User Activity And Data Detection For MIMO Uplink C-RAN Using Bayesian Learning

Anupama Rajoriya, Vidushi Katiyar, Rohit Budhiraja
Dept. of Electrical Engineering IIT Kanpur, Kanpur, India
{anupama, vidushi, rohitbr}@iitk.ac.in

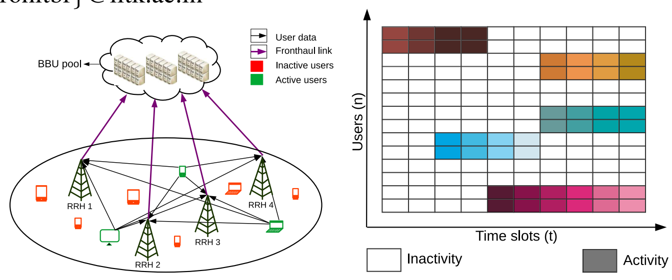
***Abstract*—We investigate user activity and data detection problem in a multiple-input multiple-output uplink cloud-radio access network, where the data matrix over a time-frame has overlapped burst sparsity due to sporadic user activity. We exploit this sparsity to recover data by proposing a weighted prior-sparse Bayesian learning algorithm. The proposed algorithm, due to carefully selected prior, captures not only the overlapped burst sparsity across time but also the block sparsity due to multi-user antennas. We also derive hyperparameter updates, and estimate the weight parameters using the support estimated via index-wise log-likelihood ratio test. We numerically demonstrate that the proposed algorithm has much lower bit error rate than the state-of-the-art competing algorithms.**

***Index Terms*—Cloud-radio access network (C-RAN), compressive sensing, sparse Bayesian learning (SBL).**

I. Introduction

The cloud-radio access network (C-RAN) is being recognized as a next generation system paradigm to provide high spectral and energy efficiency. The C-RAN architecture, as shown in Fig. 1a, consists of a baseband processing unit (BBU) pool, multiple remote radio heads (RRHs), and capacity-limited fronthaul links connecting them. The BBU pool performs signal processing tasks, while RRHs, with radio frequency (RF) components, transmits/receives the RF signals to/from the geographically distributed users [1]. The C-RANs can efficiently provide massive machine-type communication (mMTC) due to its vast geographical coverage [2].

In a multiple input multiple-output (MIMO) uplink C-RAN, only few MIMO mMTC users are active at a particular time instant [2]. This makes the uplink data from the MIMO mMTC users *block sparse*. Further, the mMTC users transmit their information in a frame consisting of multiple time slots [3] and their activity is sporadic – they can start and stop transmitting data any where in the frame [4]. This also leads to *overlapped burst sparsity* across the time slots in a frame. Fig. 1b shows aforementioned *block sparsity* and *overlapped burst sparsity* for $K = 7$ mMTC users, each with $M = 2$ antennas (corresponding to 2 rows in users axis), transmitting in a frame length of $T = 10$ time slots. This sparse nature of data can be exploited to recover it using advanced compressive sensing techniques [3], [5]–[8]. The authors in [5] exploited both, data and uplink channel sparsity to jointly detect active users and estimate channel in a C-RAN. This work, however, considered high-capacity fronthaul links between BBU pool and RRHs, which is impractical. Reference [6] proposed a compressive sampling matching pursuit based alternating algorithm to mitigate the interference and recover data. This scheme is highly sensitive to the choice of dictionary matrix, and requires the knowledge of sparsity level of the sparse vector. All of these works, crucially, considered single time slot transmission. The mMTC users, however, transmit data over multiple time slots in a frame.



(a) (b)

Fig. 1: (a) A C-RAN architecture with a BBU, fronthaul links and the RRHs. Red and green users are inactive users and active ones, respectively; (b) Device activity with time for $M = 2$ user antennas.

References [3], [8] extended the system model from a single-slot to multi-slot transmission for an uplink grant-free non-orthogonal multiple access (NOMA) system. The authors in [3] proposed joint data and user activity detection algorithm for an uplink grant-free NOMA system. This algorithm, based on approximate message passing (AMP), uses the prior information of the transmit symbols. Reference [8] proposed a threshold-aided block sparsity adaptive subspace pursuit (SP) algorithm, in which the uplink data matrix-recovery problem is transformed into its vector-recovery counterpart, with the vectors having the block sparsity. This, however, radically increases their solution complexity.

These algorithms assume joint sparsity across the time-frame but fail to capture the overlapped burst activity of users. Further, none of the existing algorithms consider MIMO user, and thus do not capture the block sparsity across the multiple user antennas. The **main** contributions of this paper to overcome the limitations of the aforementioned works, and can be summarized as follows.

- We propose a weighted prior-sparse Bayesian learning (WP-SBL) algorithm for MIMO uplink C-RAN model which jointly detects the user activity and data by exploiting its *overlapped burst sparse* structure across time and *block sparsity* across user antennas.
- We propose novel updates for the weights in the proposed prior to capture the overlapped burst sparsity of the data matrix. We estimate the user activity profile using index-wise log-likelihood ratio tests (LLRTs), which we further use to update the weights. We also derive sub-optimal updates for the hyperparameters, whose optimal updates are intractable.
- We numerically show that by capturing and appropriately modeling different sparsities using the prior, and with the novel weight updates, the proposed WP-SBL algorithm achieves lower bit error rate (BER) than them.

II. System model and problem formulation

We consider, as shown in Fig. 1a, a MIMO uplink C-RAN system with K M-antenna users and R N-antenna RRHs, which are connected to the BBUs with limited-capacity fronthaul links. The green users in Fig. 1a are active users, and the red ones are inactive. The data transmission phase consists of T time slots, and we aim to detect the active users $K_a \ll K$ users and the data sent by them.

