Type-Based Unsourced Multiple Access over Fading Channels with Cell-Free Massive MIMO

2025 IEEE International Symposium on Information Theory (ISIT 2025)

Kaan Okumus[*], Khac-Hoang Ngo[†], Giuseppe Durisi[*], Erik G. Ström[*]

[*] Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden
[†] Department of Electrical Engineering (ISY), Linköping University, Linköping, Sweden

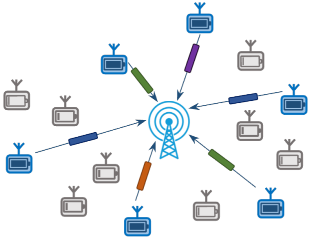
okumus@chalmers.se

June 26, 2025

CHALMERS

Massive uncoordinated access systems

Future 6G and IoT technologies will need to support massive machine-type communication (mMTC)



Requirements for mMTC

- Data collection from low-cost sensor networks over a **shared medium**
- **Uncoordinated** and **sporadic** ($\leq 1/$min) transmissions
- **Massive** number of devices ($\sim 10^7/$Km^2)

Any scheme that assigns dedicated resources to each user is not scalable

Unsourced multiple access (UMA) [[Polyanskiy, 2017]]

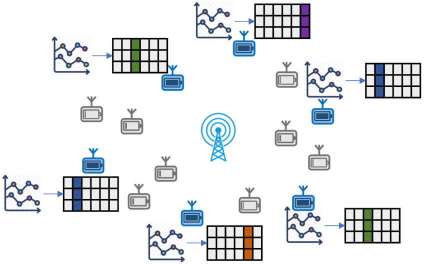
UMA framework

- **Same encoder** for all users
- Receiver produces an **unordered** list of messages $\Longrightarrow$ **“unsourced”**

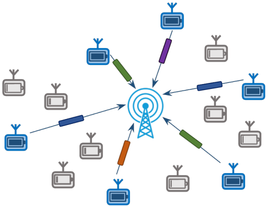
Limitation of UMA

Message collisions treated as errors in UMA

Do message collisions occur in practice?



Sampling and quantization



Data transmission

- Users may transmit the same quantized information
- Applications: multi-target tracking, federated learning
- We are interested in both the set of transmitted messages and their **multiplicities**

Type-based UMA (TUMA)

An extension of UMA that handles message collision

Goal: Estimate type, i.e., empirical distributions of messages

TUMA over Gaussian MAC [Ngo et al., 2024]

- Type estimation formulated as a compressed sensing problem
- Approximate message passing (AMP) algorithm employed for decoding
- Assumes perfect power control
- Multiplicities estimated from the received codeword power

Extension to fading is not straightforward

How to handle fading channels?

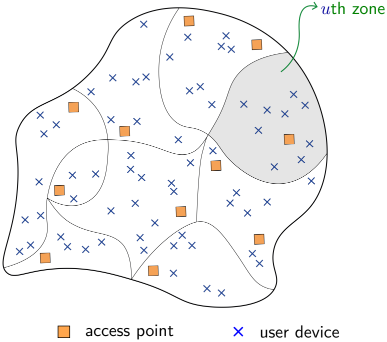
The benefits of cell-free architectures in UMA

[[Cakmak et al., 2025]]: UMA within cell-free massive MIMO

- Rayleigh fading channel
- Perfect knowledge of large-scale fading model
- Location-based codebook partition
- Multisource AMP decoder

Same ingredients can be used for TUMA under fading

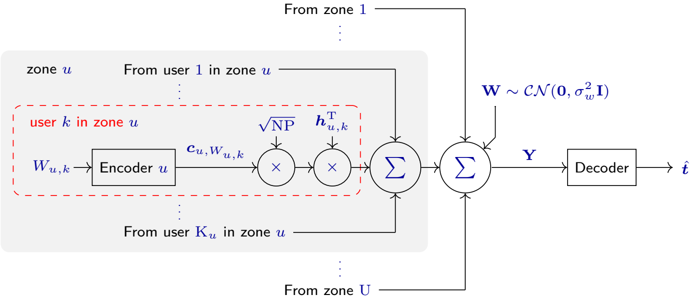
TUMA over a cell-free massive MIMO architecture



TUMA with cell-free setup:

- area partitioned into U zones
- users in a zone have similar large-scale fading coefficients
- codebook partitioned into U zones
- codebook for zone u: $\mathbf{C}_u \in \mathbb{C}^{\mathrm{N}\times\mathrm{M}}$
- N: blocklength
- M: number of possible messages
- B access points (APs) with A antennas each
- $\mathrm{F} = \mathrm{B} \times \mathrm{A}$ antennas in total
- APs connected to a CPU via fronthaul links

