Learning to Optimize in Smart Radio Environments

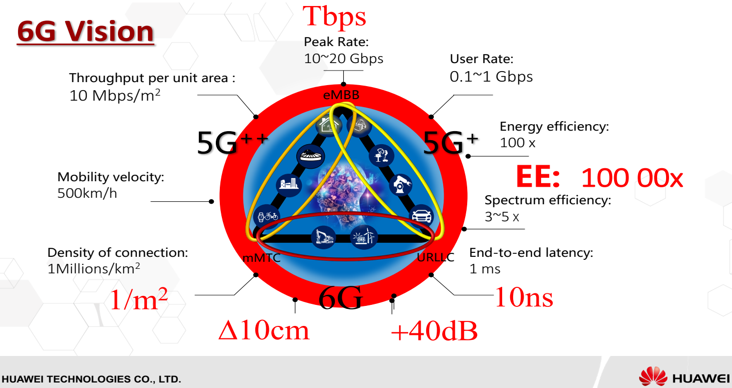
A. Zappone

LANEAS, CentraleSupelec, Paris, France

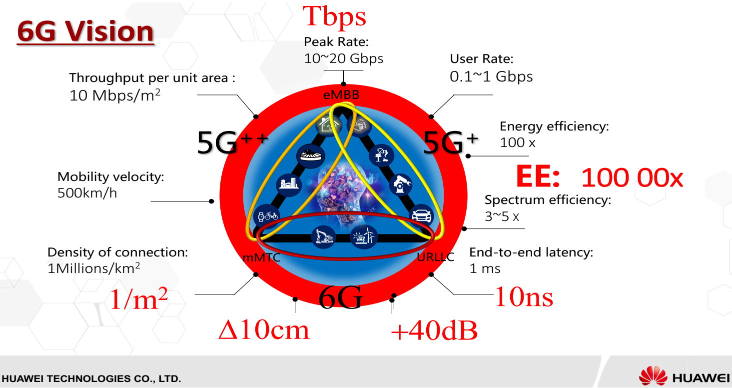
For more info

- A. Zappone, M. Di Renzo, and M. Debbah, "Wireless Networks Design in the Era of Deep Learning: Model-Based, AI-Based, or Both?", IEEE Transactions on Communincations, submitted, 2019 (invited paper), arXiv preprint arXiv:1902.02647
- M. Di Renzo, M. Debbah, D.-T. Phan-Huy, A. Zappone, M.-S. Alouini, C. Yuen, V. Sciancalepore, G. C. Alexandropoulos, J. Hoydis, H. Gacanin, J. de Rosny, A. Bounceu, G. Lerosey, M. Fink. "Smart Radio Environments Empowered by AI Reconfigurable Meta-Surfaces: An Idea Whose Time Has Come", Eurasip Journal on Wireless Communications and Networking, submitted 2019, arXiv preprint arXiv:1903.08925

Why Deep Learning in Wireless Communications?



Why Deep Learning in Wireless Communications?



Because of complexity and modeling issues!

Deep learning in wireless

Enablers:

- **Deep learning requires a lot of data to process**
- **A lot of data is available over the air (Big Data Era)**
- **Improved computing abilities (GPUs)**

Enablers:

- Deep learning requires a lot of data to process
- A lot of data is available over the air (Big Data Era)
- Improved computing abilities (GPUs)

Challenges:

- How to acquire, store, and process this much data?
- Artificial neural networks in the cloud or in each device?
- How to integrate artificial neural networks into wireless networks?

Smart radio environments and AI

Smart Radio Environments ⇒ AI

- **Meta-surfaces provide data storage and processing abilities**
- **Smart radio environments allow handling big data**

Smart radio environments and AI

Smart Radio Environments $\Rightarrow$ AI

- **Meta-surfaces provide data storage and processing abilities**
- **Smart radio environments allow handling big data**

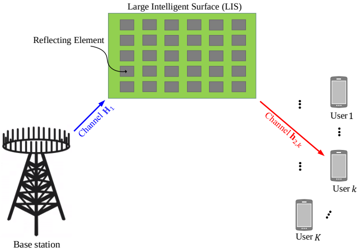
Smart Radio Environments $\Leftarrow$ AI

- **A lot of degrees of freedom to optimize**
- **AI provides a framework for low-complexity system design**