Let the transmit signal of the kth user at the tth time slot be $\mathbf{s}_k^t = [s_{k,1}^t, \ldots, s_{k,M}^t] \in \mathbb{C}^M$ with $\mathbb{E}|s_{k,m}^t|^2 = 1$ for $k \in [K]$ and $m \in [M]$, where the notation $[K]$ represents the set $\{1, \ldots, K\}$, and $[M]$ represents the set $\{1, \ldots, M\}$. The received signal at the tth time slot at the rth RRH, denoted as $\mathbf{y}_r^t \in \mathbb{C}^N$, is given as

$$\mathbf{y}_r^t = \sqrt{P_s} \sum_{k=1}^{K} \mathbf{H}_{r,k}^t \mathbf{s}_k^t + \mathbf{n}_r^t = \sqrt{P_s} \tilde{\mathbf{H}}_r^t \mathbf{s}^t + \mathbf{n}_r^t. \quad (1)$$

The matrix $\mathbf{H}_{r,k}^t \in \mathbb{C}^{N \times M}$ is the frequency domain channel between the rth RRH and the kth user. The vector $\mathbf{n}_r^t \in \mathbb{C}^N$ is the Gaussian noise distributed as $\mathcal{CN}(\mathbf{0}, (\sigma_r^t)^2 \mathbf{I}_N)$. The concatenated matrix $\tilde{\mathbf{H}}_r^t \triangleq \left[\tilde{\mathbf{H}}_{r,1}^t, \ldots, \tilde{\mathbf{H}}_{r,K}^t\right] \in \mathbb{C}^{N \times MK}$ and the concatenated vector $\mathbf{s}^t \triangleq [\mathbf{s}_1^t; \ldots; \mathbf{s}_K^t] \in \mathbb{C}^{MK}$. The constant P_s is the signal power. The channel matrix $\mathbf{H}_{r,k}^t$, for all $k \in [K]$, models both small and large scale fading. It is thus expressed as $\mathbf{H}_{r,k}^t = \mathbf{G}_{r,k}^t d_{r,k}^{1/2}$, where $\mathbf{G}_{r,k}^t$ is the small scaling fading, and each of its entry is $\mathcal{CN}(0, 1)$. The scalar $d_{r,k}$ represents the distance-dependent large scale path loss between the rth RRH and the kth user.

We assume that each RRH, due to the capacity-constrained fronthaul links between RRH and BBU, compresses its receive signal using a compression matrix $\mathbf{A}_r^t = \tilde{\mathbf{S}}_r^t \mathbf{F} \in \mathbb{C}^{\gamma N \times N}$ [6]. Here the scalar $\gamma \in (0, 1)$ is the compression factor, $\tilde{\mathbf{S}}_r^t \in \{0, 1\}^{\gamma N \times N}$ is the selection matrix formed by randomly selecting γN rows of an $N \times N$ identity matrix, and $\mathbf{F}$ is an $N \times N$ DFT matrix [9]. The compressed receive signal at the BBU pool, denoted as $\tilde{\mathbf{y}}_r^t \in \mathbb{C}^{\gamma N}$, sent from the rth RRH is

$$\tilde{\mathbf{y}}_r^t = \mathbf{A}_r^t \mathbf{y}_r^t = \sqrt{P_s} \mathbf{A}_r^t \tilde{\mathbf{H}}_r^t \mathbf{s}^t + \mathbf{A}_r^t \mathbf{n}_r^t, \forall t \in [T],$$

where $[T]$ denotes the set $\{1, \ldots, T\}$. Note that due to choice of the compression matrix, the equivalent noise $\tilde{\mathbf{n}}_r^t = \mathbf{A}_r \mathbf{n}_r^t$ is still $\mathcal{CN}(\mathbf{0}, (\sigma_r^t)^2 \mathbf{I}_N)$. The concatenated received signal $\tilde{\mathbf{y}}^t = [\tilde{\mathbf{y}}_1^t; \ldots; \tilde{\mathbf{y}}_R^t] \in \mathbb{C}^{\gamma NR}$ is

$$\tilde{\mathbf{y}}^t = \tilde{\mathbf{\Phi}}^t \mathbf{s}^t + \tilde{\mathbf{n}}^t, \forall t \in [T]. \quad (2)$$

The matrix
$$\tilde{\mathbf{\Phi}}^t \triangleq \sqrt{P_s} \begin{bmatrix} \mathbf{A}_1^t \tilde{\mathbf{H}}_1^t \\ \vdots \\ \mathbf{A}_R^t \tilde{\mathbf{H}}_R^t \end{bmatrix} \in \mathbb{C}^{\gamma NR \times MK} \quad (3)$$

is the signal measurement matrix corresponding to tth time slot. The concatenated noise vector $\tilde{\mathbf{n}}^t \triangleq [\tilde{\mathbf{n}}_1^t; \ldots; \tilde{\mathbf{n}}_R^t] \in \mathbb{C}^{\gamma NR}$ is distributed as $\mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I}_{\gamma NR})$, where we assume that $(\sigma_r^t)^2 = \sigma^2$, for all $r \in [R]$, $t \in [T]$. We assume that the channel remains constant over the entire frame [8], and we thus have $\mathbf{H}_r^t = \mathbf{H}_r$ and $\tilde{\mathbf{\Phi}}^t = \tilde{\mathbf{\Phi}}, \forall t \in [T]$. Concatenating the observations across the time instances, we get the complete observation $\mathbf{Y} = [\tilde{\mathbf{y}}^1, \ldots, \tilde{\mathbf{y}}^T] \in \mathbb{C}^{\gamma R \times T}$ at the BBU pool:

$$\mathbf{Y} = \tilde{\mathbf{\Phi}} \mathbf{S} + \mathbf{N}. \quad (4)$$

Here matrices $\mathbf{S} \triangleq [\mathbf{s}^1, \ldots, \mathbf{s}^T] \in \mathbb{C}^{MK \times T}$ and $\mathbf{N} \triangleq [\tilde{\mathbf{n}}^1, \ldots, \tilde{\mathbf{n}}^T] \in \mathbb{C}^{\gamma NR \times T}$.

