Distributed Consensus-based Estimation for Small Cell Cooperative Networks

Dirk Wübben, Henning Paul, Ban-Sok Shin, Guang Xu, Armin Dekorsy
Department of Communications Engineering
University of Bremen, 28359 Bremen, Germany
Email: {wuebben, paul, shin, xu, dekorsy}@ant.uni-bremen.de

***Abstract*—The dense deployment of small cells is a promising approach to realize the ever-growing rate demand in future wireless communication systems and centralizing RAN functionality permits joint multi-cell processing at the cost of backhaul traffic. In order to limit the backhaul requirements, cooperative processing among distributed radio access points is an interesting alternative for, e.g., advanced radio resource management, joint cooperative transmission, or joint reception. This paper focusses on cooperative multi-user detection by applying the Distributed Consensus-based Estimation (DiCE) algorithm and two recently proposed modifications for accelerating the iterative approach and to reduce communication overhead. The proposed schemes are investigated by means of computational complexity, communication overhead, and estimation performance.**

I. Introduction

Ultra dense deployment of low-power small cells is a promising candidate to deal with the exponential growth of traffic in 5G mobile networks. Small cells reduce the distance between the radio access points (RAP) and the user terminals (UEs) and allows for reusing the spectrum by neighboring RAPs [1]. In order to cope with the strong interference scenario, multi-cell processing is needed and centralized RAN (C-RAN) utilizing base-band pooling units (BBU) is currently under discussion for joint processing among RAPs [2]. However, the exchange of in-phase/quadrature (I/Q) samples between RAPs and BBUs requires deployment of fibre links over larger distances, as the central processing nodes are usually far away from RAPs. To reduce high-capacity, long-distance backhaul (BH) links it would be beneficial to interconnect the RAPs within a geometrical area and perform cooperative processing [3]. Due to the rather short distances between RAPs, the deployment of low-cost wireless links offering huge data rates, e.g., offered by mmWave transmissions, becomes feasible.

In recent publications we proposed the Distributed Consensus-based Estimation (DiCE) algorithm [4] and applied the method for joint multi-user detection by iteratively exchanging local estimates between RAPs [5]. Furthermore, we proposed the RO-DiCE (Reduced Overhead DiCE) [6] and the Fast-DiCE [7] modifications to reduce the communication overhead among RAPs and to improve the convergence speed. In this paper, we oppose the DiCE and its modifications and investigate in particular the required communication overhead for different backhaul topologies and the computational complexity.

The remainder of this paper is organized as follows. The system model is introduced in Section II and the considered distributed approaches for signal estimation are discussed in Section III. The performance of these approaches are investigated in Section IV, where also the analytical comparison w.r.t computational complexity and communication overhead is presented. The paper is concluded in Section V.

II. System Model

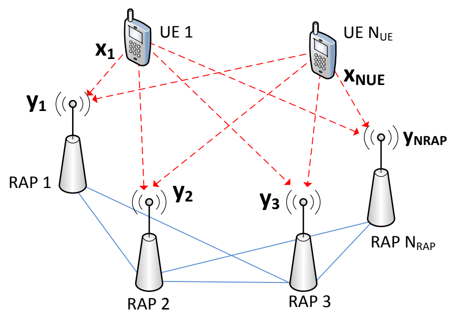

Fig. 1. Small-cell network where $N_{\rm UE}$ mobile users are served by $N_{\rm RAP}$ radio access points with information exchange between neighboring nodes.

Fig. 1 shows the small cell uplink scenario where the data vectors $\boldsymbol{x}_u$ transmitted by the $N_{\rm UE}$ users is received by a set of $N_{\rm RAP}$ radio access points (RAPs) interconnected by some kind of BH network. These RAPs represent the nodes of a network described by the geometric graph $\mathcal{G} := \{\mathcal{J}, \mathcal{E}\}$, where $\mathcal{J} := \{1, \ldots, N_{\rm RAP}\}$ denotes the set of nodes and $\mathcal{E}$ represents the set of edges for the linked nodes. For exchanging information, each node $j \in \mathcal{J}$ communicates with its neighboring nodes $i \in \mathcal{N}_j \subseteq \mathcal{J}$ (i.e., $(i, j) \in \mathcal{G}$). Without loss of generality, it is assumed that each user u uses $N_{\rm T}$ antennas to transmit complex-valued symbol vectors $\boldsymbol{x}_u \in \mathcal{A}^{N_{\rm T}\times 1}$ with elements of the modulation alphabet $\mathcal{A}$ and we can construct an overall message vector $\mathbf{x} = [\boldsymbol{x}_1^{\rm H} \;\ldots\; \boldsymbol{x}_{N_{\rm UE}}^{\rm H}]^{\rm H}$ containing $N_{\rm I} = N_{\rm UE} \cdot N_{\rm T}$ input components per time instant.

The received signal vector $\mathbf{y}_j \in \mathbb{C}^{N_{\rm R}\times 1}$ at RAP j equipped with $N_{\rm R}$ receive antennas is given by

$$\mathbf{y}_j = \mathbf{H}_j \mathbf{x} + \mathbf{n}_j \tag{1}$$

with the complex-valued channel matrix $\mathbf{H}_j \in \mathbb{C}^{N_{\rm R}\times N_{\rm I}}$ and the $\mathbf{n}_j$ additive Gaussian noise vector containing i.i.d. elements with variance σ_n^2.