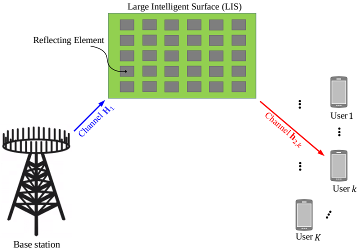
MISO downlink with intelligent surfaces

C. Huang, A. Zappone, G. Alexandropoulos, M. Debbah, C. Yuen, "Large Intelligent Surface for Energy Efficiency in Wireless Communication", IEEE Transactions on Wireless Communications, submitted (minor revision), 2019, https://arxiv.org/abs/1810.06934

Problem statement: EE Maximization



Problem statement: EE Maximization



Upon transmit zero-forcing

$$\max_{\mathbf{\Phi},\mathbf{P}} \frac{\sum_{k=1}^{K} \log_2\left(1+p_k\sigma^{-2}\right)}{\xi\sum_{k=1}^{K} p_k + P_{\mathrm{BS}} + KP_{\mathrm{UE}} + NP_n(b)}$$

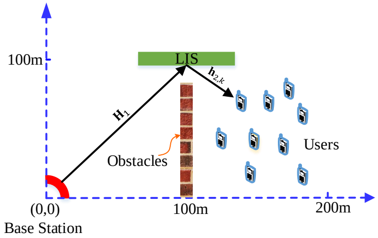
$$\text{s.t. } \log_2\left(1+p_k\sigma^{-2}\right) \geq R_{\min,k}\ \forall k=1,2,\ldots,K,$$

$$\mathrm{tr}((\mathbf{H}_2\mathbf{\Phi}\mathbf{H}_1)^{+}\mathbf{P}(\mathbf{H}_2\mathbf{\Phi}\mathbf{H}_1)^{+H}) \leq P_{\max},$$

$$|\phi_n| = 1\ \forall n=1,2,\ldots,N,$$

Performance analysis

- **Alternating maximization of P and Φ**
- **Optimization of P performed in closed-form**
- **Two iterative methods to optimize Φ**

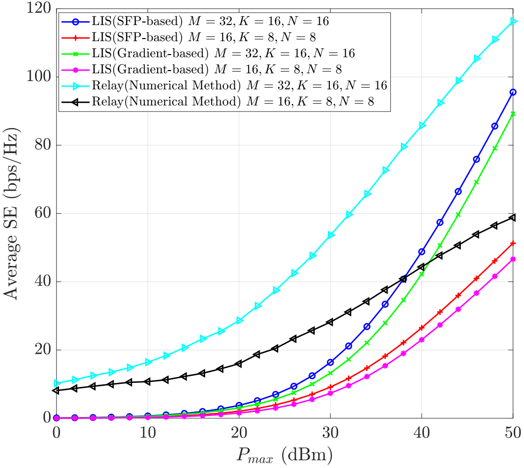


SIMULATION AND ALGORITHMIC PARAMETERS

Parameters	Values
RIS central element placement:	$(100m, 100m)$
BS central element placement:	$(0, 0)$
Small scale fading model $\forall i, k, j$:	$[\mathbf{H}_1]_{ij}, [\mathbf{h}_{2,k}]_i \sim \mathcal{CN}(0, 1)$
Large scale fading model at distance d:	$\frac{10^{-3.53}}{d^{3.76}}$
Transmission bandwidth BW:	$180k$Hz
Circuit dissipated power at BS P_{BS}:	9dBW
Circuit dissipated power coefficients at BS ξ and AF relay ξ_{AF}:	1.2
Maximum transmit power at BS and AF relay $P_{\max}$=$P_{\mathrm{R,max}}$:	20dBW
Dissipated power at each user P_{UE}:	10dBm
Dissipated power at the n-th RIS element $P_n(b)$:	10dBm
Dissipated power at each AF relay transmit-receive antenna P_R:	10dBm
Algorithmic convergence parameter:	$\epsilon = 10^{-3}$