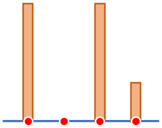
System overview of TUMA over CF Massive MIMO



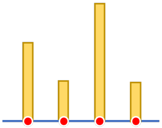
- User k in zone u has message $W_{u,k}$ and transmits codeword $\boldsymbol{c}_{u,W_{u,k}}$, scaled by $\sqrt{\mathrm{NP}}$
- Fading modeled via channel vector $\boldsymbol{h}_{u,k} \in \mathbb{C}^{\mathrm{F}}$ (from user to all AP antennas)
- CPU receives superposition signal $\mathbf{Y}$ and aims to estimate the type vector $\boldsymbol{t}$ from it

Type estimation and performance metric

- Messages $\{W_{u,k}\}$ form a global type vector $\boldsymbol{t} = [t_1, \ldots, t_M]$
- t_m is the fraction of users transmitting message m
- Decoder returns a type estimate $\hat{\boldsymbol{t}}$ from $\mathbf{Y}$, using known codebooks $\{\mathbf{C}_u\}$ and a statistical path loss model (no instantaneous CSI)



transmitted type, $\boldsymbol{t}$



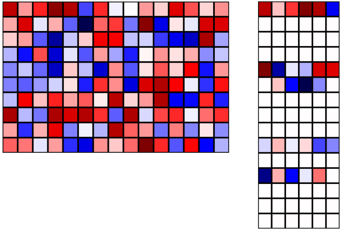
decoded type, $\hat{\boldsymbol{t}}$

Performance metric: total variation distance

$$\mathbb{TV}(\boldsymbol{t}, \hat{\boldsymbol{t}}) = \frac{1}{2} \sum_{m=1}^{\mathrm{M}} |t_m - \hat{t}_m|$$

TUMA as a compressed sensing problem

$\mathbf{C}_u \in \mathbb{C}^{\mathrm{N} \times \mathrm{M}}$ $\mathbf{X}_u \in \mathbb{C}^{\mathrm{M} \times \mathrm{F}}$



- Users who transmit the same message contribute to the same row of $\mathbf{X}_u$
- The mth row is:

$$\boldsymbol{x}_{u,m} = \sum_{k:\, W_{u,k}=m} \boldsymbol{h}_{u,k}$$

- Only a few messages are active $\rightarrow$ $\mathbf{X}_u$ is row-sparse

Global system equation:

$$\mathbf{Y} = \sqrt{\mathrm{NP}} \sum_{u=1}^{\mathrm{U}} \mathbf{C}_u \mathbf{X}_u + \mathbf{W}$$

This is a structured compressed sensing problem (multisource)

Proposed centralized TUMA decoder

We use **multisource AMP** [Cakmak et al., 2025] to estimate $\{\mathbf{X}_u\}$ and enable type estimation

For $t = 1, 2, \ldots,$ iterate:

- Effective observation: $\mathbf{R}_u^{(t)} = \mathbf{C}_u^{\mathsf{H}} \mathbf{Z}^{(t-1)} + \sqrt{\mathrm{NP}} \mathbf{X}_u^{(t-1)}$
- Bayesian denoising: $\mathbf{X}_u^{(t)} = \eta_{u,t}(\mathbf{R_u}^{(t)})$
- Onsager correction: $\mathbf{Q}_u^{(t)}$ via Jacobian of $\eta_{u,t}$
- Residual update: $\mathbf{Z}^{(t)} = \mathbf{Y} - \sqrt{\mathrm{NP}} \sum_{u=1}^{\mathrm{U}} \left(\mathbf{C}_u \mathbf{X}_u^{(t)} - \frac{\mathrm{M}}{\mathrm{N}} \mathbf{Z}^{(t-1)} \mathbf{Q}_u^{(t)} \right)$

Asymptotically, the model $\mathbf{Y} = \sqrt{\mathrm{NP}} \sum_{u=1}^{\mathrm{U}} \mathbf{C}_u \mathbf{X}_u + \mathbf{W}$ decouples into M independent submodels:

$$\boldsymbol{r}_{u,m}^{(t)} = \sqrt{\mathrm{NP}} \boldsymbol{x}_{u,m}^{(t)} + \mathcal{CN}(\mathbf{0}; \mathbf{T}^{(t)}),$$

where $\mathbf{T}^{(t)}$ is a diagonal covariance matrix computed via state evolution

The Bayesian denoiser $\eta_{u,t}$ is designed for this decoupled model to estimate $\boldsymbol{x}_{u,m}^{(t)}$

Proposed centralized TUMA decoder

Our contributions:

Bayesian denoiser $\eta_{u,t}$ is tailored to the TUMA system using:

- Prior on multiplicities $k_{u,m}$: **truncated binomial distribution**
- Prior on user positions: **uniform distribution** over the zone

