SSDP - COM 500
Throughput Optimization in 5G Networks
Project Report

Saged Bounekhel, Franck Khayat, Salim Najib

I. Introduction

With the amount of innovation happening in wireless networks, 4G communication systems become less suitable for new uses that need high and instantaneous data rates (mission-critical services, massive Internet Of Things...). Consequently, 5G base stations have to use systems comprised of *antenna arrays* performing *spatial beamforming* to target specific locations and users with a higher data rate. Beamforming concentrates the energy of the antenna array in specific directions, making it an essential technique for transmitting data in 5G systems.

We explored, studied and implemented different beamforming strategies to compare them and find throughput optimizations: we implemented the *matched beamforming* strategy, being the most common beamforming strategy in 5G systems, and the *flexibeam* method, then we analyzed them on simulated and real data. We also studied further theoretical improvements with the *collaborative* beamforming strategy, then implemented and tested it on simulated data.

To know in which direction each base station should transmit data, we have to estimate the *directions of arrival* (DOA) of users' signals. To this end, we implemented the *MUSIC* algorithm for spectral line estimation.

In this report, we first present in Section II the context and theory behind the different beamforming strategies and the DOA estimation, then we show numerical results of these strategies in Section III, and we finally conclude and discuss limitations of our project in Section IV.

II. Models and methods

A. Receiver-transmitter setup

In our system, we consider S base stations with L omni-directional antennas. We initially focus on the case $S = 1$. Each antenna L_i has a position $p_i \in \mathbb{R}^2$, and transmits the same narrow-band signal $s(t) \in \mathbb{C}$. Without loss of generality, we assume the wavelength to be equal to 1 for mathematical conciseness. All the signals transmitted by the antennas will sum up coherently, producing a *beam-shape*. The general beamforming strategy is introducing for each antenna a gain $\gamma_i > 0$ and a phase delay $\phi_i \in [0, 2\pi[$, which can be expressed as a *beamforming weight* $w_i = \gamma_i e^{-j\phi_i} \in \mathbb{C}$.

When seen from a far field target with position $r \in \mathbb{S}^1$ at time $t \in \mathbb{R}$, the ideal signal will be

$$\begin{aligned} y(t,r) &= s(t)\left(\sum_{i=1}^{L} \gamma_i e^{j\phi_i} e^{-j2\pi\langle r, p_i\rangle}\right) \\ &= s(t)\left(\sum_{i=1}^{L} w_i^* e^{-j2\pi\langle r, p_i\rangle}\right) \\ &= s(t) b^*(r) \end{aligned}$$

with $b(r) = \sum_{i=1}^{L} w_i e^{j2\pi\langle r, p_i\rangle}$ being the antenna array beam-shape [1]. In practice, it is corrupted by additive white noise $n(t,r) \in \mathbb{C}$.

B. Matched spatial beamforming

Matched beamforming is the typical beamforming strategy used in 5G communications. It follows the general beamforming strategy and defines the beamforming weights as $w_i = e^{-j2\pi\langle r_0, p_i\rangle}$, with r_0 being the array's steering direction. The signal will be concentrated around the steering direction, so it has to change periodically to cover all user cluster positions.

C. Flexibeam

It is generally the case that there are multiple clusters of user positions, in directions which are far apart. By focusing the beamshape around the steering direction, matched beamforming introduces a trade-off between focused beamforming and spatial diversity. Flexibeam [2] is an analytical filtering method which avoids this optimization problem by allowing more flexibility in the beamshape which produces larger main lobes.

Mathematically, flexibeam considers a continuum of antennas transmitting the following signal when observed at position $p \in \mathbb{R}^2$ and time $t \in \mathbb{R}$:

$$x(t,p) = \int_{\mathbb{S}^1} s(t) e^{-j2\pi\langle r, p\rangle} dr \in \mathbb{C}$$

x is denoted as the *sensor* function. Let's assume $x(t,.) \in \mathcal{L}^2(\mathbb{R}^2)$. Rather than beamforming weights $w_i \in \mathbb{C}$ corresponding to each antenna, we now consider a beamforming weight function $w \in \mathcal{L}^2(\mathbb{R}^2)$ with values in $\mathbb{C}$, where $w(p)$ is the beamforming weight of the antenna at position $p \in \mathbb{R}^2$. The beamforming function y is thus also translated into continuous time: at time $t \in \mathbb{R}$, the ideal transmitted signal becomes:

