Partially observable model-based learning for ISAC resource allocation

Aalto University School of Electrical Engineering

ICASSP 2024 Korea

Petteri Pulkkinen[1,2] & Visa Koivunen[1]

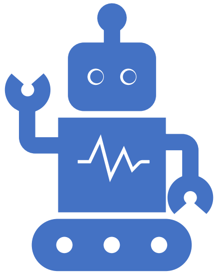
[1] Department of Information and Communications Engineering, Aalto University

[2] Saab Finland Oy

Introduction

- **I**ntegrated **S**ensing **A**nd **C**ommunications (**ISAC**) is a key new technology for 6G wireless systems and beyond
  - Radar sensing and wireless communication are **co-designed for mutual benefit**
- ISAC systems reduce **the system cost** and **power consumption** while also **mitigating congestion** in the radio spectrum
- We consider rapidly **time-frequency-space** varying shared spectrum scenarios
  - RF systems **dynamically** use the spectrum
  - **Mobility** of transmitters and receivers
- **Online** (machine) **learning** facilitates using the spectrum optimally in dynamic shared spectrum scenarios
  - **Actively** learn to allocate resources in time-frequency-space dimensions

Prior work and contributions



In our prior work:

We developed **model-free** and **model-based** online learning algorithms that learn **dynamic power allocation and sub-carrier selection** policies from experience

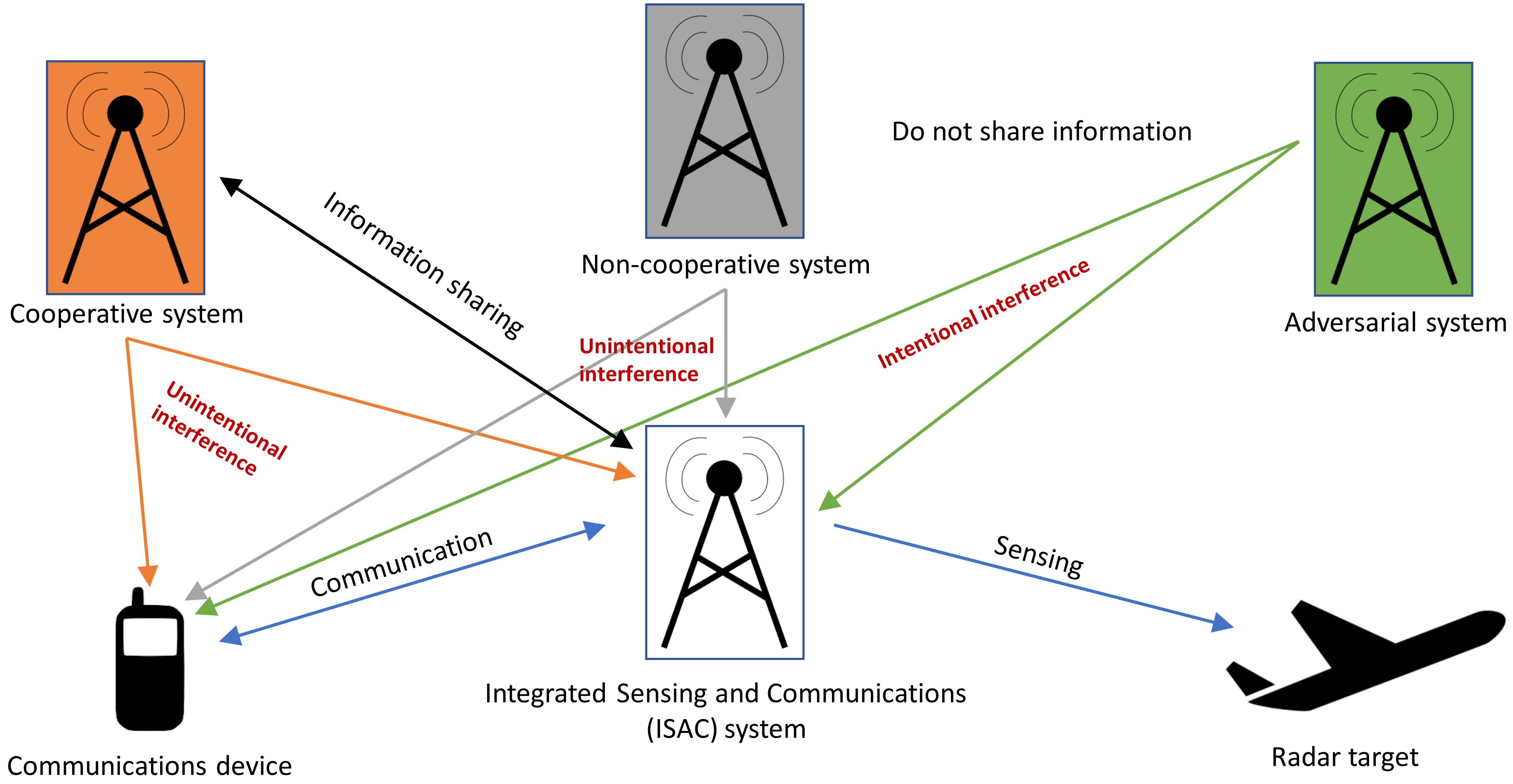
We established **bounds of the convergence rates** for the model-based approach

