) - \pi(\nu_2, z_1, z_2) = \left(1 + \frac{1}{\nu_1}\right) 1_{(-\nu_1, -\nu_2)}(z) + z \left(1 - \frac{1}{\nu_1} - \frac{1}{\nu_2}\right) 1_{(-\nu_2, 0)}(z) \geq 0.
\]

Therefore, \( \hat{f}(v, \nu) \) is nondecreasing in \( \nu \) for \( \nu > 0. \) Hence, we have

\[
\inf_{\nu > 0} \hat{f}(v, \nu) = \lim_{\nu \searrow 0} \hat{f}(v, \nu) = \frac{1}{\nu} \left[u(\nu, \nu) - u(\nu, 0)\right],
\]

where \( \nu \searrow 0 \) indicates that \( \nu \) decreasingly goes to 0. Next, we need to prove that \( \frac{\partial}{\partial \nu} u(\nu, \nu) \) exists and \( \frac{\partial}{\partial \nu} u(\nu, 0) = 0. \)

According to (50), we have \( u(\nu, \nu) = E[m(v, \xi, \nu)] - E[\max\{\nu + C_1(v, \xi), C_2(v, \xi)\}]. \) As \( \frac{\partial}{\partial \nu} (\max\{\nu + z_1, z_2\}) \triangleq 1_{(-\nu, +\infty)}(z), \) for any \( z \neq -\nu, \) and

\[
\Pr\left\{\max_{1 \leq k \leq K} d_k(v, \xi) = -\nu\right\} = 0,
\]

where \( \max_{1 \leq k \leq K} d_k(v, \xi) \triangleq C_1(v, \xi) - C_2(v, \xi) \) (48), we conclude that \( \frac{\partial}{\partial \nu} u(\nu, \nu) \) exists.

Let \( T \triangleq (\nu > 0) \) with \( T \) being an open set such that the cumulative distribution function \( F(v, \nu) \triangleq \Pr\{\max_{1 \leq k \leq K} d_k(v) \leq \nu\} \) of the random variable \( \max_{1 \leq k \leq K} d_k(v) \) is continuously differentiable for any \( \nu \in T. \) Next we will show that

\[
\frac{\partial}{\partial \nu} u(\nu, \nu) = \lim_{h \to 0} \frac{1}{h} E[m(v, \xi, \nu + h) - m(v, \xi, \nu)] = \Pr\left\{\max_{1 \leq k \leq K} d_k(v) > -\nu\right\} = 1 - F(v, -\nu).
\]

For any \( \nu \in T \) and \( v \in V, \) define the random variable

\[
X(h) \triangleq [m(v, \xi, \nu + h) - m(v, \xi, \nu)]/h,
\]

then we have the following two facts:
1) The limit of $X(h)$ exists and we have
\[
\lim_{h \to 0} X(h) = 1_{(-\nu, +\infty)} \left( \max_{1 \leq k \leq K} d_k(v, \xi) \right),
\]
with probability one.
2) $X(h)$ is dominated by a constant $C > 0$, i.e.,
\[
|X(h)| \leq C,
\]
where $0 < C < \infty$. This can be justified by
\[
|X(h)| = \frac{1}{h}[m(v, \xi, \nu) - m(v, \xi, \nu)]
\]
where $Q(v, \xi, \nu) \triangleq \nu + \max_{1 \leq k \leq K} d_k(v, \xi)$ and the last inequality is based on the fact $|x^* - y^*| \leq |x - y|$.

From the above two facts on the random variable $X(h)$, by the dominated convergence theorem to interchange an expectation and the limit as $h \to 0$, and together with Proposition 1 in [30], we have
\[
\frac{\partial}{\partial \nu} u(v, \nu) = \lim_{h \to 0} \mathbb{E}[X(h)]
\]
Therefore, we can prove the equality
\[
\inf_{\nu > 0} \hat{f}(v, \nu) = \lim_{\nu \to 0} \frac{1}{\nu} [u(v, \nu) - u(v, 0)]
\]
which is convex in $v$. Although this function is not holomorphic in its complex conjugate $v^*$, it is easy to verify that the function $m(v, v^*, \nu)$ is holomorphic in $v$ for a fixed $v^*$ and is also holomorphic in $v^*$ for a fixed $v$. Proving Lemma 2 is equivalent to prove that the gradient of $m(v, \xi)$ with respect to $v^*$ exists and equals

\[
\nabla_{v^*} \mathbb{E}[m(v, \xi, \nu)] = \mathbb{E}[\nabla_{v^*} m(v, \xi, \nu)].
\]