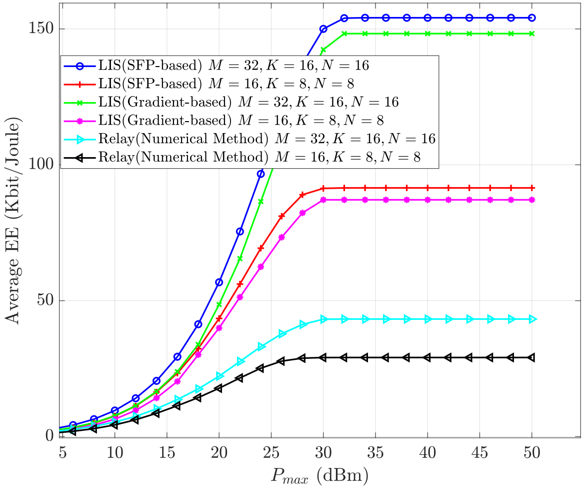
Comparison with AF relaying: Spectral Efficiency

M BS antennas, K mobile users, N reflecting elements



Comparison with AF relaying: Energy Efficiency

M BS antennas, K mobile users, N reflecting elements



- **Relatively small system with ZF, but iterative algorithms needed**
- **If multi-user interference is considered, problem is much harder**
- **Optimal online resource allocation not feasible for larger systems**

Complexity crunch in communication networks

- **Relatively small system with ZF, but iterative algorithms needed**
- **If multi-user interference is considered, problem is much harder**
- **Optimal online resource allocation not feasible for larger systems**

C.1: An accurate and tractable theoretical model is available (e.g., point-to-point channel capacity, point-to-point bit error probability).
C.2: An accurate but intractable theoretical model is available (e.g., achievable sum-rate in interference-limited systems).
C.3: A tractable but inaccurate theoretical model is available (e.g., spectral / energy efficiency of ultra-dense networks, energy consumption models, hardware impairments).
C.4: Only inaccurate and intractable theoretical models are available (e.g., molecular communication networks, optical systems, end-to-end networks optimization).

AI provides the tools to address C.2 and C.3

Learning by ANN

The distinctive trait of deep learning is to implement the learning process by artificial neural networks (ANN).

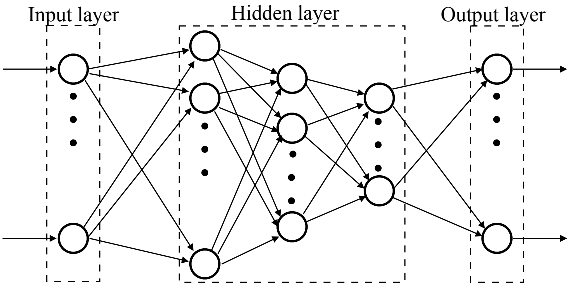
- ANNs are universal function approximators (under very mild assumptions.)
- ANNs exploit large datasets better than other machine learning techniques.

Artificial Neural Networks

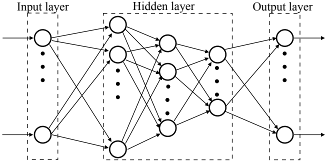
ANN model

ANNs are organized hierarchically in layers of elementary processing units, called *neurons*.

- An input layer forwards the input data to the rest of the network.
- One or more hidden layers process the input data.
- An output layer applies a final processing to the data before outputting it.
- Weights and biases model the strength of the connections among neurons.



Fully-connected networks



The input of Layer ℓ is a vector $\boldsymbol{x}_{\ell-1}$. The output $\boldsymbol{x}_\ell(n)$ of neuron i is:

$$\begin{aligned} \boldsymbol{x}_\ell(i) &= f_{n,\ell}(z_{i,\ell}) \\ z_{i,\ell} &= \boldsymbol{w}_{i,\ell}^T \boldsymbol{x}_{\ell-1} + b_{i,\ell} \ . \end{aligned}$$

- $\boldsymbol{w}_{i,\ell}$ is a vector weighting the inputs from the previous layer.
- $b_{i,\ell}$ is a bias term.
- $f_{i,\ell}(z_{i,\ell})$ is called activation function of the neuron (elementary function).

Each neuron simply takes an affine combination of the inputs, computes the value of the activation function, and propagates the result.

Training a neural network

ANNs are trained in a supervised fashion.

Training problem

Given the training set $\mathcal{S}_{TR} = \{(x_n, y_n^*)\}_{n=1}^{N_{TR}}$, optimize all weights and bias of the ANN, i.e. $\boldsymbol{\theta} = \{\boldsymbol{W}_{tot}, \boldsymbol{b}_{tot}\}$, in order to minimize the training error.

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N_{TR}} \mathcal{L}(\boldsymbol{y}_n(\boldsymbol{\theta}), \boldsymbol{y}_n^*)$$

Training a neural network

ANNs are trained in a supervised fashion.