In this presentation:

The model-based approach **extended to** learning from noisy spectrum observations (**partially observable setting**)

The performance of the proposed methods are **empirically evaluated in dynamic shared spectrum scenarios**

Considered shared spectrum scenario



Signal model

- *"Time slotted"* model used:
  - Channel and interference assumed **W**ide **S**ense **S**tationary (WSS) over single **observation period** *k*
- **Orthogonal multicarrier signal model** with *N* sub-carriers (e.g., OFDM)
- Diagonal frequency-domain channel and interference plus noise covariance matrices
  - Circulant approximation of Toeplitz structure in time-domain
- Signal model in **frequency domain** at receiver $f \in \{r, c\}$:

Sensing code, Comms. symbols, Channel

$$\mathbf{r}_k^f = (\mathbf{b}_k^r + \mathbf{b}_k^c) \odot \mathbf{h}_k^f + \mathbf{v}_k^f$$

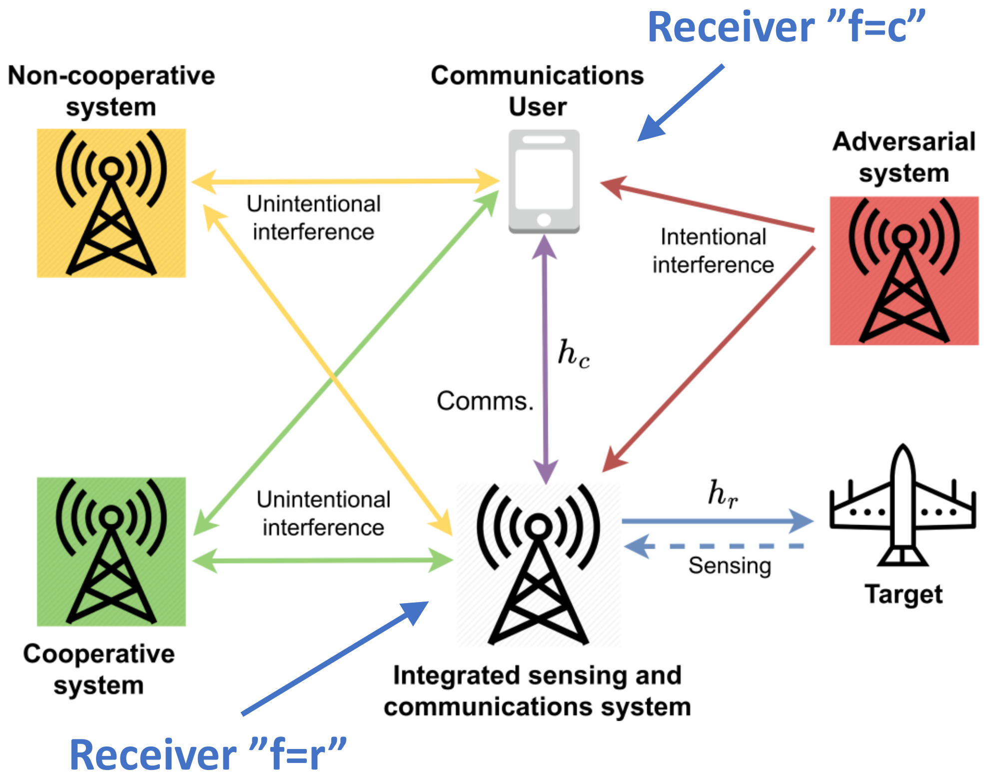
Noise + interference

$$\mathbf{v}_k^x \sim \mathcal{CN}(\mathbf{0}, \mathbf{N}_k^x)$$

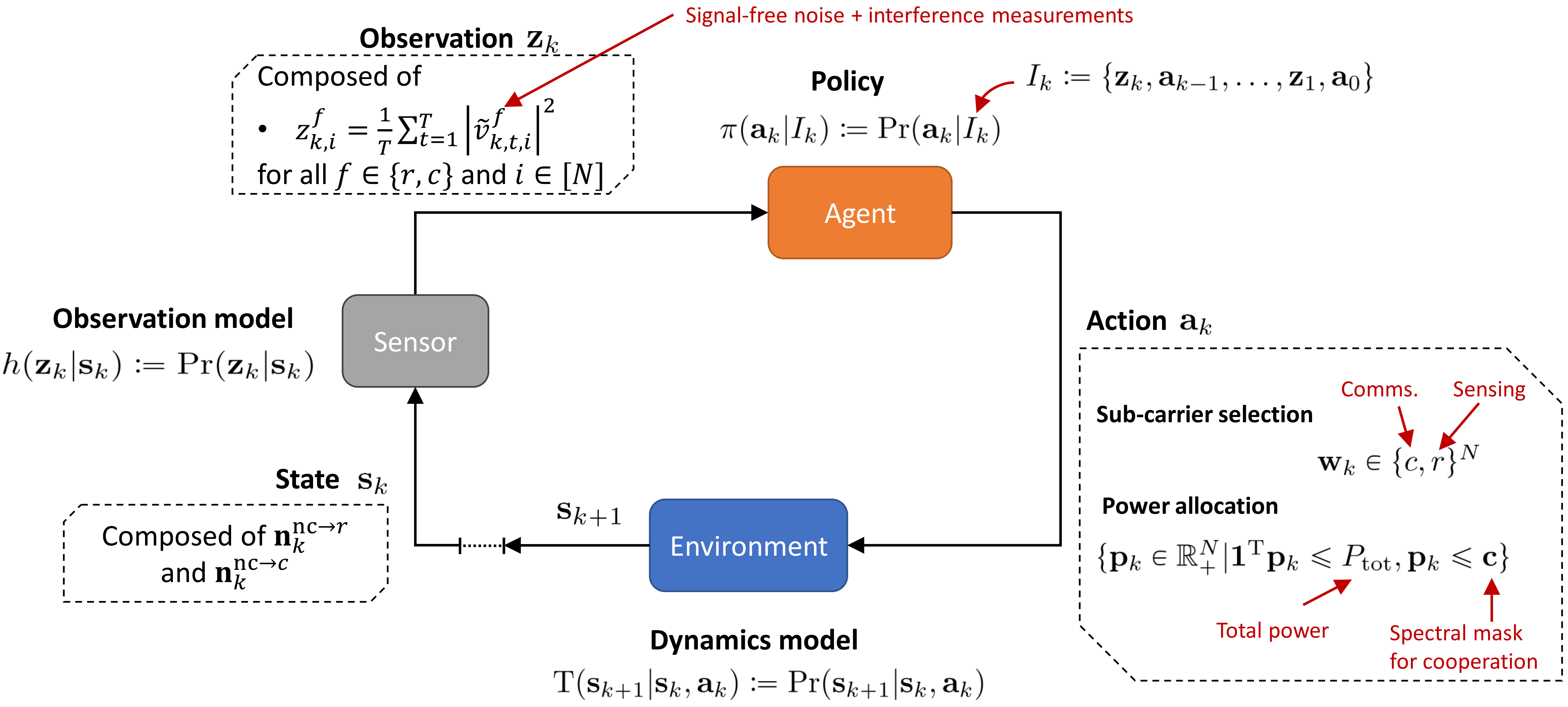
Non-coop. and adversarial systems

$$\mathbf{n}_k^f := \mathrm{diag}\left(\mathbf{N}_k^f\right) = \mathbf{n}_k^{\mathrm{co}\to f} + \mathbf{n}_k^{\mathrm{nc}\to f} + \sigma^2 = \begin{pmatrix} n_{k,1}^f \\ \vdots \\ n_{k,N}^f \end{pmatrix}$$