Based on the chain rule [31], the complex gradient of the function $m(v, \xi, \nu)$ with respect to $v^*$ exists and is given by
\[
\nabla_{v^*} m(v, \xi, \nu) \triangleq \frac{\partial m(v, \xi, \nu)}{\partial v^*} = \frac{\partial s_k(v, \xi, \nu)}{\partial v^*},
\]
with probability one, where $k^* = \arg \max_{1 \leq k \leq K+1} s_k(v, \xi, \nu)$. It is a vector operator and gives the direction of the steepest ascent of a real scalar-valued function.

Denote $\frac{\partial m(v, \xi, \nu)}{\partial v^*} \triangleq \frac{\partial m(v, \xi, \nu)}{\partial v^*} |_{1 \leq i \leq N K}$ and $\frac{\partial s_k(v, \xi, \nu)}{\partial v^*} \triangleq \frac{\partial s_k(v, \xi, \nu)}{\partial v^*} |_{1 \leq i \leq N K}$, where $v = [v_1, v_2, \ldots, v_{NK}]$, and define the following complex random variable
\[
Y(\Delta v^*_i) \triangleq \frac{1}{\Delta v_i} \left[ m(v_{-i}, v_i^* + \Delta v_i) - m(v_{-i}, v_i^*) \right],
\]
where $v_{-i} \triangleq [v_k]_{k \neq i}$, $\Delta v_i^* \in \mathbb{C}$ and $m(v) \triangleq m(v, \xi, \nu)$ for simplicity, then we have the following two facts on the random variable $Y(\Delta v_i^*)$:
1) The limit of $Y(\Delta v_i^*)$ exists and equals
\[
\lim_{\Delta v_i^* \to 0} Y(\Delta v_i^*) = \frac{\partial s_{k^*}}{\partial v_i^*},
\]
with probability one.
2) The random variable is dominated by a random variable $Z$ with $\mathbb{E}[Z] \leq +\infty$, i.e.,
\[
|Y(\Delta v_i^*)| \leq Z, \forall i,
\]
which can be verified by the following lemma.

**Lemma 5:** For any $x, y \in \mathcal{V}$, there exits a random variable $Z$ with $\mathbb{E}[Z] \leq \infty$ such that
\[
|m(x, \xi, \nu) - m(y, \xi, \nu)| \leq Z \|x - y\|.
\]

Proof: As $m(v)$ is convex in $v$, we have
\[
m(x) \geq m(y) + 2\langle \nabla_{v^*} m(y), x - y \rangle,
\]
\[
m(y) \geq m(x) + 2\langle \nabla_{v^*} m(x), y - x \rangle.
\]
Based on the above two inequalities and by the Cauchy-Schwarz inequality, we have
\[
|m(x) - m(y)| \leq 2 \left( \max_{v = x, y} \|\nabla_{v^*} m(v)\| \right) \|x - y\|. 
\]
where $Z_2$ is a random variable with $\mathbb{E}[Z_2] < +\infty$. Therefore, letting $Z = \max\{Z_1, Z_2\}$ with $\mathbb{E}[Z] < +\infty$, we have
\[
\nabla v \cdot m(v) = \frac{\partial s_k(v)}{\partial v^*} \leq \max\{Z_1, Z_2\} = Z. \tag{76}
\]
According to (73) and (76), we have the inequality (70).