Training problem

Given the training set $\mathcal{S}_{TR} = \{(x_n, y_n^*)\}_{n=1}^{N_{TR}}$, optimize all weights and bias of the ANN, i.e. $\boldsymbol{\theta} = \{\boldsymbol{W}_{tot}, \boldsymbol{b}_{tot}\}$, in order to minimize the training error.

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N_{TR}} \mathcal{L}(\boldsymbol{y}_n(\boldsymbol{\theta}), \boldsymbol{y}_n^*)$$

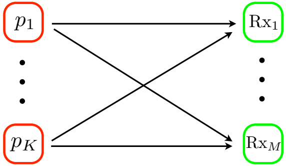
What about hyperparameters? (e.g. number of layers, neurons, etc.)
Trial and error procedure (with some insight)

EE Maximization in Interference Networks

B. Matthiesen, A. Zappone, E. A. Jorswieck, M. Debbah, "Deep Learning for Optimal Energy-Efficient Power Control in Wireless Interference Networks", submitted, 2018, https://arxiv.org/abs/1812.06920

General Interference-limited network model

Interference network with K transmitters and M receivers.



- The receive filter used by receiver m to decode the data from user k is $\boldsymbol{c}_{m,k}$.
- All receivers have N receive antennas and thermal noise power σ^2.
- Each transmitter k has a single antenna and transmits with power p_k.
- The channel between transmitter k and receiver m is $\boldsymbol{h}_{k,m}$.

The SINR for transmitter k at receiver m is:

$$\gamma_{k,m} = \frac{|\boldsymbol{c}_{m,k}^H \boldsymbol{h}_{k,m}|^2 p_k}{\sigma^2 + \sum_{j \neq k} p_j |\boldsymbol{c}_{m,k}^H \boldsymbol{h}_{j,m}|^2} = \frac{a_k p_k}{\sigma^2 + \sum_{j \neq k} p_j b_{k,j}}$$

Sum-EE maximization

Sum-EE maximization is the toughest EE maximization problem:

$$\text{Sum-EE}(\boldsymbol{p}) = \sum_{k=1}^{K} \frac{\log_2(1+\gamma_k)}{P_c + p_k}$$

- $p_k \in [0, P_{max,k}]$ for all k.
- B is the transmission bandwidth.
- P_c is the total hardware power dissipated in all network nodes.

Sum-EE maximization

Sum-EE maximization is the toughest EE maximization problem:

$$\text{Sum-EE}(\boldsymbol{p}) = \sum_{k=1}^{K} \frac{\log_2(1+\gamma_k)}{P_c + p_k}$$

- $p_k \in [0, P_{max,k}]$ for all k.
- B is the transmission bandwidth.
- P_c is the total hardware power dissipated in all network nodes.

Alternative problem formulation

We can write the problem as the computation of the function:

$$F(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{P}_{max}) = \arg\max_{\boldsymbol{p} \in \mathcal{S}} \text{Sum-EE}(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{P}_{max})$$

with $\boldsymbol{a} = \{a_k\}_k$, $\boldsymbol{b} = \{b_{j,k}\}_{j,k}$, $\boldsymbol{P}_{max} = \{P_{max,k}\}_k$.

Sum-EE maximization

Sum-EE maximization is the toughest EE maximization problem:

$$\text{Sum-EE}(\boldsymbol{p}) = \sum_{k=1}^{K} \frac{\log_2(1+\gamma_k)}{P_c + p_k}$$

- $p_k \in [0, P_{max,k}]$ for all k.
- B is the transmission bandwidth.
- P_c is the total hardware power dissipated in all network nodes.

Alternative problem formulation

We can write the problem as the computation of the function:

$$F(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{P}_{max}) = \arg\max_{\boldsymbol{p} \in \mathcal{S}} \text{Sum-EE}(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{P}_{max})$$

with $\boldsymbol{a} = \{a_k\}_k$, $\boldsymbol{b} = \{b_{j,k}\}_{j,k}$, $\boldsymbol{P}_{max} = \{P_{max,k}\}_k$.

Feedforward neural networks are universal function approximators.
An FFN with input $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{P}_{max})$ and output $\boldsymbol{p}$ can learn F.