Coop. system, Noise



Considered POMDP model



Optimization objective

Utility function:

- **Sensing**: Mutual information (MI) between received signal and target scattering matrix
- **Comms.:** Communications MI (i.e., rate)

SINR

$$u(\mathbf{s}_{k+1}^{f}, \mathbf{a}_k) = \sum_{i=1}^{N} \mathbb{1}_{\{w_{k,i}=f\}} \log\left(1 + \frac{g_{k+1,i}^{f}}{n_{k+1,i}^{f}} p_{k,i}\right)$$

Sub-carrier selection

$$\arg\max_{\pi} \; J_r(\pi)$$
$$\text{subject to } J_c(\pi) \geqslant C$$

Avg. sensing utility

Avg. comms. utility

Comms. rate requirement

Partially observable model-based learning (POMBL)

- The POMBL memory is recursive:

  $$\mathbf{x}_k = g_\theta(\mathbf{z}_k, \mathbf{a}_{k-1}, \mathbf{x}_{k-1})$$

  - For example: windowing or recursive neural networks

- POMBL agent learns a transition model:

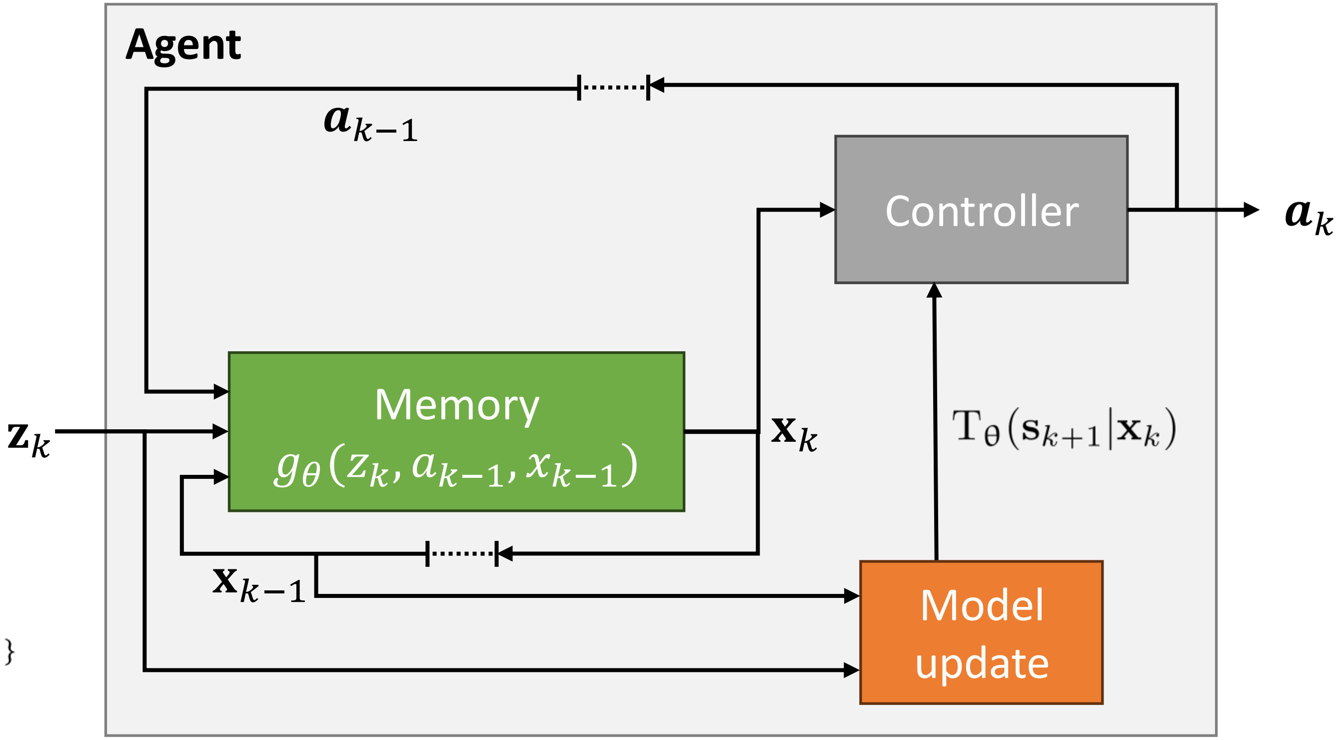
  $$\mathrm{T}_\theta(\mathbf{s}_{k+1}|\mathbf{x}_k) = \Pr\left(S_{k+1} = \mathbf{s}_{k+1}|X_k = \mathbf{x}_k\right)$$