Based on the above two facts (68) and (69) on the random variable $Y(\Delta v^*_i)$, and by the dominated convergence theorem to interchange an expectation and the limit as $\Delta v^*_i \to 0$ and Proposition 1 in [30], we have
\[
\lim_{\Delta v^*_i \to 0} \mathbb{E}[Y(\Delta v^*_i)] = \mathbb{E}\left[ \lim_{\Delta v^*_i \to 0} Y(\Delta v^*_i) \right] = \mathbb{E}\left[ \frac{\partial s_k}{\partial v^*_i} \right]. \tag{77}
\]
Based on the fact
\[
\nabla v \cdot m(v, \xi, \nu) = \left[ \lim_{\Delta v^*_i \to 0} \mathbb{E}[Y(\Delta v^*_i)] \right]_{1 \leq i \leq NK}, \tag{78}
\]
we get (65) and thus complete the proof.

Appendix E
Proof of Theorem 3

In order to prove this theorem, we first introduce some preliminaries on the conditions of optimality and Lagrange multipliers for optimization problems.

A. Preliminaries on the Optimality and Lagrange Multipliers

For simplicity, we will only consider the case with real variables and functions. The extension to complex variables is straightforward. Consider the following optimization problem:

\[
\begin{align*}
\text{minimize} & \quad f_0(x) \\
\text{subject to} & \quad f_i(x) \leq 0, i = 1, \ldots, m, \tag{79}
\end{align*}
\]

which may not be convex, where $f_i(x), i = 0, 1, \ldots, m$ are differentiable functions and the feasible set is denoted as $\mathcal{X}$.

In order to ensure there are Lagrange multipliers for problem (79) we need the following constraint qualification assumption.

Assumption 1 (Constraint Qualification): [32, Theorem 6.14] For every feasible $x^* \in \mathcal{X}$, if $\lambda = (0, \ldots, 0)$ is the only vector satisfying the following condition:

\[
\lambda_i \nabla_x f_1(x^*) + \cdots + \lambda_m \nabla_x f_m(x^*) = 0, \tag{80}
\]

where $\lambda_i \geq 0$, $\lambda_i f_i(x^*) = 0, i = 1, \ldots, m$, then we say that the feasible set $\mathcal{X}$ is regular at the point $x^*$.

This assumption is a commonly used regularity condition in numerical optimization. Please refer to Assumption 6 in [17] for details.

Based on the constraint qualification (80), we have the following optimality conditions for the optimization problem (79).

Lemma 6 (First-Order Conditions for Optimality): [32, Theorem 6.12 and Corollary 6.15] Let $x^* \in \mathcal{X}$,

1. If $x^*$ is locally optimal and the constraint qualification (80) is satisfied at $x^*$, there must be a Lagrange multiplier vector $\lambda^* = (\lambda^*_1, \ldots, \lambda^*_m)$ such that

\[
\nabla_x f_0(x) + \sum_{i=1}^m \lambda^*_i \nabla_x f_i(x^*) = 0, \tag{81}
\]

where $\lambda^*_i \geq 0, \lambda^*_i f_i(x^*) = 0, i = 1, \ldots, m$.
2. If $f_0(x)$ and $\mathcal{X}$ are convex and such vector $\lambda^*$ exists, then $x^*$ is globally optimal. $(\lambda^*, x^*)$ is called a Karush-Kuhn-Tucker (KKT) pair of (79).

In order to prove Theorem 3, we need the results [17, Property 1,2, 3] on the convergence of the successive convex approximation algorithm for the DC programming problem, which will be presented in next subsection.

B. Convergence of Successive Convex Approximation

Consider the following DC programming problem with real-valued variables and functions (the extension to complex values is straightforward):

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad g(x) - h(x) \leq 0, \tag{82}
\end{align*}
\]

where $f(x), g(x)$ and $h(x)$ are continuously differentiable convex functions in $x$ and $\mathcal{X}$ is a convex compact set. Given a vector $y \in \mathcal{X}$, define the following convex set as

\[
\Omega(y) \triangleq \{ x \in \mathcal{X} : g(x) - [h(y) + \nabla h(y)^T (x - y)] \leq 0 \},
\]

which is a subset of the feasible set $\Omega_0$ of the DC program (82). Define the sequence $\{x^{[i]}\}$ generated by the following steps:

1. Given the initial solution $x^{[0]} \in \Omega_0$ and set the iteration counter $j = 0$;
2. Stop if $x^{[j]}$ satisfies the KKT condition of the problem (82);
3. Obtain $x^{[j+1]}$ by solving the following convex optimization problem:

\[
x^{[j+1]} = \arg \min_{x \in \Omega(x^{[j]})} f(x) \tag{83}
\]

4. Set $j = j + 1$ and go to step 2).

We say that the Slater’s condition [17, 27] holds at $y$ if

\[
\text{int} \Omega(y) \neq \emptyset, \tag{84}
\]

where $\text{int} A = \{ x \mid \exists V \in N(x), V \subset A \}$ denotes the interior of the set $A$ with $N(x)$ as the collection of all neighborhoods of $x$. Define $\bar{x}$ as a cluster point of a sequence $\{x^{[i]}\}$, if it is a limit of some subsequence of $\{x^{[i]}\}$ [32]. Then we have the following convergence result of the successive convex approximation algorithm.

Proposition 2: If $f(x)$ is strictly convex and all the cluster points of the sequence $\{x^{[i]}\}$, generated by the above steps 1)-4), satisfy the Slater’s condition (84), then the sequence $\{x^{[i]}\}$ converges to a KKT point of the DC programming problem (82).

Proof: The proof follows directly from [17, Property 1,2, 3]; we omit the details due to the limited space.

C. Convergence of the Stochastic DC Programming Algorithm

Now, we address the proof of Theorem 3 based on the results in the previous two subsections. For the problem $\mathcal{P}_{\mathcal{DC}}(\kappa)$, we have the facts that its objective function is strictly convex in $v$, $\mathcal{V}$ is a convex compact set, and both the functions
As the set \( V(u, \kappa) \) and \( u(v, 0) \) are continuously differentiable convex functions in \( v \). Therefore, we need to prove that all the cluster points of \( \{v^i\} \) generated by Algorithm 1 satisfy the Slater’s condition. As any cluster point \( \hat{v} \) of \( \{v^i\} \) belongs to the following feasible set
\[
\mathcal{F}(\kappa) \triangleq \{ v \in V : u(v, \kappa) - u(v, 0) \leq \kappa \epsilon \},
\]
we need to prove that, for any cluster point \( \hat{v} \in \mathcal{F}(\kappa) \), we have \( \text{int} \mathcal{F}(\kappa, \hat{v}) \neq \emptyset \) (84), where \( \mathcal{F}(\kappa, \hat{v}) \) is defined as
\[
\mathcal{F}(\kappa, \hat{v}) \triangleq \{ v \in V : l(v, \hat{v}) \leq \kappa \epsilon \},
\]
with \( l(v, \hat{v}) \) defined in (21).