$$
\begin{aligned}
z(t) &= \langle x(t,.), w\rangle_{\mathcal{L}^2(\mathbb{R}^2)} \\
&= \int_{\mathbb{R}^2} x(t,p) w^*(p) dp \\
&= \int_{\mathbb{R}^2} dp\ w^*(p) \int_{\mathbb{S}^1} s(t) e^{-j2\pi\langle r,p\rangle} dr \\
&= \int_{\mathbb{S}^1} dr\ s(t) \underbrace{\int_{\mathbb{R}^2} dp\ w^*(p) e^{-j2\pi\langle r,p\rangle}}_{=b^*(r)} \\
&= s(t) \int_{\mathbb{S}^1} b^*(r) dr
\end{aligned}
$$

and similarly the signal seen at a far field target $r \in \mathbb{S}^1$ at time $t \in \mathbb{R}$ is

$$
y(t,r) = s(t) b^*(r)
$$

Observe that $b(r) = \mathcal{F}_2^{-1}(w)(r) = \int_{\mathbb{R}^2} w(p) e^{j2\pi\langle r,p\rangle} dp$ is only defined on $r \in \mathbb{S}^1$ by the equation above. We extend it to $\mathbb{R}^2$, such that w and b now form a Fourier transform pair:

$$
w(p) = \mathcal{F}_2(b)(p) = \int_{\mathbb{R}^2} b(r) e^{-j2\pi\langle r,p\rangle}
$$

This way, finding w and finding b are equivalent.

We recover continuous-space matched beamforming as a particular case of the new setup, letting $b(r) = \delta(r - r_0)$ then sampling the corresponding $w(p) = e^{-j2\pi\langle r_0,p\rangle}$ where r_0 is the steering direction. The novelty is that b does not have to be a Dirac anymore, as it can be set to be any function with analytically tractable Fourier transform and thus with analytically tractable $w = \mathcal{F}_2(b)$. Thus we can for example set b as an indicator function of a ball

$$
b(r) = \mathcal{X}\left(\frac{r}{||r||_2} \in B(r_0, \Delta r)\right)
$$

emitting around the steering direction rather than trying to focus on it. More systematically, we can design a *preference function* over the plane $f : \mathbb{R}^2 \to \mathbb{R}$ emulating the density of users on $\mathbb{R}^2$. Once we have f, we can set b to be:

$$
b(r) = \frac{1}{\mu(r)} \int_{\Gamma_r} f(l) dl
$$

where p_0 is the antenna array's center: $p_0 = \frac{1}{L}\sum_{i=1}^{L} p_i$, $\Gamma_r = \{p_0 + tr \mid t > 0\}$ is the half-line going from p_0 in the direction $\frac{r}{||r||_2}$ and $\mu(r) = \int_{\Gamma_r} \frac{1}{||l-p_0||} dl$ is the line-attenuation coefficient of Γ_r going out of p_0. Thus b is set to be the radial projection of f. The question now becomes: how to find f?

D. Collaborative beamforming with flexibeam

Let's now move to the case $S \geq 1$ where S is the number of stations. The intuition is that with multiple stations, it should be possible to estimate f by some sort of triangulation. Collaborative beamforming [3] formalizes this idea.

We will derive collaborative beamforming in the receiver's case. Assume the system is comprised of S stations i.e antenna arrays, each with L antennas. Define the steering vector of the sth station towards user cluster q, $1 \leq q \leq Q$ coming from direction $r_q \in \mathbb{S}^1$:

$$
a_s(r_q) := a_{s,q} = \begin{bmatrix} e^{-j2\pi\langle p_{1,s}, r_q\rangle} & \dots & e^{-j2\pi\langle p_{L,s}, r_q\rangle} \end{bmatrix}^T \in \mathbb{C}^L
$$

where $p_{l,s}$ is the position of the lth antenna at station s. This defines a *steering matrix* $A_s \in \mathbb{C}^{L\times Q}$ at each station s. Define the received signal at station s to be $v_s(t) = A_s s(t) + n(t) \in C^L$ where $s(t) \in \mathbb{C}^L$ is the signal and $n(t) \in \mathbb{C}^L$ is noise of mean 0. Further assume that $s(t) \sim \mathcal{N}_L(0, \Sigma)$ and $n(t) \sim \mathcal{N}_L(0, \Sigma_n)$ where $\Sigma = \mathrm{diag}(\sigma_1^2, \dots, \sigma_L^2)$ and $\Sigma_n = \sigma_n^2 I_L$ and $s(t)$ and $n(t)$ are uncorrelated. Then it is easy to compute the autocorrelation of $v_s(t)$:

$$
R_{v_s} = \mathbb{E}\left(v_s(t) v_s(t)^H\right) = A_s \Sigma A_s^H + \Sigma_n
$$