Challenges due to message collisions:

- High-dimensional integrals in denoiser and Onsager correction
- Solved using **Monte Carlo sampling** for scalability

Type estimation procedure:

- Bayesian estimation of type $\hat{\boldsymbol{t}}$ (using posteriors already computed to evaluate $\eta_{u,t}$)

Proposed distributed TUMA decoder

To increase scalability, we design a **distributed decoder** inspired by **distributed AMP** [Bai et al., 2022]

Key idea: Move likelihood computations from CPU to the access points (APs)

At each AP:

- Process local received signal
- Compute local likelihoods using Monte Carlo sampling

At the CPU:

- Fuse them to form global likelihoods
- Perform Bayesian type estimation (same as centralized)

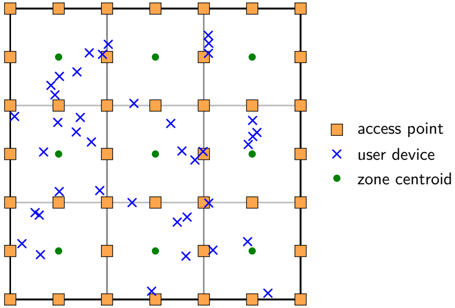
Benefits compared to centralized decoder:

Reduced CPU load, reduced fronthaul bandwidth, scalable to large CF networks

Drawbacks:

Slight performance loss compared to centralized decoder

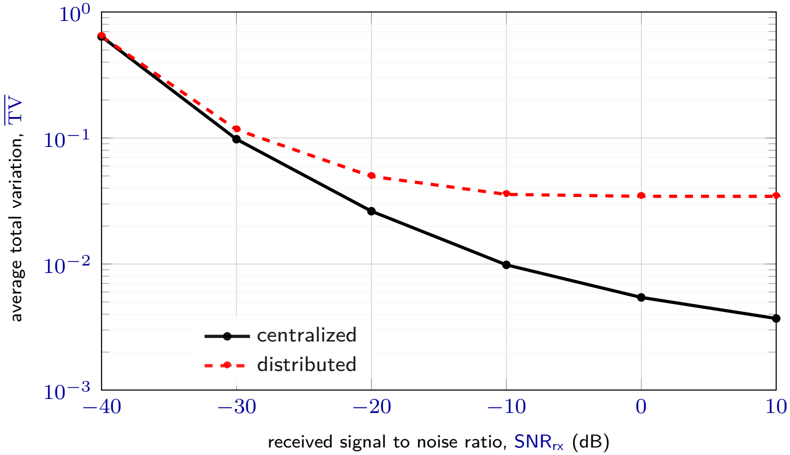
Simulation settings



Example of TUMA with cell-free setup:

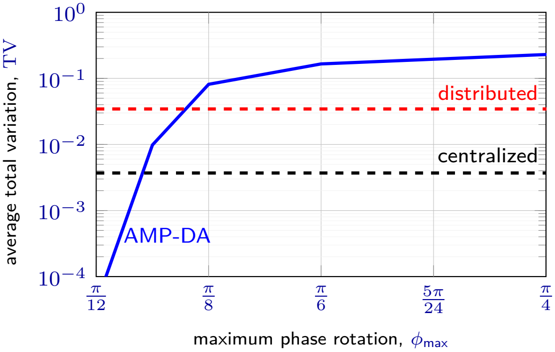
- 300 m $\times$ 300 m coverage area
- $\mathrm{U} = 9$ zones
- $\mathrm{B} = 40$ APs with $\mathrm{A} = 4$ antennas each
- Blocklength $\mathrm{N} = 1024$
- $\mathrm{M} = 2^{12}$ possible messages
- 180 active users (20 users per zone)
- 13 distinct messages per zone
- average multiplicity is 2
- maximum multiplicity is 7

Simulation results–comparison between proposed decoders



Simulation results–comparison with AMP-DA decoder

- AMP-DA [Qiao et al., 2024] is a type-based decoder tailored for federated learning
- AMP-DA requires perfect CSI at the UEs for channel pre-equalization
- In contrast, our TUMA decoders do not require instantaneous CSI
- Benchmark: performance of AMP-DA under imperfect CSI (random phase rotation)



Our decoders outperform AMP-DA even for small phase rotation

Conclusion

- We extended the **TUMA** framework to **fading channels** using a **CF massive MIMO** architecture
- Proposed **centralized** and **distributed decoders** based on multisource AMP
- Our decoders handle **message collisions** and unknown user positions using **Bayesian estimation**
- **Monte Carlo sampling** enables scalable implementation of the denoiser
- Simulation results show improved **type estimation** and **robustness compared to AMP-DA**

Code and examples available at: `github.com/okumuskaan/tuma_fading_cf`