In a 5G IoT/mMTC random access system [4], out of the K users, only few ($K_a \ll K$) are active at a particular time instant [2]. Also, if the kth user is active, $\mathbf{s}_k^t \in \mathbb{C}^M$ is non-zero, which makes the vector $\mathbf{s}^t$ in (1), block sparse. Further, the user activity in a frame is sporadic [10], i.e., a IoT/mMTC user can start and stop transmitting data any where in the frame. The data symbol matrix $\mathbf{S}$ thus has *overlapping burst sparsity* across time in addition to *block sparsity* across antennas, which is depicted in Fig. 1b for $K = 7$ users with $M = 2$ antennas each and frame length $T = 10$. The proposed algorithm exploits both these sparsities to estimate active user's data.

We assume, similar to [6], [11], that the BBU pool has complete channel information of all the users in the system. We assume that the block-faded channel, with coherence time greater than the frame duration T, can be estimated by the BBU in the channel estimation phase, when all the users transmit pilots to the BBU. The BBU applies CS techniques to estimate channels of all the users [5].

III. Sparse Bayesian learning based data recovery

We now develop Bayesian learning framework to detect the user activity and their data. This framework assumes that the unknown data is generated from a prior distribution which captures its burst sparsity across different time and block sparsity across antennas. Using this prior and the observations (likelihood), the posterior distribution of unknown data is evaluated. To achieve this aim, we assign a weighted Gaussian prior on the unknown data matrix $\mathbf{S}$, such that the data sent from the kth user in tth time slot $\mathbf{s}_k^t$ is generated from

$$p(\mathbf{s}_k^t | \bar{\boldsymbol{\alpha}}_k, \mathbf{W}_k) = \prod_{m=1}^{M} \mathcal{CN}\left(s_{k,m}^t | 0, \left(\sum_{q=1}^{T} \mathbf{W}_k(t, q) \alpha_k^q\right)^{-1}\right)$$
$$= \prod_{m=1}^{M} \frac{(\mathbf{W}_k(t, :)\bar{\boldsymbol{\alpha}}_k)}{\pi} \exp\left(-\mathbf{W}_k(t, :)\bar{\boldsymbol{\alpha}}_k |s_{k,m}^t|^2\right). \quad (5)$$

Here α_k^q represents the hyperparameter of the kth user, associated with the time slot q. The scalar $\mathbf{W}_k(t, q)$ represents the weight/importance of α_k^q in the tth data vector $\mathbf{s}_k^t$. The row vector $\mathbf{W}_k(t, :) = [\mathbf{W}_k(t, 1), \ldots, \mathbf{W}_k(t, T)] \in \mathbb{R}_+^{1 \times T}$, and the column vector $\bar{\boldsymbol{\alpha}}_k = [\alpha_k^1, \ldots, \alpha_k^T]$ denote the weight and the hyperparameter vectors, respectively. The overall prior distribution of the data matrix $\mathbf{S}$ thus becomes

$$p(\mathbf{S} | \bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}}) = \prod_{t=1}^{T} \prod_{k=1}^{K} \prod_{m=1}^{M} p(s_{k,m}^t | \bar{\boldsymbol{\alpha}}_k, \mathbf{W}_k(t, :))$$
$$= \prod_{t=1}^{T} \mathcal{CN}(\mathbf{s}^t | \mathbf{0}_{MK}, \mathbf{\Lambda}^t), \quad (6)$$

where $\bar{\boldsymbol{\alpha}} \triangleq \{\bar{\boldsymbol{\alpha}}_1, \ldots, \bar{\boldsymbol{\alpha}}_K\}$, $\overline{\mathbf{W}} \triangleq \{\overline{\mathbf{W}}_1, \ldots, \overline{\mathbf{W}}_K\}$, $\mathbf{\Lambda}^t \triangleq \text{diag}(\lambda_1^t \mathbf{1}_M; \ldots; \lambda_K^t \mathbf{1}_M) \in \mathbb{R}_+^{MK}$ and $\lambda_k^t \triangleq (\mathbf{W}_k(t, :)\bar{\boldsymbol{\alpha}}_k)^{-1}$.

Intuition of weights: We consider, as given in (5), a Gaussian prior with zero mean and the variance, which is the inverse of weighted sum of local hyperparamters of data from all the time slots. An intuition behind the weights and the choice of hyperparameters can be explained via a toy example. We consider a user k, which transmits data in $t = 2, 3, 6$ time slots out of total $T=6$ time slots. The sparsity profile of the user is shown in Fig. 2. The prior distribution of $\mathbf{s}_k^2$ thus depends on

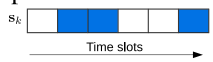


Fig. 2: Activity pattern of user k for $T=6$.