When beamforming $v_s(t)$ with the beamforming weights $w_s \in \mathbb{C}^L$ we obtain the signal $y_s(t) = w_s^H v_s(t) \in \mathbb{C}$ with autocorrelation:

$$
R_{y_s} := R_s = \mathbb{E}\left(y_s(t) y_s(t)^*\right) = \sum_{q=1}^{Q} \sigma_q^2 |w_s^H a_{s,q}|^2 + \sigma_n^2 ||w_s||_2^2
$$

The collaborative aspect comes from correlating signals received by pairs of stations, eliminating noise. Indeed, if $s_1 \neq s_2$, then

$$
R_{s_1,s_2} = \mathbb{E}\left(y_{s_1}(t) y_{s_2}(t)^*\right) = \sum_{q=1}^{Q} \sigma_q^2 w_{s_1}^H a_{s_1,q} w_{s_2}^H a_{s_2,q} \in \mathbb{C}
$$

Not only is noise eliminated by correlating, but also phase and magnitude information from R_{s_1,s_2} can be used. This induces an optimization problem based on the cross-correlation information of the whole system, as follows.

Define the beamforming system matrix $W \in \mathbb{C}^{SL\times S}$ as the diagonal block matrix of the beamforming weights of all stations:

$$
W = \begin{bmatrix} w_1 & 0_L & 0_L & \dots & 0_L \\ 0_L & w_2 & 0_L & \dots & 0_L \\ 0_L & \vdots & & & \vdots \\ 0_L & 0_L & 0_L & \dots & w_S \end{bmatrix} \in \mathbb{C}^{SL\times S}
$$

where $0_L \in \mathbb{R}^L$ is the zero vector. Then define the system's steering matrix $A \in \mathbb{C}^{SL\times N_Q}$ where N_Q is an upper bound estimate on Q:

$$
A = \begin{bmatrix} A_1 \\ \vdots \\ A_S \end{bmatrix} \in \mathbb{C}^{SL\times N_Q}
$$

Define parameterizations of f on the grid: $\hat{f} \in \mathbb{R}^{N_Q}$, and estimates of the noise level σ_n^2: $\hat{\sigma}_n^2 \in \mathbb{R}_+$. Lastly, we can define $\hat{R} \in \mathbb{C}^{S \times S}$ [4] which is an empirical estimate of $R \in \mathbb{C}^{S \times S}$:

$$R = \mathbb{E}\left(y(t)y(t)^H\right) \in \mathbb{C}^{S \times S} \text{ where } y(t) = \begin{bmatrix} y_1(s) \\ \vdots \\ y_S(t) \end{bmatrix}$$

and, given samples $y_1, \ldots, y_N$ of $y(t)$:

$$\hat{R} = \frac{1}{N} \sum_{i=1}^{N} y_i y_i^H$$

Then we can define the following LASSO problem to find optimal $\hat{f}$ and $\hat{\sigma}_n^2$:

$$\arg\min_{\hat{f}, \hat{\sigma}_n^2} \left|\left| \hat{R} - W^H \left(A\mathrm{diag}\left(\hat{f}\right) A^H - \hat{\sigma}_n^2 I_{SL} \right) W \right|\right|_F^2 + \lambda ||\hat{f}||_1$$

This way, a sparse representation of f is recovered. Now, [3] has tested the approach for sparse sky image reconstruction with weights W computed through matched beamforming. As suggested but not investigated in [2], we will try the same approach with weights W obtained with flexibeam performed with a rough estimate $\tilde{f}$ of f, e.g placing a Gaussian around each DOA as shown in [2].