In general, each RAP could separately perform a local estimation for $\mathbf{x}$. However, improved estimation performance can be achieved by processing all receive signals $\mathbf{y}_j$, $j \in \mathcal{J}$, jointly by, e.g., forwarding all observations $\mathbf{y}_j$ and all channel matrices $\mathbf{H}_j$ to a central node (*fusion center*). By constructing the stacked observation vector $\mathbf{y} = [\mathbf{y}_1^{\mathrm{H}} \dots \mathbf{y}_{N_{\mathrm{RAP}}}^{\mathrm{H}}]^{\mathrm{H}}$, stacked channel matrix $\mathbf{H} = [\mathbf{H}_1^{\mathrm{H}} \dots \mathbf{H}_{N_{\mathrm{RAP}}}^{\mathrm{H}}]^{\mathrm{H}}$, and stacked noise vector $\mathbf{n} = [\mathbf{n}_1^{\mathrm{H}} \dots \mathbf{n}_{N_{\mathrm{RAP}}}^{\mathrm{H}}]^{\mathrm{H}}$, the system equation

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{2}$$

for the general $N_{\mathrm{I}} \times N_{\mathrm{O}}$ MIMO system with $N_{\mathrm{I}} = N_{\mathrm{UE}} N_{\mathrm{T}}$ input and $N_{\mathrm{O}} = N_{\mathrm{RAP}} N_{\mathrm{R}}$ output signals can be set up facilitating central estimation. As an example, we consider the central Least Squares (LS) problem

$$\tilde{\mathbf{x}} = \arg \min_{\mathbf{x}' \in \mathbb{C}^{N_{\mathrm{I}} \times 1}} \|\mathbf{y} - \mathbf{H}\mathbf{x}'\|^2 \tag{3}$$

achieving the estimate $\tilde{\mathbf{x}}$ based on all observations $\mathbf{y}$. The solution is given by the Zero-Forcing (ZF) linear equalizer

$$\tilde{\mathbf{x}}_{\mathrm{ZF}} = (\mathbf{H}^{\mathrm{H}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{H}}\mathbf{y} = \mathbf{H}^{+}\mathbf{y} \,, \tag{4}$$

which filters the observation vector $\mathbf{y}$ with the Moore-Penrose Pseudo-Inverse of the channel matrix $\mathbf{H}^{+} = (\mathbf{H}^{\mathrm{H}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{H}}$. It is well known, that the ZF equalizer suffers from noise amplification and better results can generally be achieved by applying the Minimum Mean Square Error (MMSE) criterion. The MMSE criterion reduces the overall estimation error and calculates as

$$\tilde{\mathbf{x}}_{\mathrm{MMSE}} = (\mathbf{H}^{\mathrm{H}}\mathbf{H} + \sigma_n^2 \mathbf{I}_{N_{\mathrm{I}}})^{-1}\mathbf{H}^{\mathrm{H}}\mathbf{y} = \underline{\mathbf{H}}^{+}\underline{\mathbf{y}} \,, \tag{5}$$

with the augmented channel matrix $\underline{\mathbf{H}} = [\mathbf{H}^{\mathrm{H}}\ \sigma_n \mathbf{I}_{N_{\mathrm{I}}}]^{\mathrm{H}}$ and the augmented receive vector $\underline{\mathbf{y}} = [\mathbf{y}^{\mathrm{H}}\ \mathbf{0}_{1,N_{\mathrm{I}}}]^{\mathrm{H}}$ [8].

In order to reduce the required communication overhead of forwarding all local observations to the central node being usually far away, distributed estimation approaches are of particular interest. If $N_{\mathrm{R}} \geq N_{\mathrm{I}}$ holds, also local linear estimation can be performed with respect to (1) leading to the local ZF estimate $\tilde{\mathbf{x}}_{j,\mathrm{ZF}} = \mathbf{H}_j^{+}\mathbf{y}_j$ or MMSE estimate $\tilde{\mathbf{x}}_{j,\mathrm{MMSE}} = \underline{\mathbf{H}}_j^{+}\underline{\mathbf{y}}_j$ at node j. However, in general these local estimates will be differ at the different RAPs and lead to worse performance compared to the central solution.

III. Distributed Detection Schemes

A. DiCE Algorithm

In [4], the authors proposed an approach that solves the LS problem (3) in a distributed fashion following the idea in [9]. This is facilitated by reformulating the optimization problem into a set of optimizations over local estimates $\mathbf{x}_j$ with coupling through a consensus constraint $\mathbf{x}_i = \mathbf{x}_j$ for $i \in \mathcal{N}_j$. This constraint enforces equality between the local variable of node j and the variables of its neighboring nodes $i \in \mathcal{N}_j$. However, due to the direct coupling of the variables $\mathbf{x}_j, j \in \mathcal{J}$, this set of estimation problems cannot be solved in parallel as is. Introducing auxiliary variables $\mathbf{z}_j$ per node j, an iterative, distributed solution employing the Alternating Direction Method of Multipliers (ADMM) [10] becomes possible, resulting in the update equations in iteration k for every node $j \in \mathcal{J}$ [4]:

$$\mathbf{z}_j^k = \frac{\mu}{|\mathcal{N}_j^+|} \sum_{i \in \mathcal{N}_j^+} \left[\tfrac{1}{\mu}\mathbf{x}_i{}^{k-1} - \boldsymbol{\lambda}_{ij}^{k-1} \right] \tag{6a}$$

$$\boldsymbol{\lambda}_{ji}^k = \boldsymbol{\lambda}_{ji}^{k-1} - \tfrac{1}{\mu}\left(\mathbf{x}_j^{k-1} - \mathbf{z}_i^k\right) \qquad \forall i \in \mathcal{N}_j^+ \tag{6b}$$

$$\boldsymbol{\lambda}_{ij}^k = \boldsymbol{\lambda}_{ij}^{k-1} - \tfrac{1}{\mu}\left(\mathbf{x}_i^{k-1} - \mathbf{z}_j^k\right) \qquad \forall i \in \mathcal{N}_j \tag{6c}$$

$$\mathbf{x}_j^k = \left(\mathbf{H}_j^{\mathrm{H}}\mathbf{H}_j + \frac{|\mathcal{N}_j^+|}{\mu}\mathbf{I}_{N_{\mathrm{I}}}\right)^{-1} \cdot \left[\mathbf{H}_j^{\mathrm{H}}\mathbf{y}_j + \sum_{i \in \mathcal{N}_j^+}\left(\frac{\mathbf{z}_i^k}{\mu} + \boldsymbol{\lambda}_{ji}^k\right)\right] \,. \tag{6d}$$