$\{\mathbf{s}_k^3, \mathbf{s}_k^6\}$ and vice-versa. To fully exploit the shared sparsity, we assign a prior on the kth user's data at the tth time slot, $\mathbf{s}_k^t$, which couples all the time slot's hyperparameters $\alpha_k^q, \forall q$ together through the weights $\mathbf{W}_k(t,q), \forall q$. For example, for the user activity pattern in Fig. 2, the overall weight matrix should be as follows

$$\mathbf{W}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 0 & 0 & 1/3 \\ 0 & 1/3 & 1/3 & 0 & 0 & 1/3 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1/3 & 1/3 & 0 & 0 & 1/3 \end{bmatrix}. \quad (7)$$

We note that for the inactive time slots, i.e., $\forall t \in \{1,4,5\}$, $\mathbf{W}_k(t,t)=1$ and $\mathbf{W}_k(t,q)=0, \forall q \neq t$. This case thus becomes equivalent to conventional SBL in [7], as the prior for these indices becomes independent Gaussian distribution. For the active time slots, we exploit their shared sparse structure and assign equal weights to the all the non-zero time slots. In other words, for all $t \in \{2,3,6\}$, the weights $\mathbf{W}_k(t,q)=1/3$, for all $q \in \{2,3,6\}$, and $\mathbf{W}_k(t,q)=0$, for all $q \in \{1,4,5\}$.

We next assign, similar to [7], a Gamma distribution as a hyperprior over the hyperparameters α_k^t, which is given as

$$p(\alpha_k^t) = \text{Gamma } (\alpha_k^t | c,d) = \Gamma(c)^{-1} d^c (\alpha_k^t)^c e^{-d\alpha_k^t}. \quad (8)$$

Here $\Gamma(c)=\int_0^\infty t^{c-1}e^{-t}dt$ is the Gamma function. This hierarchical choice of priors (Gaussian-Gamma) becomes equivalent to Student-t distribution and thus results in the sparse solution [7]. We assign smaller values of order 10^{-4} to d and large value to c (e.g. $c=1$), similar to [12], which encourages large value of α_k^t, and thus promotes the sparsity.

We now develop a Bayesian learning based algorithm for the proposed choice of hierarchical prior. The likelihood distribution $p(\mathbf{Y}|\mathbf{S})$ for the observation matrix $\mathbf{Y} \triangleq [\tilde{\mathbf{y}}^1, \ldots, \tilde{\mathbf{y}}^T]$ can be written as

$$p(\mathbf{Y}|\mathbf{S}) = \prod_{t=1}^T p(\tilde{\mathbf{y}}^t|\mathbf{s}^t) = \prod_{t=1}^T \pi^{-\gamma NR} \exp\left(-||\tilde{\mathbf{y}}^t - \tilde{\mathbf{\Phi}}^t \mathbf{s}^t||^2/\sigma^2\right).$$

For fixed values of weights $\overline{\mathbf{W}}$, and the hyperparameters $\bar{\boldsymbol{\alpha}}$, the posterior distribution of $\mathbf{S}$ can be obtained as

$$p(\mathbf{S}|\mathbf{Y}, \bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}}) = \prod_{t=1}^T p(\mathbf{s}^t|\tilde{\mathbf{y}}^t, \bar{\boldsymbol{\alpha}}, , \overline{\mathbf{W}}). \quad (9)$$

Here

$$p(\mathbf{s}^t|\tilde{\mathbf{y}}^t, \bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}}) \propto p(\tilde{\mathbf{y}}^t|\mathbf{s}^t) p(\boldsymbol{s}^t|\bar{\boldsymbol{\alpha}}, \mathbf{W}) = \mathcal{CN}(\boldsymbol{\mu}^t, \boldsymbol{\Sigma}^t). \quad (10)$$

The vector $\boldsymbol{\mu}^t \in \mathbb{C}^{MK}$ and the matrix $\boldsymbol{\Sigma}^t \in \mathbb{C}^{MK \times MK}$ are the posterior mean and covariance matrix of $\mathbf{s}^t$ given by

$$\boldsymbol{\mu}^t = \sigma^{-2} \boldsymbol{\Sigma}^t (\tilde{\mathbf{\Phi}}^t)^H \tilde{\mathbf{y}}^t; \boldsymbol{\Sigma}^t = \left(\sigma^{-2} (\tilde{\mathbf{\Phi}}^t)^H \tilde{\mathbf{\Phi}}^t + \mathbf{\Lambda}^t\right)^{-1}, \quad (11)$$

respectively [7]. The matrix $\mathbf{\Lambda}^t = \text{diag}(\lambda_1^t \mathbf{1}_M; \ldots; \lambda_K^t \mathbf{1}_M) \in \mathbb{R}_+^{MK}$ is the variance matrix. The hyperparameters $\bar{\boldsymbol{\alpha}}_k$, and optimal weights $\overline{\mathbf{W}}_k$ are unknown, which are estimated next.

Hyperparameter Estimation: We now evaluate the maximum a posteriori estimate of the hyperparameters $\bar{\boldsymbol{\alpha}}_k, \forall k \in [K]$, by maximizing the posterior distribution $p(\bar{\boldsymbol{\alpha}}|\mathbf{Y})$

$$\hat{\alpha}_k^t = \arg\max_{\alpha_k^t} \log p(\bar{\boldsymbol{\alpha}}|\mathbf{Y}). \quad (12)$$

We employ expectation maximization (EM) algorithm to iteratively maximize its lower bound, $\mathbb{E}\left[\log p(\bar{\boldsymbol{\alpha}}, \mathbf{S}|\mathbf{Y}, \overline{\mathbf{W}})\right]$, where $\mathbb{E}(\cdot)$ denotes the expectation with respect to the posterior $p(\mathbf{S}|\mathbf{Y}, \bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}})$ [12]. In the ith iteration:

E-step: the posterior, $p(\mathbf{S}|\mathbf{Y}, \hat{\bar{\boldsymbol{\alpha}}}^{(i-1)}, \overline{\mathbf{W}})$ given by (10), is calculated, where $\hat{\bar{\boldsymbol{\alpha}}}^{(i-1)}$ are the hyperparameter estimates from the previous iteration.