We can go one step further by showing each station only the clusters in f that are closest to it. Formally, once $\hat{f}$ is obtained, for each station s replace $\hat{f}$ by $\hat{f}\mathcal{X}_s$ where

$$\mathcal{X}_s(r) = \mathcal{X}(s \in \arg\min_{s'=1\ldots S} ||r - p_{s'}||_2)$$

i.e $\mathcal{X}_s(r)$ indicates whether r is closest to station s or not.

E. MUSIC for DOA estimation

We use *MUSIC* in order to estimate the antenna steering directions in the presented beamforming strategies (i.e. r_0 for matched beamforming, and r_q for collaborative beamforming). As seen in [5], letting M be the number of steering directions $\theta_1, \ldots, \theta_M$, define

$$a(\theta) = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}\langle p_1, \theta \rangle} & \ldots & e^{-j\frac{2\pi}{\lambda}\langle p_L, \theta \rangle} \end{bmatrix}$$

Then we recover steering directions as peaks of the following spectrum estimate:

$$P_M(\theta) = \frac{a^H(\theta)a(\theta)}{a^H(\theta)\Pi^{\perp}a(\theta)}$$

where $\Pi^{\perp}$ is an estimate of the projector operator onto the noise subspace.
The number M is left as a hyperparameter, to be chosen according to the dataset.

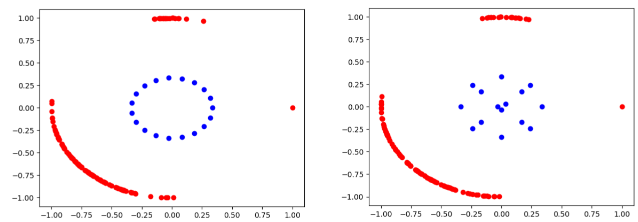

Fig. 1. Antenna and transmitter positions, data 1 (left) and 2 (right) Antennas are denoted as blue dots and user positions as red dots.

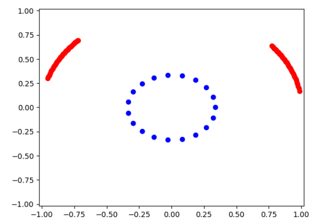

Fig. 2. Antenna and transmitter positions, simulated data Antennas are denoted as blue dots and user positions as red dots.

III. Results

A. Data description

We tested the matched beamforming and the flexibeam techniques on 2 datasets based on real data, and on simulated data. The datasets contain the antenna positions of a single base station, as well as the covariance matrices between antennas through time, the channel wavelength and the directions of the users in the area of this base station.
The simulated data contains Q arbitrary user positions (that can be grouped in 2 clusters), a base station of $L = 19$ antennas forming a circle, with covariance matrix between the antennas generated using the formula:

$$\Sigma = A\mathrm{diag}(f)A^H$$

with $A \in \mathbb{C}^{L \times Q}$ being the steering matrix of the simulated base station, and $f \in \mathbb{R}^Q$ a model for the user density.

B. DOA estimation

We empirically optimally set the number of steering directions M to find through MUSIC, obtaining the plots (3) and (4) after applying MUSIC as described previously.

	M
Dataset 1	13
Dataset 2	7
Simulated data	10

C. Matched beamforming results

The beamshapes estimated inferred from dataset 1, dataset 2 and the simulated dataset at time 0 are respectively figures (5), (6) and (7). We only plotted the beamshape towards the most important direction of arrival, as when using matched beamforming the station has to change its steering direction periodically.

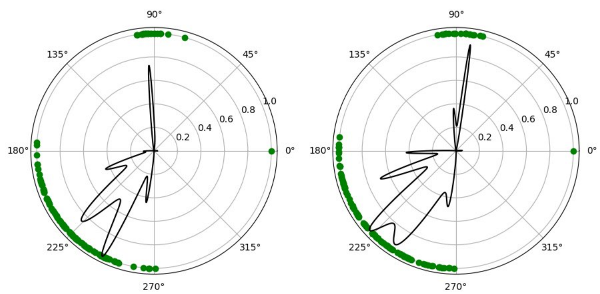

Fig. 3. DOA estimation, dataset 1 (left) and 2 (right) - Green dots denote user positions