- Suppose pre-determined set of interference levels: $s_{k,i}^{f} := n_{k,i}^{\mathrm{nc}\to f} \in \{\sigma_j^2\}_{j=1}^{L}$

- The model for $f \in \{r, c\}$ can be written as:

$$\mathrm{T}_\theta(\mathbf{s}_{k+1}^f|\mathbf{x}_k^f) = \prod_{i=1}^{N}\prod_{j=1}^{L}\left[\phi_{i,j}^f(\mathbf{x}_k^f, \theta)\right]^{\mathbb{1}_{\{s_{k+1,i}^f = \sigma_j^2\}}}$$

learned probability masses



Model learning problem

- The model is learnt online by “sampling” from the real environment
- Indicate quantized states as: $Y_{k,i,j} = \mathbb{1}_{\{s_{k,i}=\sigma_j^2\}}$
- The POMBL minimizes cross-entropy minimization loss (i.e., the KL divergence):

$$L(\theta) = -\mathop{\mathbb{E}}_{\mathbf{s}_k, \mathbf{x}_{k-1}} \left[\sum_{i=1}^{N} \sum_{j=1}^{L} Y_{k,i,j} \log \phi_{i,j}(\mathbf{x}_{k-1}, \theta) \right]$$

Not observable to the agent → this objective cannot be directly minimized

- The unobservability problem is like one faced in “classification with noisy labels” problem in the context of machine learning
  - Addressed with methods based on loss-correction[1]
  - Find loss function that effectively optimizes $L(\boldsymbol{\theta})$

[1]G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L. Qu, “Making deep neural networks robust to label noise: A loss correction approach,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2233–2241, 2017.

Backward loss correction method

- Use detector (e.g., maximum likelihood) to estimate $\mathbf{Y}_k$:
  - Detector $D_i(z_{k,i}) \in \{0,1\}^L$ outputs one-hot encoded vector indicated the detected state
- We can compute the confusion matrix using model h:

$$\mathbf{U}_i = \left[\mathbb{E}\left[D_i(z_{k,i})|s_{k,i} = \sigma_1^2\right], \ldots, \mathbb{E}\left[D_i(z_{k,i})|s_{k,i} = \sigma_L^2\right]\right]$$

- Unbiased gradient estimator can be written as:

$$\widehat{\nabla}_\theta L(\theta) = -\sum_{i=1}^{N}\sum_{j=1}^{L}[\mathbf{U}_i^{-1} D_i(\mathbf{z}_k)]_j \nabla_\theta \log \phi_{i,j}(\mathbf{x}_{k-1}, \theta)$$

  - where $\mathbb{E}\left[\mathbf{U}_i^{-1} D_i(z_{k,i})|s_{k,i} = \sigma_j^2\right] = \mathbf{e}_j \;\forall\; j \in [L], i \in [N]$
- The condition number may be large for difficult observation models, inevitably increasing the gradient estimator variance

[1]G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L. Qu, “Making deep neural networks robust to label noise: A loss correction approach,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2233–2241, 2017.

Forward loss correction method

- Consider a surrogate objective (effectively the same[1]):

$$\widehat{L}(\theta) = \mathop{\mathbb{E}}_{\mathbf{z}_k, \mathbf{x}_{k-1}} [\log \Pr(\mathbf{z}_k | \mathbf{x}_{k-1}, \theta)]$$

- By definition, can be written as:

$$\widehat{L}(\theta) = \mathop{\mathbb{E}}_{\mathbf{z}_k, \mathbf{x}_{k-1}} \left[\sum_{i=1}^{N} \log \sum_{j=1}^{L} h(z_{k,i} | s_{k,i} = \sigma_j^2) \phi_{i,j}(\mathbf{x}_{k-1}, \theta) \right]$$

Known probability density for the observations

[1]G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L. Qu, “Making deep neural networks robust to label noise: A loss correction approach,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2233–2241, 2017.