M-step: the lower bound $\mathbb{E}\left[\log p(\bar{\boldsymbol{\alpha}}, \mathbf{S}|\mathbf{Y}, \overline{\mathbf{W}})\right]$ is maximized with respect to $\bar{\boldsymbol{\alpha}}$. We next derive the M step to update the hyperparameters $\bar{\boldsymbol{\alpha}}$, as follows

$$\begin{aligned} \hat{\alpha}_k^{t,(i)} &= \arg\max_{\alpha_k^t \geq 0} \mathbb{E}\left[\log p(\bar{\boldsymbol{\alpha}}, \mathbf{S}|\mathbf{Y}, \overline{\mathbf{W}})\right] \\ &\overset{(a)}{=} \arg\max_{\alpha_k^t \geq 0} \mathbb{E}\left[\log\left(\prod_{k=1}^K \prod_{m=1}^M \prod_{t=1}^T p\left(s_{k,m}^t | \bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}}\right)\right) p(\alpha_k^t)\right] \\ &\overset{(b)}{=} \arg\max_{\alpha_k^t \geq 0} \sum_{k=1}^K \sum_{m=1}^M \sum_{t=1}^T \log\big(\sum_{q=1}^T (\mathbf{W}_k(t,q)\alpha_k^q)\big) \\ &\quad - \big(\sum_{q=1}^T \mathbf{W}_k(t,q)\alpha_k^q\big) \mathbb{E}[||\mathbf{s}_k^t||^2] + Mc\log\alpha_k^t - Md\alpha_k^t. \quad (13) \end{aligned}$$

Equality (a) is obtained using $p(\bar{\boldsymbol{\alpha}}, \mathbf{S}|\mathbf{Y}, \mathbf{W}) \propto \prod_{k=1}^K \prod_{m=1}^M \prod_{t=1}^T p(s_{k,m}^t | \bar{\boldsymbol{\alpha}}_k, \mathbf{W}_k) p(\alpha_k^t)$. Equality (b) is derived by using the prior distributions $p(\mathbf{s}_k^t|\boldsymbol{\alpha})$ and $p(\alpha_k^t)$ from (5) and (8), respectively. We next differentiate (13) with respect to α_k^t and set to zero, to get

$$\frac{Mc}{\alpha_k^t} - Md + \sum_{j=1}^T \frac{M\mathbf{W}_k(j,t)}{\sum_{q=1}^T \mathbf{W}_k(j,q)\alpha_k^q} - \sum_{j=1}^T \mathbf{W}_k(j,t)\mathbb{E}||\mathbf{s}_k^j||^2 = 0$$

The optimal point α_k^t, thus satisfies

$$\frac{Mc}{\alpha_k^t} + \sum_{j=1}^T \frac{M\mathbf{W}_k(j,t)}{\sum_{q=1}^T \mathbf{W}_k(j,q)\alpha_k^q} = Md + \sum_{j=1}^T \mathbf{W}_k(j,t)\mathbb{E}||\mathbf{s}_k^j||^2. \quad (14)$$

We notice that a closed form expression for the optimal hyperparameter α_k^t is difficult to calculate. We, thus, similar to [12], calculate a sub-optimal solution of (14). Since the hyperparameters α_k^t and the weights $\mathbf{W}_k(t,j), \forall k \in [K], t,j \in [T]$ are non-negative, we have

$$\begin{aligned} &\mathbf{W}_k(j,t)\alpha_k^t < \sum_{q=1}^T \mathbf{W}_k(j,q)\alpha_k^q \\ \implies &\frac{1}{\mathbf{W}_k(j,t)\alpha_k^t} > \frac{1}{\sum_{q=1}^T \mathbf{W}_k(j,q)\alpha_k^q} > 0. \quad (15) \end{aligned}$$

Multiplying the non-negative scalar $M\mathbf{W}_k(j,t)$ and summing over all $j \in [T]$, we get

$$\frac{MT}{\alpha_k^t} \geq M \sum_{j=1}^T \frac{\mathbf{W}_k(j,t)}{\sum_{q=1}^T \mathbf{W}_k(j,q)\alpha_k^q} \geq 0. \quad (16)$$

Adding Mc/α_k^t to all sides, we get the lower and upper bounds on the L.H.S. in (14) as follows

$$\frac{Mc+MT}{\alpha_k^t} \geq \frac{Mc}{\alpha_k^t} + M\sum_{j=1}^T \frac{\mathbf{W}_k(j,t)}{\sum_{q=1}^T \mathbf{W}_k(j,q)\alpha_k^q} \geq \frac{Mc}{\alpha_k^t}. \quad (17)$$

Using (14) and (17), we get the bounds on the R.H.S. in (14)

$$\frac{Mc+MT}{\alpha_k^t} \geq Md + \sum_{j=1}^T \mathbf{W}_k(j,t)\mathbb{E}||\mathbf{s}_k^j||^2 \geq \frac{Mc}{\alpha_k^t}, \quad (18)$$

which results in

$$\hat{\alpha}_k^t \in \left[\frac{Mc}{Md+\sum_{j=1}^{T}\mathbf{W}_k(j,t)\mathbb{E}||\mathbf{s}_k^j||^2}, \frac{Mc+MT}{Md+\sum_{j=1}^{T}\mathbf{W}_k(j,t)\mathbb{E}||\mathbf{s}_k^j||^2}\right]. \quad (19)$$

Using (19), we chose a sub-optimal solution to (14) [12]

$$\hat{\alpha}_k^t = \frac{Mc}{Md+\sum_{j=1}^{T}\mathbf{W}_k(j,t)\mathbb{E}[||\mathbf{s}_k^j||^2]}, \forall k \in [K], t \in [T]. \quad (20)$$

Reference [12] proposed pattern-coupled SBL to recover block sparse signals which assigned a hierarchical Gaussian prior over the block sparse vector. The prior for each entry, along with its own hyperparameter, depended on its immediate neighbor hyperparameters also. The work considered block-sparse vector recovery problem, unlike the current problem of user activity and data matrix recovery with burst-sparsity in time and block sparsity across antennas, in an uplink C-RAN MIMO system. Also, it can only capture the block sparsity of size three (i.e., itself and its two neighbors) and the weights are assumed to be constant over the EM iterations. As discussed in paragraph after (6), any random choice of the weights, similar to [12], will not be able to capture the burst sparsity among different time instances. We next propose a novel method to estimate the weights in each EM iteration.