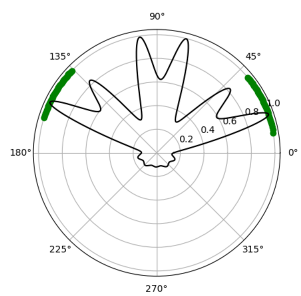

Fig. 4. DOA estimation, simulated data

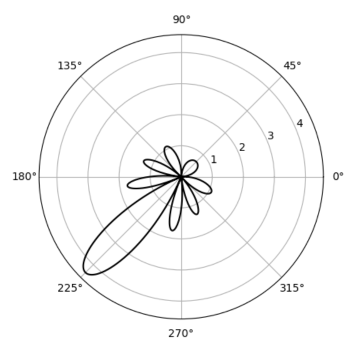

Fig. 5. Beamshape steered at angle 224.25, dataset 1

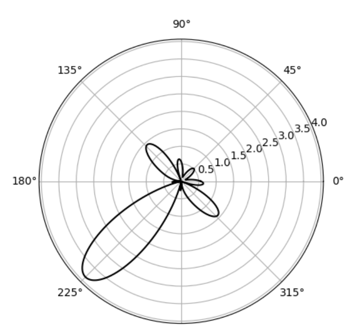

Fig. 6. Beamshape steered at angle 224.25, dataset 2

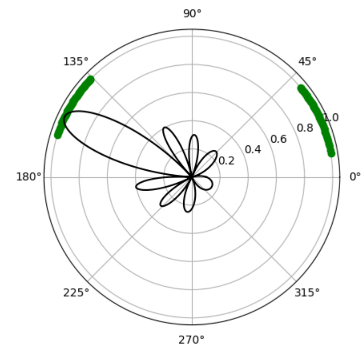

Fig. 7. Beamshape steered at angle 154.65, simulated data

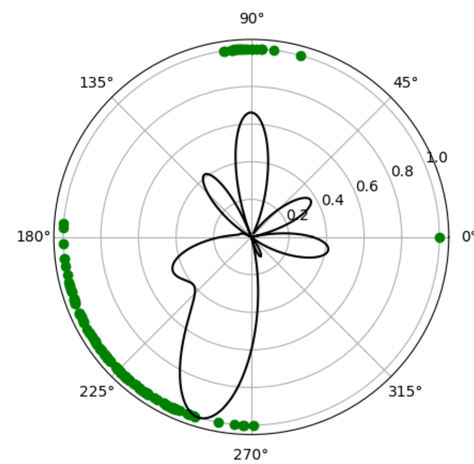

Fig. 8. Beamshapes evaluated by flexibeam, dataset 1

D. Flexibeam results

The beamshapes estimated by flexibeam inferred from dataset 1, dataset 2 and the simulated dataset at time 0 are respectively figures (8), (9) and (10).

E. Throughput evaluation of beamforming strategies

For each beamforming strategy, we computed the average throughput following steering direction estimation with *MUSIC*: using Shannon's channel capacity [6], we estimate the data rate at time t at the position r of a user to be:

$$\text{Bitrate}(r) = B \log_2 \left(1 + C_0 \frac{|b(r)|^2}{\sigma^2} \right)$$

where it is assumed that $s(t) = 1$ w.l.o.g, B is the channel bandwidth, C_0 is a constant that denotes the gap between the channel capacity and the achieved data rate (due to factors like imperfect channel coding), and σ is the white noise variance (all being constants) reported below.