The controller

- Expected utility given $\mathbf{x}_k^f$: $J_\theta(\mathbf{x}_k^f, \mathbf{a}) = \sum_{i=1}^{N} \sum_{j=1}^{L} \mathbb{1}_{\{w_{k,i}=f\}} \phi_{i,j}(\mathbf{x}_k^f, \theta) \log\left(1 + \frac{g_{k+1,i}^f}{n_{k+1,i}^{\mathrm{co}\to f} + \sigma_j^2 + \sigma^2} p_{k,i}\right)$

- Objective:

$$\arg\max_{\mathbf{a}\in\mathcal{A}} \; J_\theta(\mathbf{x}_k^r, \mathbf{a}) \tag{1a}$$

$$\text{s.t. } J_\theta(\mathbf{x}_k^c, \mathbf{a}) \geqslant C \tag{1b}$$

- The controller problem is non-convex → the following procedure is used

Comms. allocation

Set $w_i = c \; \forall \; i$

Optimize power by maximizing comms. rate (convex problem)

Find minimum number of carriers that satisfy the comms. constraint

Remove excess power to meet the constraint strictly

→

Sensing allocation

Allocate remaining carriers and power budget to sensing function

Optimize power by maximizing sensing MI (convex problem)

Numerical examples

- Simulate multiple dynamic interference sources within the considered bandwidth
- Each source modelled using two Markov chains
  - One determines the probabilities for changing sub-channels
  - One determines the interference power between subsequent time slots
- We compare different memory and model architectures when the learning labels Y_k are fully observable
- The loss correction methods are compared where all use the same (best) architecture

Table 1: Simulation parameters.

Parameter	**Symbol**	**Value**
# of subcarriers	N	16
Capacity req.	C	3
Power	P_{tot}	3
Ch. gain	$g_i \ \forall\, i \in [N]$	1e-2
Noise power	σ^2	1e-3
# of interf. levels	L	6 (easy), 16 (difficult)
# of sig.-free meas.	T	1
# of interf. sources		12

Architecture comparison

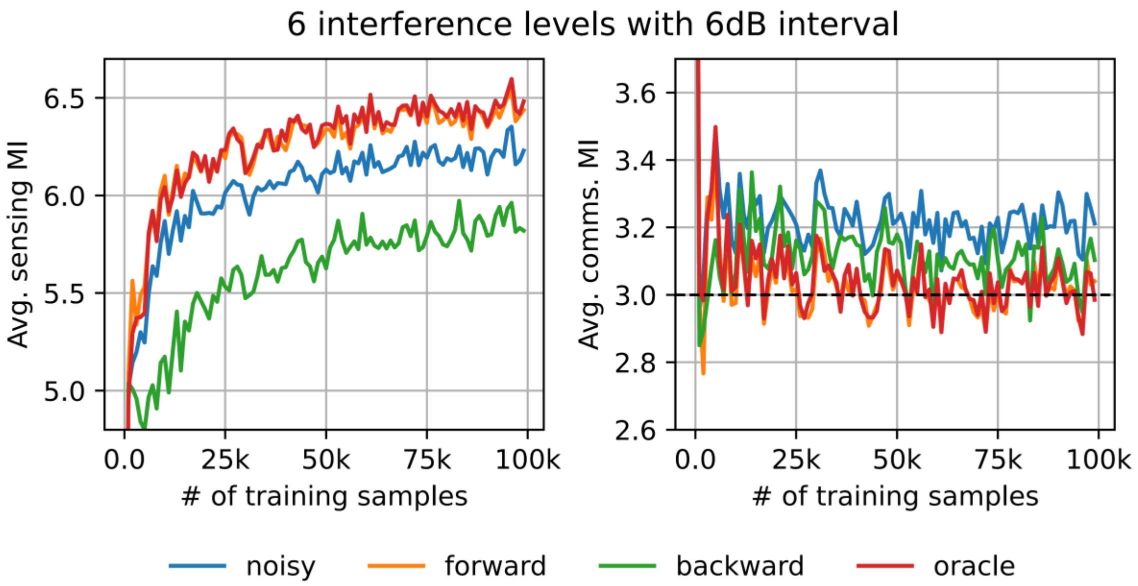
Name	Memory	Model
Linear	No memory	Generalized linear
Linear W.	Windowed	Generalized linear
Non-linear	No memory	Fully connected neural network (FNN) w. Softmax output layer
Non-linear W.	Windowed	— 〃 —
LSTM	LSTM	— 〃 —