Here, $\mathcal{N}_j^+ = \mathcal{N}_j \cup \{j\}$ contains the neighboring nodes of RAP j and itself. Variables $\mathbf{x}_j^k$ and $\mathbf{z}_j^k$ represent intermediate estimates at node j after iteration k, and $\boldsymbol{\lambda}_{ji}^k$ and $\boldsymbol{\lambda}_{ij}^k$ denote Lagrangian multipliers. The optimization of the step size μ is intended in further studies and is currently fixed to 1 in numerical evaluations.

Initializing the variables $\mathbf{z}_j^0$ and $\boldsymbol{\lambda}_{ij}^0$ with zeros, the initial estimate $\mathbf{x}_j^0$ at node j depends only on the local observation $\mathbf{y}_j$ and the channel matrix $\mathbf{H}_j$ following (6d):

$$\mathbf{x}_j^0 = \left(\mathbf{H}_j^{\mathrm{H}}\mathbf{H}_j + \frac{|\mathcal{N}_j^+|}{\mu}\mathbf{I}_{N_{\mathrm{I}}}\right)^{-1} \cdot \mathbf{H}_j^{\mathrm{H}}\mathbf{y}_j \,. \tag{7}$$

This expression can be interpreted as filtering $\mathbf{y}_j$ with a regularized inverse. Each iteration k starts with the update of the auxiliary variables $\mathbf{z}_j^k$ according to (6a) which are then shared with the neighbouring nodes. After this exchange, the Lagrangian multipliers are updated using (6b) and (6c) and, finally, the update of $\mathbf{x}_j^k$ takes place. This intermediate estimate $\mathbf{x}_j^k$ is again exchanged with the neighbors $i \in \mathcal{N}_j$. Thus, in each iteration the local estimates $\mathbf{x}_j^k$ and the auxiliary variables $\mathbf{z}_j^k$ are updated by considering information from the neighboring nodes and the previous own estimates, i.e., incorporating information from all nodes $i \in \mathcal{N}_j^+$.

Generally, the Lagrangian multipliers $\boldsymbol{\lambda}_{ji}$ and $\boldsymbol{\lambda}_{ij}$ required for the update equations (6d) and (6a) can be calculated for all $i \in \mathcal{N}_j^+$ locally at node j using (6b) and (6c) as the required variables $\mathbf{x}_i^{k-1}$ and $\mathbf{z}_i^k$ have been forwarded by the neighbors $i \in \mathcal{N}_j$. However, in case of erroneous inter-node links (i.e., errors on the BH channels indicated by the edges $\mathcal{E}$) these exchanged variables are corrupted differently at the RAPs prohibiting the convergence of the DiCE algorithm [11]. Nevertheless, a simple modification leads to convergence in the mean sense. To this end, only the Lagrangian $\boldsymbol{\lambda}_{ji}^k$ for $i \in \mathcal{N}_j$ are calculated at node j and transmitted to the neighbors as well. Correspondingly, node j collects the Lagrangian $\boldsymbol{\lambda}_{ij}^k$ from the other RAPs such that (6c) is obsolete. Subsequently, we will always consider this more general implementation of the DiCE algorithm.

In each iteration of the DiCE algorithm node j has to exchange the variables $\mathbf{x}_j^k$, $\mathbf{z}_j^k$, and $\boldsymbol{\lambda}_{ji}^k$ with its neighboring nodes leading to considerable communication overhead as elaborated in Section IV-C. In order to reduce the total communication overhead one may either reduce the number of iterations while maintaining performance or reduce the number of signals to be exchanged within an iteration. Both approaches are tackled by the modifications discussed subsequently.

B. Fast-DiCE Algorithm

The Fast-DiCE algorithm proposed in [7] introduces a prediction step for the auxiliary variables $\mathbf{z}_i^k$ and the Lagrangian $\boldsymbol{\lambda}_{ji}^k$ to accelerate the iterative detection algorithm by adopting Nesterov's optimal gradient descend method [12]. The main idea is to calculate at node j a *predictor* $\tilde{\mathbf{z}}_{ji}^k$ for the auxiliary variable of node i based on the two latest received estimates $\mathbf{z}_i^k$ and $\mathbf{z}_i^{k-1}$ by $\tilde{\mathbf{z}}_{ji}^k = \mathbf{z}_i^k + \gamma_k \left(\mathbf{z}_i^k - \mathbf{z}_i^{k-1}\right)$. Thus, the newest received estimate $\mathbf{z}_i^k$ is *extended* by the gradient of the auxiliary variable $\mathbf{z}_i^k - \mathbf{z}_i^{k-1}$ weighted by the step size parameter γ_k. Similarly, the predictors $\tilde{\boldsymbol{\lambda}}_{ji}^k$ are calculated based on $\boldsymbol{\lambda}_{ji}^k$ and $\boldsymbol{\lambda}_{ji}^{k-1}$. These predictors are then used for calculating the local variables leading to modified update equations at node j