C_0	0.8
B	2×10^6
σ^2	0.1

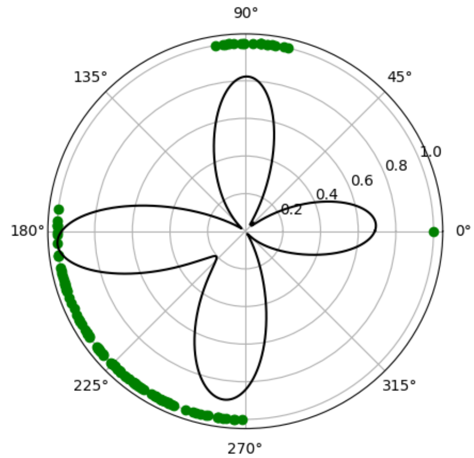

Fig. 9. Beamshapes evaluated by flexibeam, dataset 2

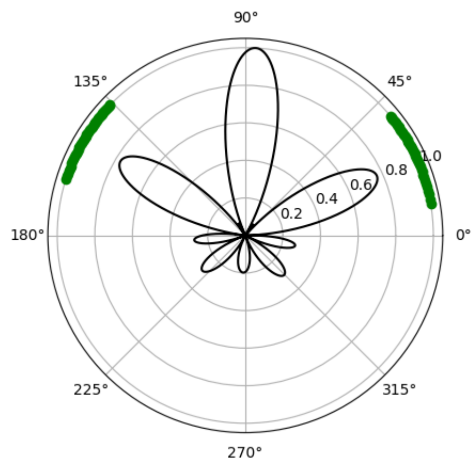

Fig. 10. Beamshapes evaluated by flexibeam, simulated data

Note that these constants were arbitrarily set, thus the average throughput estimates of a beamforming strategy only make sense when compared to throughputs of other beamforming strategies. We present the results on real data:

	Dataset 1	Dataset 2
Matched beamforming	7.27 Mb/s	6.87 Mb/s
Flexibeam	8.77 Mb/s	10.86 Mb/s

We can see that overall, flexibeam always gives a better average throughput than the matched beamforming strategy, and that the antenna dispositions in dataset 2 significantly improves the throughput of flexibeam.

F. Testing collaborative beamforming

To test collaborative beamforming, the setup in figure (11) was created. It is comprised of 3 base stations of 20 antennas disposed along a circle, with 100 users of positions each belonging to one of 6 uniformly weighted mixtures of a Gaussian mixture model. The considered

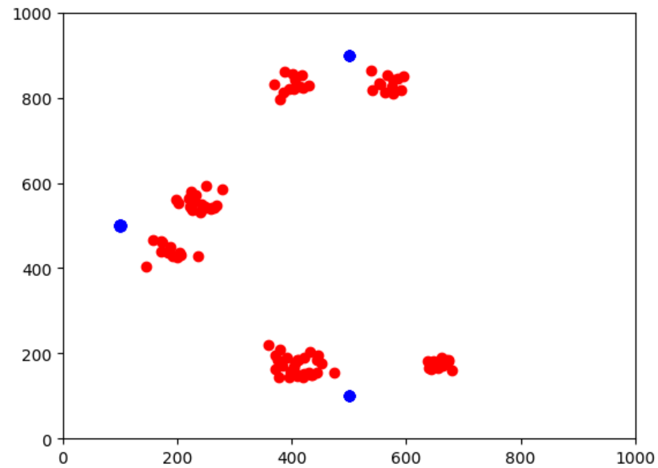

Fig. 11. Setup for collaborative beamforming: stations in blue, users in red

region is of size $N_s \times N_s$ m^2 and it is tessellated with $N_z \times N_z$ m^2 zones. Thus f is parameterized as a grid $\hat{f} \in \mathbb{R}^{\sqrt{N_Q} \times \sqrt{N_Q}}$ where $N_Q = \left(\frac{N_s}{N_z}\right)^2$, with concrete values as follows.

Size of region $N_s \times N_s$	1000 x 1000 m^2
Size of zone $N_z \times N_z$	20 x 20 m^2
$N_Q = \left(\frac{N_s}{N_z}\right)^2$	$\left(\frac{1000}{20}\right)^2 = 2500$

As explained before, collaborative beamforming reduces to an optimization problem. To implement it, we use the `Pycsou` [7] Python library. It combines proximal optimization algorithms as well as a powerful arithmetic unit for representing functions. Here, we construct the target function as:

$$F(\hat{f}, \hat{\sigma}_n^2) = \left|\left| \hat{R} - W^H \left(A\mathrm{diag}\left(\hat{f}\right) A^H - \hat{\sigma}_n^2 I_{SL} \right) W \right|\right|_F^2$$

$$G(\hat{f}, \hat{\sigma}_n^2) = ||\hat{f}||_1$$

where F is differentiable - and in fact quadratic, and G is proximable. It can be shown that:

$$F(\hat{f}, \hat{\sigma}_n^2) = \begin{bmatrix} \hat{f} \\ \hat{\sigma}_n^2 \end{bmatrix}^T Q \begin{bmatrix} \hat{f} \\ \hat{\sigma}_n^2 \end{bmatrix} + b^T \begin{bmatrix} \hat{f} \\ \hat{\sigma}_n^2 \end{bmatrix} + t$$

where:

$$Q = \begin{bmatrix} D \circ D^T & -2\Re\{\mathrm{diag}(D^2)\} \\ 0 & ||C^H C||_F^2 \end{bmatrix} \in \mathbb{R}^{(N_Q+1)\times(N_Q+1)}$$

$$b = -2\Re\left\{ \begin{bmatrix} \mathrm{diag}(C\hat{R}C^H) \\ \mathrm{Tr}(C\hat{R}C^H) \end{bmatrix} \right\} \in \mathbb{R}^{N_Q+1}$$

$$t = ||\hat{R}||_F^2 \in \mathbb{R}$$

with $\circ$ denoting the Hadamard product, and:

$$D = CC^H, \; C = A^H W$$

Hence F is implemented as a `QuadraticFunc`$(2Q, b, t)$ and G as a slightly modified `L1Norm`, applied only on $\hat{f}$,

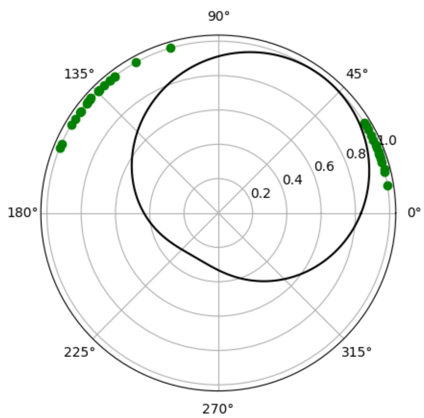

Fig. 12. Collaboratively estimated beamshape of the bottom station

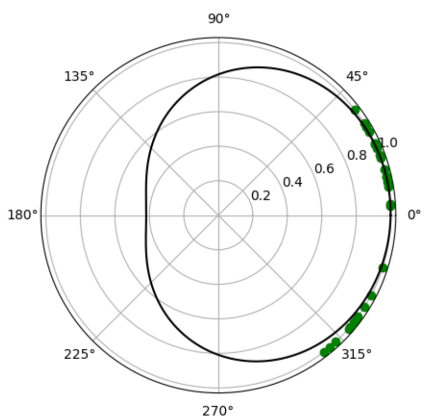

Fig. 13. Collaboratively estimated beamshape of the left station

i.e all coordinates of the input but the last one, $\hat{\sigma}_n^2$.

Then the function $F(\hat{f}, \hat{\sigma}_n^2) + \lambda G(\hat{f}, \hat{\sigma}_n^2)$ is minimized with 3000 steps of Proximal Gradient Descent, taking approximately half a minute on a 16 GB RAM laptop equipped with an AMD Ryzen 7 CPU. Once $\hat{f}$ is computed, the estimated clusters are each assigned to the nearest station, and from there DOAs then flexibeam beamshapes are recomputed. Results for the station at the bottom are in (12), for the station on the left in (13) and for the station at the top in (14).

The results for the left station are encouraging, but as shown by the top and bottom stations' beamshapes convergence is slow. As a sanity check, we simplified the setup down to 3 mixtures, one per station, and still 100 users, as shown in (15). where we get good results with much faster: (16) shows the beamshape of the top station after 300 iterations of PGD. So spread of data greatly affects the efficiency of the method.