Computation of weights: We discussed in paragraph after (6) that the weight $\mathbf{W}_k(t,q)$ quantifies the relevance of the kth user's data in qth time slot $\mathbf{s}_k^q$, in the recovery of its data in the tth time slot $\mathbf{s}_k^t$. Fixing the values of the weights will make it difficult for the algorithm to capture the hidden burst-sparsity structure. We, therefore, propose a rule to update weights, along with the updates of posterior parameters of $\mathbf{S}$ and the hyperparameters $\bar{\boldsymbol{\alpha}}$, given by (11) and (20), respectively. We also discussed in paragraph after (6) that the optimal weights depend on the sparsity structure of the data matrix $\mathbf{S}$. The weight $\mathbf{W}_k(t,q), \forall q \in [T]$, used to recover the data $\mathbf{s}_k^t$, is zero if $\mathbf{s}_k^q$ and $\mathbf{s}_k^t$ do not share any sparsity, and non-zero otherwise. Equivalently, if the binary variable β_k^t denotes the hard support (activity/inactivity) of $\mathbf{s}_k^t$, i.e., $\beta_k^t = \mathbb{I}(\mathbf{s}_k^t \neq \mathbf{0}_M)$, where the notation $\mathbb{I}(\cdot)$ denotes the indicator function, the optimal weights $\mathbf{W}_k(t,q), \forall q \in [T]$, have the form

$$\text{if } \beta_k^t = 1: \;\; \mathbf{W}_k(t,q) = \frac{\beta_k^q}{\sum_{j=1}^{T}\beta_k^j}, \quad (21)$$

$$\text{if } \beta_k^t = 0: \;\; \mathbf{W}_k(t,q) = \mathbb{I}(q=t). \quad (22)$$

To update the weights, we therefore require the true hard support, denoted by $\beta_k^t, \forall k \in [K], \forall t \in [T]$. The authors in [13] proposed index wise LLRTs to detect the non-zero support of the sparse vector, and to fasten the convergence of the EM algorithm for sparse vector recovery. Reference [14] used the LLRT-based support detection to reduce the number of message exchanges between different nodes in a distributed network, for joint sparse vector recovery. We, unlike [13] and [14], detect sparse data matrix, with burst and block sparsity. We employ the LLRTs to detect the non-zero support (activity pattern) of the data matrix $\mathbf{S}$, which we will use to update the weights $\overline{\mathbf{W}}$. The two hypotheses in this LLRT for the kth user data at tth time slot are defined as [14]

$$\mathcal{H}_0: \mathbf{s}_k^t = \mathbf{0}_M; \quad \mathcal{H}_1: \; \mathbf{s}_k^t \neq \mathbf{0}_M. \quad (23)$$

From the definition of the prior in (6), we have $\mathbf{s}_k^t = \mathbf{0}$ if $\lambda_k^t = 0$ and $\mathbf{s}_k^t \neq \mathbf{0}_M$ if $\lambda_k^t > 0$. The two hypotheses in (23) are thus equivalent to $\lambda_k^t = 0$ and $\lambda_k^t > 0$. We decide in favour of one of the hypotheses by comparing the ratio of the two marginalized likelihood distributions $p(\tilde{\mathbf{y}}^t|\mathcal{H}_1)$ and $p(\tilde{\mathbf{y}}^t|\mathcal{H}_0)$ with a detection threshold [14]. The marginal likelihood $p(\tilde{\mathbf{y}}^t|\bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}}) = p(\tilde{\mathbf{y}}^t|\boldsymbol{\Lambda}^t)$ has the following form

$$p(\tilde{\mathbf{y}}^t|\boldsymbol{\Lambda}^t) = \int p(\tilde{\mathbf{y}}^t|\mathbf{s}^t)p(\mathbf{s}^t|\bar{\boldsymbol{\alpha}}, \overline{\mathbf{W}})d\mathbf{s}^t \overset{(a)}{=} \mathcal{CN}(\mathbf{0}_{\gamma NR}, \sigma^2\mathbf{I}_{\gamma NR} + \tilde{\boldsymbol{\Phi}}^t\boldsymbol{\Lambda}^t(\tilde{\boldsymbol{\Phi}}^t)^H). \quad (24)$$