$$\mathbf{z}_j^k = \frac{\mu}{|\mathcal{N}_j^+|} \sum_{i \in \mathcal{N}_j^+} \left[\tfrac{1}{\mu}\mathbf{x}_i{}^{k-1} - \tilde{\boldsymbol{\lambda}}_{ij}^{k-1}\right] \tag{8a}$$

$$\boldsymbol{\lambda}_{ji}^k = \tilde{\boldsymbol{\lambda}}_{ji}^{k-1} - \tfrac{1}{\mu}\left(\mathbf{x}_j^{k-1} - \mathbf{z}_i^k\right) \qquad \forall i \in \mathcal{N}_j^+ \tag{8b}$$

$$\tilde{\mathbf{z}}_{ji}^k = \mathbf{z}_i^k + \gamma_k \left(\mathbf{z}_i^k - \mathbf{z}_i^{k-1}\right) \qquad \forall i \in \mathcal{N}_j^+ \tag{8c}$$

$$\tilde{\boldsymbol{\lambda}}_{ji}^k = \boldsymbol{\lambda}_{ji}^k + \gamma_k \left(\boldsymbol{\lambda}_{ji}^k - \boldsymbol{\lambda}_{ji}^{k-1}\right) \qquad \forall i \in \mathcal{N}_j^+ \tag{8d}$$

$$\mathbf{x}_j^k = \left(\mathbf{H}_j^{\mathrm{H}}\mathbf{H}_j + \frac{|\mathcal{N}_j^+|}{\mu}\mathbf{I}_{N_{\mathrm{I}}}\right)^{-1} \cdot \left[\mathbf{H}_j^{\mathrm{H}}\mathbf{y}_j + \sum_{i \in \mathcal{N}_j^+}\left(\frac{\tilde{\mathbf{z}}_{ji}^k}{\mu} + \tilde{\boldsymbol{\lambda}}_{ji}^k\right)\right]. \tag{8e}$$

The predictors are initialized as $\tilde{\mathbf{z}}_{ji}^0 = \tilde{\boldsymbol{\lambda}}_{ji}^0 = \mathbf{0}$ and the step size parameter in iteration k is given by

$$\gamma_k = \frac{\alpha_{k-1} - 1}{\alpha_k} \quad \text{and} \quad \alpha_k = \frac{1 + \sqrt{1 + 4(\alpha_{k-1})^2}}{2}. \tag{9}$$

with $\alpha^0 = 1$. In the first iteration, the Fast-DiCE equals DiCE as no prediction step is possible. However, for $k \geq 2$ the estimates $\mathbf{z}_i^{k-1}$ and $\boldsymbol{\lambda}_{ji}^{k-1}$ from last iteration can be used together with the latest estimates $\mathbf{z}_i^k$ and $\boldsymbol{\lambda}_{ji}^k$ to calculate the predictors $\tilde{\mathbf{z}}_{ji}^k$ and $\tilde{\boldsymbol{\lambda}}_{ji}^k$ according to (8c) and (8d), respectively. Similar to the DiCE algorithm, the auxiliary variable $\mathbf{z}_j^k$ and the multipliers $\boldsymbol{\lambda}_{ji}^k$ are calculated at node j and delivered to the neighbors $i \in \mathcal{N}_j$. After receiving these exchanged variables, predictors $\tilde{\mathbf{z}}_{ji}^k$ and $\tilde{\boldsymbol{\lambda}}_{ji}^k$ are then calculated at every RAP $j \in \mathcal{J}$ improving the estimation of $\mathbf{x}_j^k$ as well as $\mathbf{z}_j^{k+1}$ and $\boldsymbol{\lambda}_{ji}^{k+1}$ in the following step. Unlike other variables, the predictors are calculated locally and do not have to be exchanged among RAPs, such that the communication overhead per iteration compared to DiCE remains the same. Nevertheless, as demonstrated in Section IV, fewer iterations are required by Fast-DiCE to achieve the same estimation quality as DiCE leading to a considerable reduction of the overall communication effort at the expense of a slightly higher computational complexity.

C. Reduced Overhead DiCE

As mentioned, the variables $\mathbf{x}_j$, $\mathbf{z}_j$, and $\boldsymbol{\lambda}_{ji}$ have to be transmitted by node j to its neighboring RAPs $i \in \mathcal{N}_j$ in each iteration of the DiCE algorithm. Obviously, this data transmission causes a high communication overhead among the RAPs as detailed in Section IV-C. In order to reduce this overhead, the Reduced Overhead DiCE (RO-DiCE) proposed in [6] introduces the approximations

$$\frac{1}{|\mathcal{N}_j^+|} \sum_{i \in \mathcal{N}_j^+} \boldsymbol{\lambda}_{ij}^{k-1} \approx \boldsymbol{\lambda}_{jj}^{k-1} \approx \frac{1}{|\mathcal{N}_j^+|} \sum_{i \in \mathcal{N}_j^+} \boldsymbol{\lambda}_{ji}^{k-1} \tag{10}$$

in the update equation for the auxiliary variable $\mathbf{z}_j^k$ in (6a) and the estimate $\mathbf{x}_j^k$ in (6d). Thus, the sums of multipliers are approximated by the locally available multiplier $\boldsymbol{\lambda}_{jj}^{k-1}$ which omits the exchange of multipliers among nodes at all. This approximation is motivated by a relaxed optimization problem, which only achieves consensus in the mean sense. For RO-DiCE the update equations for RAP j are given by