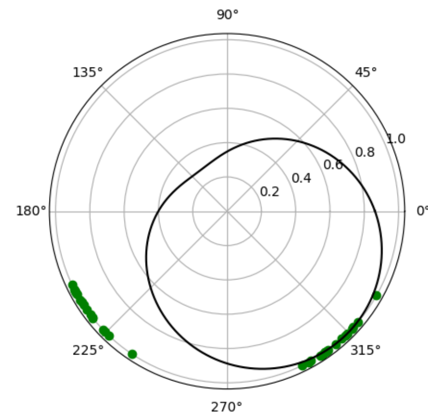

Fig. 14. Collaboratively estimated beamshape of the top station

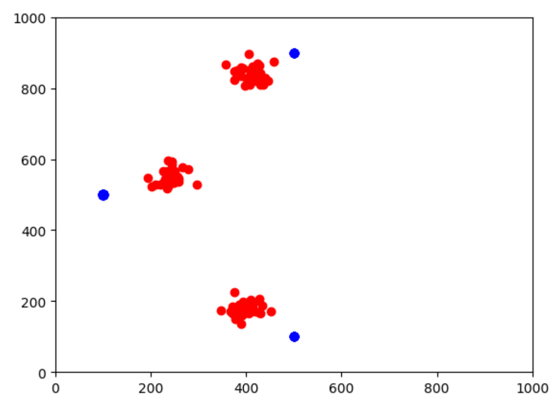

Fig. 15. Alternative setup for collaborative beamforming

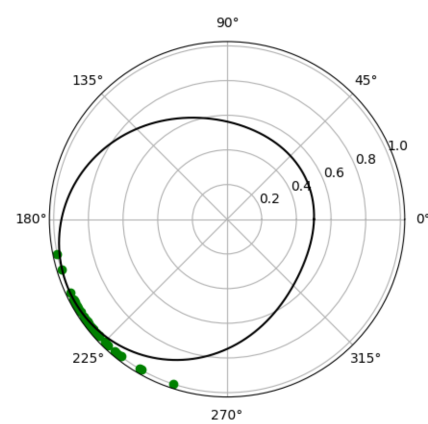

Fig. 16. Collaboratively estimated beamshape of the top station, alt. setup

IV. Conclusions and Discussion

As expected, we observe that matched beamfiltering fails to produce beamshapes with lobes covering entire clusters of receivers. This was theoretically expected as it aimed to produce δ-like beamforming weight vectors, centered around the steering direction of the station.

As supported by theory, flexibeam provides a greater throughput compared to the matched beamfiltering method, since it creates a beamshape with different lobes for different clusters of user, removing the need of matched beamforming to periodically change the steering directions to cover them all one by one.

However, the advanced methods we studied have theoretical limitations.

Regarding flexibeam, although it can theoretically produce multiple main lobes, the continuous field of antennas assumption may prove to be hard to approximate in practice with a finite number of antennas.

The presented strategy for collaborative beamforming with flexibeam is computationally expensive, since it requires solving a penalized LASSO problem. It was also uncertain how well the method works for the problem at hand because it was initially developed and tested in [3] for sparse sky image reconstruction, and as shown by the numerical results, its validity in practice is questionable as it takes very long to converge. On top of that, the sparseness assumption may prove to be unrealistic. That said, the $\mathcal{L}^1$ regularizer may also be swapped for any other that may promote desired properties.

V. Future Works

The datasets we had at disposal only had data for a single base station and we did not have access to real data for multiple stations. Since collaborative beamforming needs to consider multiple base stations, it became impossible to test it on real data. We only tested it on simulated data, which neglects a lot of external factors such as channel fading, handover, etc. Further research could collect real data from a group of base stations and test out collaborative beamforming.

On the topic of handovers, we did not consider what to do when a cluster of people is approximately at the same distance from different base stations. Indeed, this would lead to a considerable amount of people constantly switching between two base stations, which is not optimal. A possible way to solve this could be to assign the entire user cluster to one of the candidate base stations, but this project did not study how to consider these clusters.

Lastly, the LASSO optimization problem may be solved faster with stronger assumptions and/or better-suited algorithms, such as Polyatomic Frank-Wolfe [8].

References

[1] P. Hurley and M. Simeoni, “Beamforming towards regions of interest for multi-site mobile networks,” A. Lapidoth and S. M. Moser, Eds. Zurich: ETH Zurich, 2016, Conference Paper.

[2] ——, “Flexibeam: Analytic spatial filtering by beamforming,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2877–2880.

[3] O. Æçal, P. Hurley, G. Cherubini, and S. Kazemi, “Collaborative randomized beamforming for phased array radio interferometers,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5654–5658.

[4] A.-J. Veen and S. Wijnholds, *Signal Processing Tools for Radio Astronomy*, 05 2013, pp. 421–463.

[5] H. Krim and M. Viberg, “Two decades of array signal processing research: the parametric approach,” *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[6] C. E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, pp. 623–656, 1948.

[7] “Pycsou github repository,” https://github.com/matthieumeo/pycsou.

[8] A. Jarret, J. Fageot, and M. Simeoni, “A fast and scalable polyatomic frank-wolfe algorithm for the LASSO,” *IEEE Signal Processing Letters*, vol. 29, pp. 637–641, 2022. [Online]. Available: https://doi.org/10.1109%2Flsp.2022.3149377