The LLRT for the nth index, for all $n \in [N_s]$, where $N_s = MK$ in the tth time slot thus reduces to

$$\log\frac{p(\tilde{\mathbf{y}}^t|\mathcal{H}_1)}{p(\tilde{\mathbf{y}}^t|\mathcal{H}_0)} = \log\frac{p(\tilde{\mathbf{y}}^t|\boldsymbol{\Lambda}^t)}{p(\tilde{\mathbf{y}}^t|\boldsymbol{\Lambda}_{s,-n}^t)} \geq \theta, \quad (25)$$

where θ is the detection threshold, the marginal likelihood $p(\tilde{\mathbf{y}}^t|\boldsymbol{\Lambda}^t)$ is given by (24), and the diagonal matrix $\boldsymbol{\Lambda}_{-n}^t \in \mathbb{R}_+^{N_s \times N_s}$ is such that $\boldsymbol{\Lambda}_{-n}^t(n,n) = 0$ and $\boldsymbol{\Lambda}_{-n}^t(n',n') = \boldsymbol{\Lambda}^t(n',n'), \forall n' \neq n \in [N_s]$. After substituting marginal likelihood from (24) and simplifying it, the index-wise support, which we denote as $b_n^t, \forall n \in [N_s]$ can be obtained as [14]

$$b_n^t = \mathbb{I}\left(\frac{\{(\tilde{\boldsymbol{\Phi}}_n^t)^H(\sigma^2\mathbf{I} + \tilde{\boldsymbol{\Phi}}^t\boldsymbol{\Lambda}_{-n}^t(\tilde{\boldsymbol{\Phi}}^t)^H)^{-1}\tilde{\mathbf{y}}^t\}^2}{(\tilde{\boldsymbol{\Phi}}_n^t)^H(\sigma^2\mathbf{I} + \tilde{\boldsymbol{\Phi}}^t\boldsymbol{\Lambda}_{-n}^t(\tilde{\boldsymbol{\Phi}}^t)^H)^{-1}\tilde{\boldsymbol{\Phi}}^t} \geq \bar{\theta}\right). \quad (26)$$

The normalized threshold $\bar{\theta}$ is computed as $\bar{\theta} = \left(\mathcal{Q}^{-1}\left(\frac{\eta}{2}\right)\right)^2$, where $\mathcal{Q}(\cdot)$ is standard Q-function and η is algorithm parameter [14]. As the activity/inactivity of a user is same for all its antennas $m \in [M]$, the binary support estimate, denoted as $\hat{\beta}_k^t, \forall k \in [K]$ can be computed as

$$\hat{\beta}_k^t = \mathbb{I}\left(\sum_{m=1}^{M} b_{(k-1)M+m}^t > \left\lceil\frac{(M-1)}{2}\right\rceil\right). \quad (27)$$

We update the weight parameters $\mathbf{W}_k(t,q)$ using the above hard-support estimate $\hat{\beta}_k^t, \forall t \in [T]$. Depending on the value of the binary variable $\hat{\beta}_k^t$, we propose two separate updates for the weight parameters.

If $\hat{\beta}_k^t = 1$: When the LLRT-based support detection results in $\hat{\beta}_k^t = 1$, it implies that the kth user is active at the tth time slot. Our goal in this case is to exploit the possibly existing burst or overlapped-sparsity across different time slots. As discussed in the example in paragraph after (6), the optimal update for this case, should be to give equal weights to all the active time slots, i.e., $\mathbf{W}_k(t,q) = \mathbf{W}_k(t,t)$ such that $\hat{\beta}_k^q = 1$. The weight updates $\mathbf{W}_k(t,q)$ are thus given by (21).

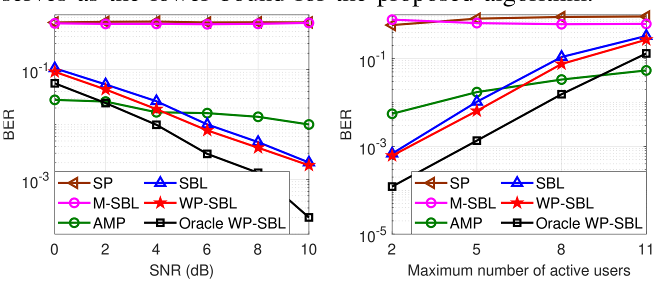
If $\hat{\beta}_k^t = 0$: When the support detection rule results in $\hat{\beta}_k^t = 0$ for the kth user at the tth time slot, we update the weights according to (22). In this case, we have $\hat{\beta}_k^t = 0$, which implies that the kth user is inactive at the tth time slot and no data is received corresponding to the tth time slot from the kth user. There is no coupling between s_k^t, data from kth user at tth time slot and $s_k^q, \forall q \neq t$ at the other time slots. We thus make the priors of different time-slots independent of each other, by updating the weights as given in (22).

Remark 1. We note that when the users are active at all time slots, data detection problem becomes jointly sparse matrix recovery problem [15]. Our algorithm then gives $\beta_k^t = 1, \forall t \in$

$[T]$, which results in $\mathbf{W}_k(t,q) = 1/T, \forall t, q$ from (21). *It is interesting to note that in this case, our algorithm for active users reduces to the well-known SBL based algorithm for joint sparse signal recovery, called multiple SBL (M-SBL) [15].*

IV. Simulation results

We now, similar to [3], compare the bit error rate (BER) of the proposed WP-SBL algorithm with following schemes (i) *SBL* [7]: estimates the vectors $\mathbf{s}^t, \forall t \in [T]$ by applying SBL on each of the vector; (ii) *M-SBL* [15]: estimates $\mathbf{S}$ by applying M-SBL and assuming the joint sparsity; (iii) *AMP* [3]: estimates $\mathbf{S}$ by assuming joint sparsity. The prior distribution assumed therein captures the information of the discrete data symbols; (iv) *SP* [8]: a threshold-aided adaptive SP, which estimates the data assuming the joint sparsity alone and; (v) *Oracle WP-SBL*: assumes that the true support of the data matrix $\beta_k^t, \forall k, t$, is known. It estimates the data by applying proposed WP-SBL algorithm using the knowledge of its true support. The BER of the oracle WP-SBL algorithm serves as the lower bound for the proposed algorithm.



(a) (b)

Fig. 3: BER obtained by varying (a) SNR; (b) active users K_a with $P_s = 8$ dB.