Implementation and complexity

Algorithm

- Generate a training set $\{(\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{P}_{max}), \boldsymbol{p}_i\}_i$ by maximizing the Sum-EE.
- Train the FNN to adjust parameters and hyperparameters.
- Use the trained network to obtain the optimal power allocation for new (not in the training set) channels.

Implementation and complexity

Algorithm

- Generate a training set $\{(\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{P}_{max}), \boldsymbol{p}_i\}_i$ by maximizing the Sum-EE.
- Train the FNN to adjust parameters and hyperparameters.
- Use the trained network to obtain the optimal power allocation for new (not in the training set) channels.

Complexity

- The trained FNN provides a formula for the optimal power allocation.
- Training the FNN can be done *offline* and *sporadically*.
- Complexity of training reduced by the optimization approach in [1].

References

[1] B. Matthiesen, A. Zappone, E. A. Jorswieck, and M. Debbah, "Deep learning for optimal energy-efficient power control in wireless interference networks," *http://export.arxiv.org/pdf/1812.06920*, 2018

Neural network architecture

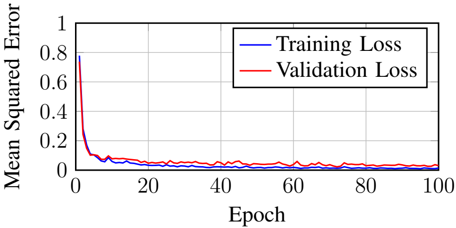
Layer Type	Size	Activation function
Input	20	–
Layer 1 (fully-connected)	128	elu
Layer 2 (fully-connected)	64	relu
Layer 3 (fully-connected)	32	elu
Layer 4 (fully-connected)	16	relu
Layer 5 (fully-connected)	8	elu
Layer 6 (fully-connected)	4	linear

- Supervised learning (Keras)
- ADAM with MSE
- Training set: $N_{TR} = 102,000$. Validation set of $N_V = 10,200$.
- Testing set: $N_{Test} = 10,000$ for each considered value of $P_{max} = -30, \ldots, 20$ dB, with 1 dB step.
- The total number of generated data samples is 622,000.

- Interference channel with $K = 4$ links in a square area with edge of $2km$.
- Single-antenna transmitters; receivers with $N = 2$ antennas each.
- Rayleigh fast-fading, path-loss, (and shadowing).

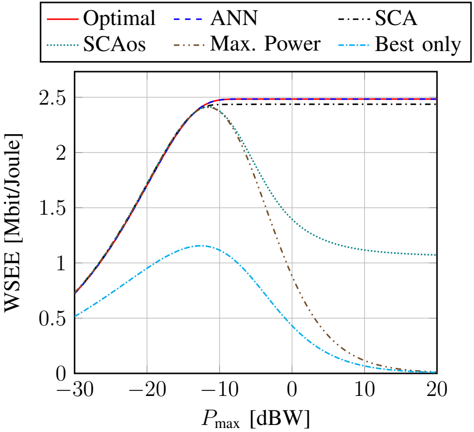
Numerical analysis: Training performance

The 622,000 data samples were generated in 8.4 CPU hours on Intel Haswell nodes with Xeon E5-2680 v3 CPUs running at 2.50 GHz.



Both training and validation error monotonically decrease.
Neither underfitting nor overfitting occurs.

Numerical analysis: Testing performance



- The FNN achieves optimal performance.
- SCA performs close to the FNN only with a sophisticated initialization rule.

For more applications

- **A. Zappone, M. Di Renzo, M. Debbah, T. T. Lam, X. Qian, "Model-Aided Wireless Artificial Intelligence: Embedding Expert Knowledge in Deep Neural Networks Towards Wireless Systems Optimization", submitted, 2018, https://arxiv.org/abs/1808.01672**

- **A. Zappone, L. Sanguinetti, M. Debbah, "User Association (and Load Balancing) for Massive MIMO through Deep Learning", IEEE Asilomar 2018**

- **L. Sanguinetti, A. Zappone, M. Debbah, "On-line power allocation in Massive MIMO", IEEE Asilomar 2018**

Thank you for listening

Alessio Zappone

LANEAS, CentraleSupelec, Paris, France

alessio.zappone@l2s.centralesupelec.fr

CentraleSupélec

B. Matthiesen, A. Zappone, E. A. Jorswieck, and M. Debbah, “Deep learning for optimal energy-efficient power control in wireless interference networks,” *http://export.arxiv.org/pdf/1812.06920*, 2018.