$$\mathbf{z}_j^k = -\mu\boldsymbol{\lambda}_{jj}^{k-1} + \frac{1}{|\mathcal{N}_j^+|} \sum_{i \in \mathcal{N}_j^+} \mathbf{x}_i^{k-1} \tag{11a}$$

$$\boldsymbol{\lambda}_{jj}^k = \boldsymbol{\lambda}_{jj}^{k-1} - \tfrac{1}{\mu}(\mathbf{x}_j^{k-1} - \mathbf{z}_j^k) \tag{11b}$$

$$\mathbf{x}_j^k = \left(\mathbf{H}_j^{\mathrm{H}}\mathbf{H}_j + \frac{|\mathcal{N}_j^+|}{\mu}\mathbf{I}_{N_{\mathrm{I}}}\right)^{-1} \cdot \left[\mathbf{H}_j^{\mathrm{H}}\mathbf{y}_j + |\mathcal{N}_j^+|\boldsymbol{\lambda}_{jj}^k + \tfrac{1}{\mu}\sum_{i \in \mathcal{N}_j^+}\mathbf{z}_i^k\right]. \tag{11c}$$

Note that in DiCE each node j forwards different multipliers $\boldsymbol{\lambda}_{ji}^k$ to each of its neighbors $i \in \mathcal{N}_j$ in a unicast fashion. Thus, avoiding the exchange of multipliers reduces the communication overhead significantly. In contrast, the intermediate estimates $\mathbf{z}_j^k$ and $\mathbf{x}_j^k$ could be broadcasted in principle leading to smaller burden. In the next section we will investigate the communication overhead in detail for different BH topologies based on unicast and broadcast transmissions. Furthermore, the performance degradation caused by the systematic error due to the approximation (10) will be investigated.

IV. Performance Evaluation

A. Bit Error Rate

In order to investigate the performance of the presented distributed estimation algorithms in a small cell scenario, Monte-Carlo simulations for a system with $N_{\mathrm{RAP}} = 4$ RAPs each equipped with $N_{\mathrm{R}} = 2$ receive antennas serving $N_{\mathrm{UE}} = 2$ UEs with $N_{\mathrm{T}} = 2$ transmit antennas have been performed.

Fig. 2 shows the average bit error rate (BER) over all RAPs for uncoded QPSK transmissions over i.i.d. Rayleigh fading channels, where the central ZF (cZF) and MMSE (cMMSE) solutions are shown for reference. Here, a fully meshed BH network is assumed with either ideal or noisy BH connections, where AWGN channels with fixed $\mathrm{SNR}_{\mathrm{BH}} = 30$ dB are assumed. The DiCE algorithms were terminated after $N_{\mathrm{It}} = 20$ iterations, leading to an error floor. In general, this error floor can be decreased by allowing more iterations as convergence of the DiCE algorithm to the cental solution is guaranteed [4]. In case of error-free BH links only a small loss is visible for RO-DiCE due to the introduced approximation (10). However,

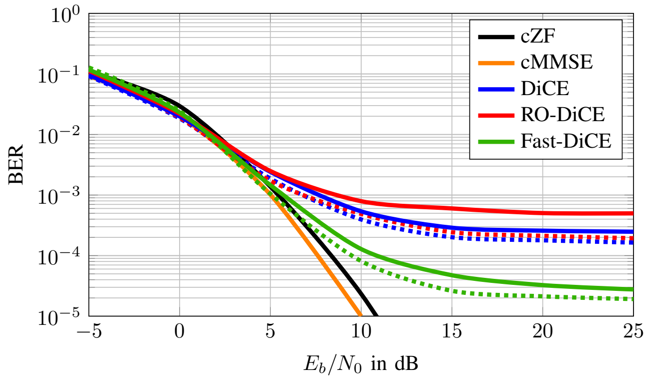

Fig. 2. Average BER over all RAPs for a system with $N_{\mathrm{RAP}} = 4$, $N_{\mathrm{R}} = 2$, $N_{\mathrm{UE}} = 2$, $N_{\mathrm{T}} = 2$ using QPSK modulation, with perfect (····) and noisy (—) inter-node links, $N_{\mathrm{It}} = 20$ iterations for DiCE algorithms.

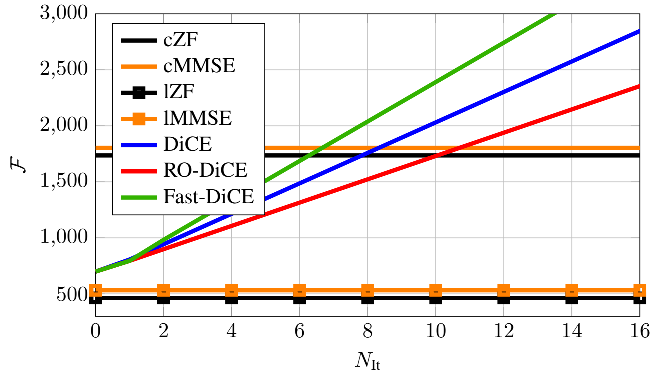

Fig. 3. Floating points operations per node for a system with $N_{\mathrm{RAP}} = 4$, $N_{\mathrm{R}} = 2$, $N_{\mathrm{UE}} = 2$, $N_{\mathrm{T}} = 2$.

the loss increases for disturbed BH links as consensus is only achieved in the mean sense. The Fast-DiCE outperforms both DiCE and RO-DiCE significantly because of its accelerated convergence due to the applied prediction step. Thus, with the same number of iterations the error floor is reduced by an order of magnitude.