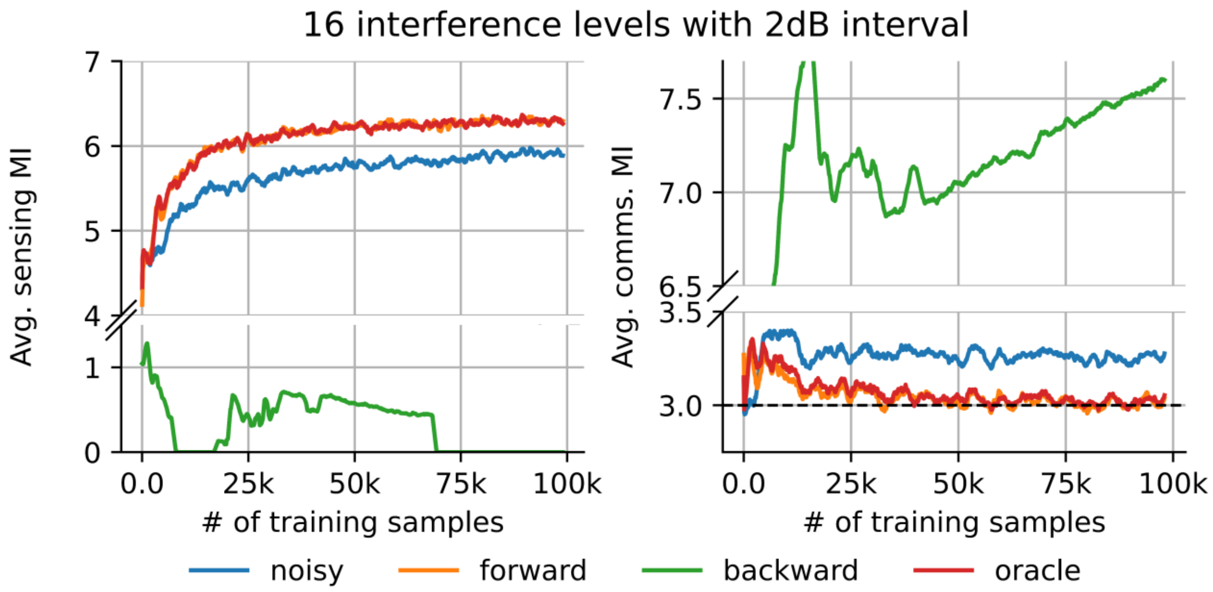
Table 2: Performance with different architectures.

Architecture	MI (sensing)	MI (comms.)	Model loss
LSTM	6.71	3.0	2.93
Non-linear W.	6.55	2.96	2.98
Linear W.	6.63	2.9	3.02
Linear	6.07	3.04	3.06
Non-linear	6.08	2.99	3.07

LSTM with FNN architecture performs the best in the simulation

Loss correction comparison





- All methods improve performance while meeting communications constraints
- The performance of backward method worse than the noisy method
  - Slow converge due to increased variance
- The forward method converges nearly as fast as the oracle method

- Backward method fails miserably due to very badly conditioned confusion matrix
- The forward method converges approximately as fast as the oracle method

Conclusions

- We considered active resource allocation for ISAC systems operating in dynamic shared spectrum environments
- Previous Model-Based Online Learning (MBOL) methods are extended to accommodate partial observability
- A proposed LSTM architecture and forward correction method significantly boost performance and meet communication rate constraints under partial observability
- Simulations indicate that the forward loss correction achieves performance almost identical to the oracle method