For this study, we consider a C-RAN system with total number of users $K = 25$ and number of RRH, $R = 10$. Each user and RRH are equipped with $M = N = 2$ antennas. The maximum number of active users is set as $K_a = 4$. We set the fronthaul compression factor $\gamma = 0.5$, the frame size, as in the 5G standard [4] as, $T = 14$ and the maximum burst size $B = 8$. The channel is generated by considering both, small and large scale fading [16]. We assume that the users and the RRHs are located uniformly in a region of $1000 \times 1000\ m^2$. The large scale fading coefficients are modeled as $d_{r,k} = \text{PL}_{r,k} 10^{\frac{z_{r,k}}{10}}$, where the term $10^{\frac{z_{r,k}}{10}}$ accounts for the shadow fading effects, with $z_{r,k} \sim \mathcal{N}(0, \sigma_{sh}^2)$ and $\text{PL}_{r,k}$ denotes the path loss between the rth RRH and kth user [16]. The path loss is modeled same as [16, Eq. (52)] and the frequency, distances and heights are taken same as that in [16]. We normalize the channel with the noise power $N_0 = -121.4$ dB and take $\sigma^2 = 1$. The SBL hyperparameters are initialized as $\alpha_k^t = 100, \forall k, \forall t$, the maximum number of EM iteration, $I_{\max} = 100$ and the stopping threshold is set as $\epsilon = 10^{-3}$, and $\eta = 10^{-2.5}$ [14]. The values of these parameters are same for all the following simulations.

We first plot in Fig. 3a the BER obtained by varying transmit SNR $= P_s$. We see that the proposed WP-SBL algorithm outperforms all the existing algorithms after SNR $= 4$ dB. This is because (i) the standalone SBL does not takes into account the burst nature of the user activity across the frame and; (ii) the M-SBL, AMP and SP algorithms wrongly model the burst sparsity as the joint/common sparsity across time slots. The AMP algorithm also models the burst sparsity as the joint sparsity, its discrete prior, however, results in better data recovery as compared to SP and M-SBL. The proposed algorithm, in contrast, considers the burst activity of users.

We next show in Fig. 3b the BER for different number of active users, K_a with $P_s = 8$ dB. We note as the number of active users K_a increases, the number of unknown data symbols to be recovered (MK_a) also increases. As the number of observations (γR) still remains same, the BER of all the algorithms degrade. The proposed WP-SBL algorithm has still lower BER than the algorithms.

V. Conclusion

We designed a weighted prior SBL algorithm to recover data in MIMO uplink C-RAN. We derived the updates of the hyperparameters, and proposed the novel weight updates using the index-wise LLRTs. We numerically showed that the proposed algorithm outperforms the state-of-the-art algorithms.

References

[1] M. Hadzialic, B. Dosenovic, M. Dzaferagic, and J. Musovic, "Cloud-RAN: Innovative radio access network architecture," 01 2013.

[2] Z. Utkovski, O. Simeone, T. Dimitrova, and P. Popovski, "Random access in C-RAN for user activity detection with limited-capacity fronthaul," *IEEE Signal Process. Lett.*, vol. 24, no. 1, pp. 17–21, Jan 2017.

[3] C. Wei, H. Liu, Z. Zhang, J. Dang, and L. Wu, "Approximate message passing-based joint user activity and data detection for NOMA," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 640–643, Mar 2017.

[4] S. Ahmadi, *5G NR: Architecture, Technology, Implementation, and Operation of 3GPP New Radio Standards*. Elsevier Science, 2019.

[5] X. Xu, X. Rao, and V. K. N. Lau, "Active user detection and channel estimation in uplink CRAN systems," in *2015 IEEE International Conf. on Comm. (ICC)*, Jun 2015, pp. 2727–2732.

[6] J. Liu, A. Liu, and V. K. N. Lau, "Compressive interference mitigation and data recovery in cloud radio access networks with limited fronthaul," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1437–1446, Mar 2017.

[7] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug 2004.

[8] Y. Du, C. Cheng, B. Dong, Z. Chen, X. Wang, J. Fang, and S. Li, "Block-sparsity-based multiuser detection for uplink grant-free NOMA," *IEEE Trans. on Wireless Commun.*, vol. 17, no. 12, pp. 7894–7909, Dec 2018.

[9] T. T. Do, L. Gan, N. H. Nguyen, and T. D. Tran, "Fast and efficient compressive sensing using structurally random matrices," *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 139–154, Jan 2012.

[10] I. Sousa, M. P. Queluz, and A. Rodrigues, "A survey on qoe-oriented wireless resources scheduling," *arXiv preprint arXiv:1705.07839*, 2017.

[11] H. Zhu and G. B. Giannakis, "Exploiting sparse user activity in multiuser detection," *IEEE Trans. on Commun.*, vol. 59, no. 2, pp. 454–465, Feb 2011.

[12] J. Fang, Y. Shen, H. Li, and P. Wang, "Pattern-coupled sparse Bayesian learning for recovery of block-sparse signals," *IEEE Trans. on Signal Process.*, vol. 63, no. 2, pp. 360–372, Jan 2015.

[13] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, AISTATS 2003, Key West, Florida, USA, Jan 3-6, 2003*, 2003.

[14] S. Khanna and C. R. Murthy, "Communication-efficient decentralized sparse Bayesian learning of joint sparse signals," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 617–630, Sep 2017.

[15] D. P. Wipf and B. D. Rao, "An empirical Bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Trans. on Signal Process.*, vol. 55, no. 7, pp. 3704–3716, 2007.

[16] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive mimo versus small cells," *IEEE Trans. on Wireless Commun.*, vol. 16, no. 3, pp. 1834–1850, Mar 2017.