B. Computational Complexity

In Table I we list the number of floating point operations $\mathcal{F}$ (FLOPS) of the linear equalization approaches, DiCE, Fast-DiCE and RO-DiCE per processing node. The complexity of the central linear equalization schemes depends on the number of input signals $N_{\mathrm{I}} = N_{\mathrm{UE}} N_{\mathrm{T}}$, the number of output signals $N_{\mathrm{O}} = N_{\mathrm{RAP}} N_{\mathrm{R}}$ as well as the frame length L. In contrast, the number of output signals for local linear equalization (lZF and lMMSE) and DiCE reduces to N_{R}. The complexity of the DiCE approaches further depend on the number of iterations N_{It} as well as the number of neighboring nodes per RAP j given by $|\mathcal{N}_j|$.

TABLE I
NUMBER OF FLOATING POINT OPERATIONS PER NODE

Scheme	FLOPS $\mathcal{F}$
cZF	$\frac{2}{3}N_{\mathrm{I}}^3 + (6N_{\mathrm{O}} - \frac{1}{2})N_{\mathrm{I}}^2 + (\frac{4}{3} - \frac{3}{2}N_{\mathrm{O}})N_{\mathrm{I}}$
cMMSE	$2N_{\mathrm{I}}^3 + (6N_{\mathrm{O}} - 2)N_{\mathrm{I}}^2 + (3 - \frac{3}{2}N_{\mathrm{O}})N_{\mathrm{I}} + 4N_{\mathrm{I}}N_{\mathrm{O}}L$
lZF	$\frac{2}{3}N_{\mathrm{I}}^3 + (6N_{\mathrm{R}} - \frac{1}{2})N_{\mathrm{I}}^2 + (\frac{4}{3} - \frac{3}{2}N_{\mathrm{R}})N_{\mathrm{I}} + 4LN_{\mathrm{I}}N_{\mathrm{R}}$
lMMSE	$2N_{\mathrm{I}}^3 + (6N_{\mathrm{R}} - 2)N_{\mathrm{I}}^2 + (3 - \frac{3}{2}N_{\mathrm{O}})N_{\mathrm{I}} + 4N_{\mathrm{I}}N_{\mathrm{R}}L$
DiCE	$\frac{4}{3}N_{\mathrm{R}}^3 + (6N_{\mathrm{I}}+5)N_{\mathrm{R}}^2 + (2N_{\mathrm{I}}^2 - \frac{3}{2}N_{\mathrm{I}} + 4LN_{\mathrm{I}} - L - \frac{17}{6})N_{\mathrm{R}}$ $+(4LN_{\mathrm{It}} - 4L - \frac{1}{2})N_{\mathrm{I}}^2 + (6\lvert\mathcal{N}_j\rvert + 6)LN_{\mathrm{It}}N_{\mathrm{I}}$ $-(8\lvert\mathcal{N}_j\rvert + 8)LN_{\mathrm{I}} + 1$
Fast-DiCE	$\frac{4}{3}N_{\mathrm{R}}^3 + (6N_{\mathrm{I}}+5)N_{\mathrm{R}}^2 + (2N_{\mathrm{I}}^2 - \frac{3}{2}N_{\mathrm{I}} + 4LN_{\mathrm{I}} - L - \frac{17}{6})N_{\mathrm{R}}$ $+(4LN_{\mathrm{It}} - 4L - \frac{1}{2})N_{\mathrm{I}}^2 + (8\lvert\mathcal{N}_j\rvert + 12)LN_{\mathrm{It}}N_{\mathrm{I}}$ $-(12\lvert\mathcal{N}_j\rvert + 20)LN_{\mathrm{I}} + 1$
Ro-DiCE	$\frac{4}{3}N_{\mathrm{R}}^3 + (6N_{\mathrm{I}}+5)N_{\mathrm{R}}^2 + (2N_{\mathrm{I}}^2 - \frac{3}{2}N_{\mathrm{I}} + 4LN_{\mathrm{I}} - L - \frac{17}{6})N_{\mathrm{R}}$ $+(4LN_{\mathrm{It}} - 4L - \frac{1}{2})N_{\mathrm{I}}^2 + (2\lvert\mathcal{N}_j\rvert + 6)LN_{\mathrm{It}}N_{\mathrm{I}}$ $-(2\lvert\mathcal{N}_j\rvert + 7)LN_{\mathrm{I}} + 1$

Fig. 3 depicts the number of FLOPS of the DiCE approaches with varying number of iterations for the introduced system configuration and a full meshed network, i.e., $|\mathcal{N}_j| = N_{\mathrm{RAP}} - 1 = 3$. For comparison, the complexity of the linear local and centralized equalizations are depicted as well.

In the first iteration, the complexity of DiCE is comparable to the complexity of local MMSE. For each iteration k, DiCE is updating the auxiliary variables $\mathbf{z}_j^k$, $\boldsymbol{\lambda}_{ji}^k$ and $\mathbf{x}_j^k$ based on the update equations (6) leading to an increased complexity. For this system setup, the complexity per node of DiCE approaches the complexity of the linear equalization in a central node at approximately $k = 8$ iterations. Furthermore, the complexity of RO-DiCE compared to DiCE is remarkably reduced, as only the local Lagrangian $\boldsymbol{\lambda}_{jj}^k$ is considered for the update of $\mathbf{z}_j^k$ in (11a) and $\mathbf{x}_j^k$ in (11c), respectively. In contrast, Fast-DiCE increases the complexity per iteration due to the prediction steps (8c) and (8d). However, as shown in the following subsection, Fast-DiCE requires fewer iterations to achieve the same estimation performance.