We prove it by contradiction. Suppose that there exist \( \kappa_i \rightarrow \infty \) and \( \hat{v}_i \in \mathcal{F}(\kappa_i) \) such that \( \text{int} \mathcal{F}(\kappa_i, \hat{v}_i) = \emptyset \), which implies \( u(\hat{v}_i, \kappa_i) - u(\hat{v}_i, 0) = \kappa_i \epsilon \) and
\[
\begin{align*}
\hat{v}_i & \in \arg\min_{v \in V} l(v, \hat{v}_i) - \kappa_i \epsilon.
\end{align*}
\]
As the set \( V \) is convex, the above optimization problem is a convex problem. Therefore, we have the following necessary optimality condition that the derivative of the Lagrangian with respect to \( v^* \) at the optimal solution \( \hat{v}_i \) is zero, i.e.,
\[
\frac{1}{\kappa_i} \left[ \nabla_v \cdot u(\hat{v}_i^L, \kappa_i) - \nabla_v \cdot u(\hat{v}_i^L, 0) \right] + \sum_{l=1}^{L} \lambda_l \hat{v}_l^L = 0,
\]
where \( \lambda_l \)’s are the Lagrange dual variables and \( \hat{v}_i \triangleq [\hat{v}_i^L]_{1 \leq l \leq L} \in \mathbb{C}^{N_k L} \) with \( v_l^i \in \mathbb{C}^{N_k} \), and the corresponding optimal value is given by
\[
u(\hat{v}_i, \kappa_i) - u(\hat{v}_i, 0) = \kappa_i \epsilon.
\]
As \( \{ \hat{v}_i \} \) in \( V \) and \( \mathcal{F}(\kappa_i) \) is compact, \( \{ \hat{v}_i \} \) has a cluster point \( \hat{v} \triangleq [\hat{v}_i^L]_{1 \leq l \leq L} \in \mathcal{F}_0 \), where \( \mathcal{F}_0 \) is the feasible set of the original problem \( \mathcal{P}_\mathcal{SCB} \). That is, there is a subsequence \( \{ \hat{v}_{k_i} \} \) of \( \{ \hat{v}_i \} \) such that \( \hat{v}_{k_i} \rightarrow \hat{v} \) as \( i \rightarrow +\infty \). According to (62), we have
\[
\frac{\partial}{\partial \nu^L} \nabla_v \cdot u(\hat{v}_{k_i}, \kappa_{k_i}) = \nabla_v \cdot \left( \frac{\partial}{\partial \nu^L} u(\hat{v}_{k_i}, \kappa_{k_i}) \right) = -\nabla_v \cdot F(\hat{v}_{k_i}, -\lambda_{k_i}). \tag{89}
\]
Therefore, we have
\[
\lim_{i \rightarrow +\infty} \nabla_v \cdot u(\hat{v}_{k_i}, \kappa_{k_i}) - \nabla_v \cdot u(\hat{v}_{k_i}, 0) = -\nabla_v \cdot F(\hat{v}_0, 0) = \nabla_v \cdot f(\hat{v}),
\]
\[
\lim_{i \rightarrow +\infty} \lambda_{k_i} \hat{v}_{k_i}^L = \sum_{l=1}^{L} \lambda_l \hat{v}_l^L,
\]
where \( \hat{v}_{k_i} \) and \( \hat{v} \) belong to the compact set \( V \).

Therefore, according to (90) and (91), we have the following two results:
\[
\nabla_v \cdot f(\hat{v}) + \sum_{l=1}^{L} \lambda_l \hat{v}_l^L = 0, \quad \hat{v} = [\hat{v}_l^L]_{1 \leq l \leq L} \in \mathcal{F}_0,
\]
and
\[
f(\hat{v}) = \epsilon, \hat{v} \in \mathcal{F}_0.
\]
However, for the original problem \( \mathcal{P}_\mathcal{SCB} \), we have the following constraint qualification (80). Specifically, by [32, Theorem 6.14], for the feasible point \( \nu \in \mathcal{F}_0 \), the following two constraints should be satisfied simultaneously
\[
\begin{align*}
\lambda_0 \nabla_v \cdot f(v^L) + \sum_{l=1}^{L} \lambda_l v_l^L &= 0, \\
\lambda_0 (f(\nu) - \epsilon) &= 0,
\end{align*}
\]
to ensure the existence of KKT pairs, where \( \lambda_0 = 0 \). We can see that the results (92) and (93) contradict the constraint qualification conditions (94). Combing Theorem 2, we complete the proof.

**APPENDIX F**

**PROOF OF THEOREM 4**

By [16, Theorem 7. 50] and [17, Theorem 6], we have that the SAA estimate \( \hat{I}(\nu, \hat{v}^j, M) \) converges to \( I(\nu, v^j) \) uniformly on the convex compact set \( V \) with probability one as \( M \rightarrow +\infty \), i.e.,
\[
\sup_{v \in V} [\hat{I}(\nu, \nu^j, M) - I(\nu, \nu^j)] \rightarrow 0, \quad M \rightarrow +\infty,
\]
with probability one. Furthermore, by [16, Theorem 5.5] and theorem 2, we have \( V^*_M(\nu^j) \rightarrow V^*(\nu^j) \) and \( \mathbb{D}(S_M^*(\nu^j), S^*(\nu^j)) \rightarrow 0 \) with probability one as \( M \rightarrow +\infty \). Therefore, we complete the proof.

**REFERENCES**