C. Communication Overhead

As detailed above, the information exchange on the links between RAPs contains variables which need to be made available to all neighbors and variables which are specific to certain neighbors. In case of omnidirectional wireless inter-node communication, the broadcast nature allows to transmit the intermediate estimates $\mathbf{z}_j^k$ and $\mathbf{x}_j^k$ conveniently to all neighbors $i \in \mathcal{N}_j$, while the exchange of $\boldsymbol{\lambda}_{ji}^k$ requires dedicated transmissions to each neighbor (unicast). In such a scenario, the RO-DiCE offers significant reductions due to the omitted exchange of Lagrangian multipliers. However, in case of point-to-point (P2P) links between RAPs only unicast transmissions are possible. Thus, for investigating the actual communication overhead the applied BH approach has to be considered. To this end, we will consider an arbitrary network topology with N_{RAP} nodes and logical BH topology defined by $\mathcal{E}$ (e.g., a full mesh) which is realized by different physical variants. As examples, we discuss point-to-point (P2P) links (e.g., mmWave, fiber or copper connections between RAPs), point-to-multi-point (P2MP) links (e.g. omnidirectional wireless transmissions), and a physical star topology where the communication between RAPs is routed over a transport node (TN) as depicted in Fig. 4.

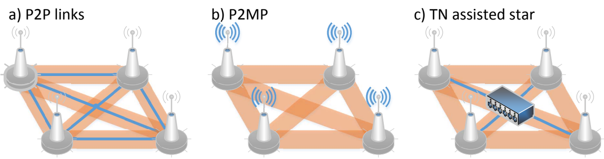

Fig. 4. Backhaul topologies for $N_{\mathrm{RAP}} = 4$ RAPs.

In case of P2P links, in each DiCE iteration the RAP j has to transmit the local variables to each of its neighbor $i \in \mathcal{N}_j$ and receives the current variables of that RAP leading to an overall link load of $\mathrm{D}^{\mathrm{P2P}}_{j\to i} = 6N_{\mathrm{I}}L$ for this particular inter-node link between RAP j and RAP i. In case of P2MP links, RAP j has to transmit $\mathrm{D}^{\mathrm{P2MP}}_{j\to\mathcal{N}_j} = (2 + |\mathcal{N}_j|)N_{\mathrm{I}}L$ signals and the overall broadcast rate equals $\mathrm{D}^{\mathrm{P2MP}} = N_{\mathrm{RAP}}\mathrm{D}^{\mathrm{P2MP}}_{j\to\mathcal{N}_j}$. In case of RO-DiCE only reduced link load $\mathrm{D}^{\mathrm{P2P}}_{j\to i} = 4N_{\mathrm{I}}L$ or $\mathrm{D}^{\mathrm{P2MP}}_{j\to\mathcal{N}_j} = 2N_{\mathrm{I}}L$ have to be supported.

Table II lists for DiCE and RO-DiCE the total amount of exchanged variables per iteration considering different physical BH topologies. For an arbitrary physical topology (i.e., each logical link is realized by dedicated physical link) the total number of exchanged signals are defined for P2P and P2MP connections in the first two rows demonstrating a significant reduction for P2MP transmission. Regarding the case of a TN, we assume that the hub is able to multicast packets to multiple destinations. Thus, variables $\mathbf{x}_j^k$ and $\mathbf{z}_j^k$ have to be transmitted only once to the TN, which then duplicates this information when forwarding to the logical neighbors $i \in \mathcal{N}_j$.

TABLE II
TOTAL COMMUNICATION OVERHEAD FOR DIFFERENT TOPOLOGIES

PHY Topology	Links	$\mathrm{D}_{\mathrm{DiCE}}$	$\mathrm{D}_{\mathrm{Ro\text{-}DiCE}}$
Arbitrary	P2P	$6\lvert\mathcal{E}\rvert N_{\mathrm{I}}L$	$4\lvert\mathcal{E}\rvert N_{\mathrm{I}}L$
Arbitrary	P2MP	$2(\lvert\mathcal{E}\rvert + N_{\mathrm{RAP}})N_{\mathrm{I}}L$	$2N_{\mathrm{RAP}}N_{\mathrm{I}}L$
TN	P2P	$(2N_{\mathrm{RAP}} + 8\lvert\mathcal{E}\rvert)N_{\mathrm{I}}L$	$(2N_{\mathrm{RAP}} + 4\lvert\mathcal{E}\rvert)N_{\mathrm{I}}L$

Fig. 5 shows the average BER versus the total communication overhead assuming the same system configuration as before for a fixed $E_{\mathrm{b}}/N_0 = 10$ dB and $\mathrm{SNR}_{\mathrm{BH}} = 30$ dB. Notice, that the error rates after $k = \{1, 5, 10, 15, 20\}$ iterations are labeled by markers. Depicted are the results for P2P links, P2MP links and star topology for realizing a logical full-meshed network.

The graphs indicate, that the RO-DiCE reduces significantly the communication overhead compared to the original DiCE approach. However, due to the introduced approximation the estimation performance is slightly deteriorated considering the same number of iterations. Fast-DiCE and DiCE require the same communication overhead per iteration, but Fast-DiCE is able to achieve the same BER with fewer iterations leading to a reduced communication overhead as well. For example, to achieve a BER of 10^{-3} in case of P2P links the Fast-DiCE requires on average 10 iterations, whereas the DiCE approach needs about 16 iterations. Contrary, the approximation introduced for RO-DiCE derogates the performance and requires on

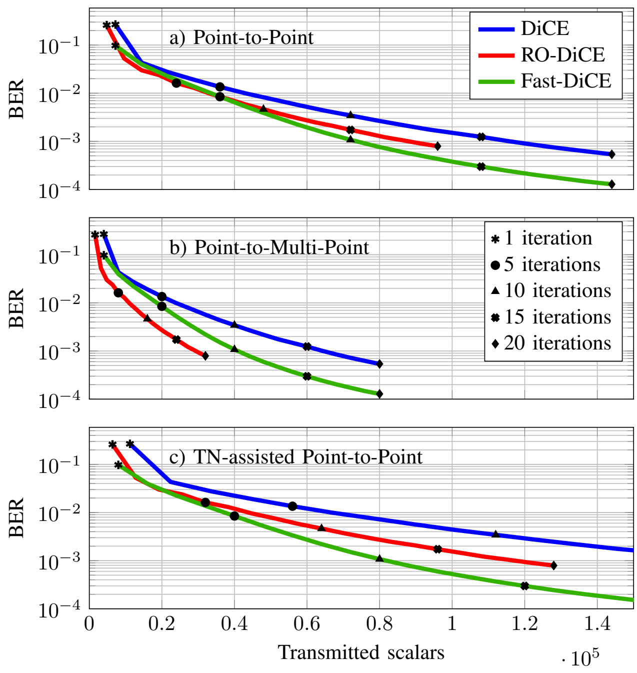

Fig. 5. Total communication overhead for a) physical unicast, b) physical broadcast, and c) multicast star topology.

average 18 iterations for the same BER. Thus, in case of P2P BH links, the Fast-DiCE algorithm achieves the best overhead-performance trade-off.

When considering P2MP links, a significant reduction of communication overhead is visible for all approaches as the variables $\mathbf{x}_j^k$ and $\mathbf{z}_j^k$ can be broadcasted. As costly unicast transmissions of Lagrangian multipliers are avoided by RO-DiCE, this approach leads here to the best trade-off. Finally, the TN-assisted topology performs similarly to the arbitrary topology with P2P links with only slightly increased overhead on a reduced number of physical links.

V. CONCLUSION

For a small cell network we proposed the Distributed Consensus-based Estimation (DiCE) algorithm for cooperatively solving optimization problems. We discussed the two DiCE modifications Fast-DiCE and Reduced Overhead DiCE (RO-DiCE) and compared the performance of these schemes by means of error rate, computational complexity and communication overhead when applied for multi-user detection. However, as the considered problem is quite general, the presented framework can be applied also for other tasks in distributed systems. In the future, these approaches will be compared with respect to quantization effects on the backhaul links, latency requirements, and combined with forward error correction techniques.

VI. ACKNOWLEDGEMENT

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 317941. The authors would like to acknowledge the contributions of their colleagues in iJOIN, although the views expressed are those of the authors and do not necessarily represent the project.

REFERENCES

[1] P. Rost, C. Bernardos, A. D. Domenico, M. D. Girolamo, M. Lalam, A. Maeder, D. Sabella, and D. Wübben, "Cloud Technologies for Flexible 5G Radio Access Networks," *IEEE Communications Magazine*, vol. 52, no. 5, May 2014.

[2] NGMN, "Suggestions on Potential Solutions to C-RAN by NGMN Alliance," NGMN, Tech. Rep., Jan. 2013.

[3] D. Wübben, P. Rost, J. Bartelt, M. Lalam, V. Savin, M. Gorgoglione, A. Dekorsy, and G. Fettweis, "Benefits and Impact of Cloud Computing on 5G Signal Processing," *IEEE Signal Processing Magazine*, Nov 2014.

[4] H. Paul, J. Fliege, and A. Dekorsy, "In-Network-Processing: Distributed Consensus-Based Linear Estimation," *IEEE Communications Letters*, vol. 17, no. 1, pp. 59–62, Jan. 2013.

[5] H. Paul, B.-S. Shin, D. Wübben, and A. Dekorsy, "In-Network-Processing for Small Cell Cooperation in Dense Networks," in *IEEE VTC2013-Fall Workshop (CLEEN 2013)*, Las Vegas, USA, Sep. 2013.

[6] B.-S. Shin, H. Paul, D. Wübben, and A. Dekorsy, "Reduced Overhead Distributed Consensus-Based Estimation Algorithm," in *IEEE Globecom Workshop (IWCPM 2013)*, Atlanta, USA, Dec. 2013.

[7] G. Xu, H. Paul, D. Wübben, and A. Dekorsy, "Fast Distributed Consensus-based Estimation for Cooperative Wireless Sensor Networks," in *18th International ITG Workshop on Smart Antennas (WSA 2014)*, Erlangen, Germany, Mar. 2014.

[8] D. Wübben, R. Böhnke, V. Kühn, and K.-D. Kammeyer, "MMSE Extension of V-BLAST based on Sorted QR Decomposition," in *IEEE 58th Semiannual Vehicular Technology Conference (VTC2003-Fall)*, Orlando, FL, USA, Oct. 2003.

[9] H. Zhu, A. Cano, and G. Giannakis, "Distributed Consensus-Based Demodulation: Algorithms and Error Analysis," *IEEE Transactions on Wireless Communications*, vol. 9, no. 6, pp. 2044–2054, Jun. 2010.

[10] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, USA: Athena Scientific, 1997.

[11] H. Paul, B.-S. Shin, and A. Dekorsy, "Distributed Consensus-Based Linear Estimation with Erroneous Links," in *17th International ITG Workshop on Smart Antennas (WSA 2013)*, Stuttgart, Germany, Mar. 2013.

[12] Y. Nesterov, "A Method of Solving a Convex Programming Problem with Convergence Rate $O(1/k^2)$," *Soviet. Math. Dokl.*, vol. 27, pp. 372–376, 1983.