Deep Actor-Critic for Continuous 3D Motion Control in Mobile Relay Beamforming Networks

Spilios Evmorfos†, Athina Petropulu†

†Rutgers, The State University of New Jersey, Piscataway, NJ

May 4, 2022

22-27 MAY 2022

ICASSP 2022

SINGAPORE

Work supported by ARO under grant W911NF2110071

Outline

Outline

Motion Control in Mobile Relay Beamforming Networks

- Next Generation Networks need to accommodate high bandwidth applications
- High bandwidth becomes available at high frequencies
- High frequencies experience high attenuation
- **Relaying** $\Longrightarrow$ extend the communication range
- **Mobile relays** $\Longrightarrow$ more degrees of freedom $\Longrightarrow$ potentially better performance
- We consider **mobile relays** $\Longrightarrow$ urban environments $\Longrightarrow$ spatiotemporally correlated channels

Applications

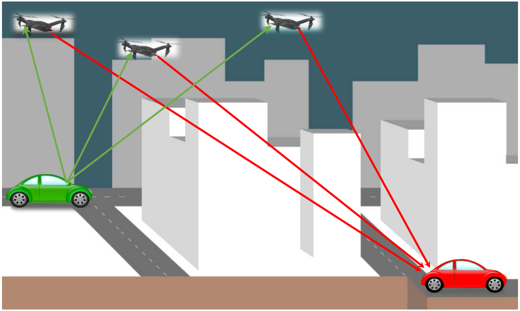


Figure: Urban communications scenario

- Swarm of drones $\implies$ vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communications
- UAVs over a stadium $\implies$ extended coverage and surveillance
- Group of drones $\implies$ search-and-rescue missions

Background

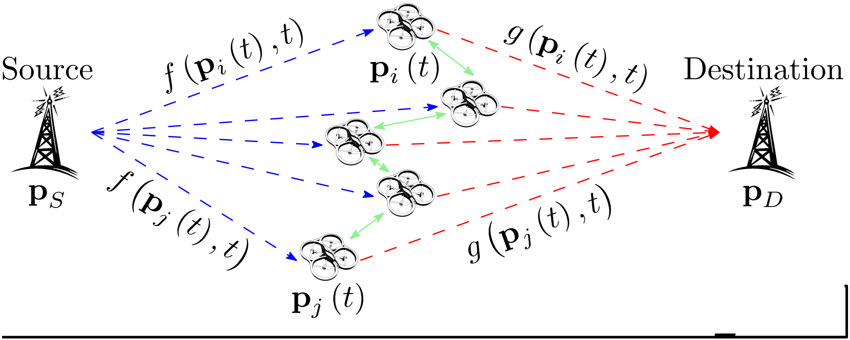
Previous methods:

1. Assume knowledge of channels statistics $\rightarrow$ model-based [Kalogerias, Petropulu, IEEE TSP, 2018]
2. Relays move in 2 dimensions (Rectangular grid) [Huang, Mo, IEEE WCNC, 2018] [Evmorfos, Petropulu, IEEE TSP, 2022]
3. Motion of the relays $\implies$ discrete in space

Our Contributions

- Our approach $\implies$ model-free (no assumptions for channels stats)
- We formulate the problem as a continuous MDP $\implies$ motion continuous in space (<u>but</u> discrete in time)
- Randomness of channels $\implies$ stochastic policies
- We propose a soft actor-critic algorithm with Sinusoidal Representation Networks for the critic
- Continuous control $\implies$ necessary for performance and scaling in 3D motion
- Our proposition $\implies$ excellent performance in 2D and 3D motion $\implies$ without additional complexity or retuning

Outline



- Network with R mobile relays
- Source S, at position $\mathbf{p}_\mathsf{S}$ and Destination D at $\mathbf{p}_\mathsf{D}$
- $\mathbf{p}_\mathsf{S}$ and $\mathbf{p}_\mathsf{D}$ can either belong in $\mathbb{R}^2$ or in $\mathbb{R}^3$
- $f(\mathbf{p}_i(t), t)$ is the channel from the source to the relay i
- $g(\mathbf{p}_i(t), t)$ is the channel from the relay i to the destination
- The channels exhibit correlations with respect to time and space

Set up

LoS communication is not feasible, $\rightarrow R$ relays, each at position $\mathbf{p}_k(t)$

Motion of the relays:

- Time-slotted (time slot denoted as t)
- Confined in a 2D plane or 3D cube

During every time slot t, each relay should:

1. Optimally beamform to destination (maximize SINR)
2. Decide where to move for the next slot

Signal Model

Source S transmits the symbol $s(t) \in \mathbb{C}$ using power $\sqrt{P_S} > 0$

The signal received at the relay located at $\mathbf{p}_k(t)$ is

$$x_k(t) = \sqrt{P} f_k(\mathbf{p}_k, t) s(t) + n_k(t), \quad (1)$$

- f_k: source-relay channel for the k-th relay
- $n_k(t)$: reception noise at the k-th relay, white with variance σ^2

Signal Model (2)

Each relay multiplies the signal, $x_k(t)$, by weight $w_k(t) \in \mathbb{C}$

All R relays transmit the weighted signal simultaneously

The signal received at D equals

$$y(t) = \sum_{k=1}^{R} g_k(\mathbf{p}_\mathsf{D}, t) w_k(t) x_k(t) + n_\mathsf{D}(t), \quad (2)$$

- g_k: relay-destination channel for the k-th relay
- $n_\mathsf{D}(t)$: reception noise at the destination, assumed white with variance σ_D^2

SINR at Destination

Maximum SINR solving w.r.t relay weights, s.t total power constraint:

$$V(t) = \sum_{k=1}^{R} \frac{P_R P_S |f_k(\mathbf{p}_k, t)|^2 |g_k(\mathbf{p}_k, t)|^2}{P_S \sigma_D^2 |f_k(\mathbf{p}_k, t)|^2 + P_R \sigma^2 |g_k(\mathbf{p}_k, t)|^2 + \sigma^2 \sigma_D^2}$$

$$= \sum_{k=1}^{R} V_I(\mathbf{p}_k, t). \quad (3)$$

[Havary-Nassab et al, IEEE TSP, 2008]

- P_R: Total power budget of the relays.
- P_S: Total power budget of the Source.

Outline

Reinforcement Learning (RL) $\implies$ **Markov Decision Process(MDP)**:

The agent, at every time step:

1. experiences state s_t.
2. chooses action a_t from a continuous set of actions A.
3. transitions to the next state s_{t+1}.
4. collects reward r_t.
5. γ, discount factor: how far-sighted the agent is.

Goal: Learn a **Policy** for choosing actions, to maximize the **expected sum of discounted rewards**:

$$R = \mathbb{E}[\sum_{t=t'}^{T} \gamma^{t'-t} r_{t'}]$$

Continuous Control vs Discrete Control

Previous works on relay motion $\rightarrow$ relays move in space in a **discrete fashion**

The drawbacks of discrete control:

- The space needs to be discretized $\rightarrow$ large overhead + unrealistic for real-world deployment
- If motion is considered in the 3D space or better performance is required $\rightarrow$ finer discretization $\rightarrow$ curse of dimensionality in Dynamic Programming

For the above reasons, we consider continuous control $\rightarrow$ the relays can move continuously in the space of interest

Deep Actor-Critic Methods

Deep actor-critic $\implies$ State-of-The-Art in model-free continuous control

Model-free $\implies$ deep neural nets for function approximation

- **Critic (Value Function)**:
  Neural Network: Learns expected sum of rewards from state-action pair (MSE with bootstrapping)
- **Actor (Policy Function)**:
  Neural Network: Learns the action that maximizes the expected sum of rewards from given state (policy gradient)

Outline

1. Introduction
2. Problem Formulation
3. Deep Reinforcement Learning for Continuous Control
4. Proposed Method
5. Experiments
6. Conclusions

MDP for Continuous Relay Motion

To employ deep actor-critic we need to formulate an MDP

SINR expression is distributed, therefore we construct one MDP-Policy shared by all relays

The MDP:

- **state(s)**: position vector of the relay $s = [x, y, z]^T$ (or $s = [x, y]^T$ for the 2D case)
- **action(a)**: relay displacement vector $a = [dx, dy, dz]$ (or $a = [dx, dy]$ for the 2D case)
- **reward(r)**: relay's contribution to the SINR at destination $V_I(\mathbf{p}_k, t) \equiv V_I(s, t)$
- **discount(γ)**: quantification of how far sighted the agent (0.99)

Constraints on the Relay Motion

Relay motion $\Longrightarrow$ continuous in space

But:

- Motion remains discrete w.r.t time
- Clip action to respect space boundaries
- Clip action to avoid collision
- During time displacement interval $\Longrightarrow$ channels do not change

Soft Actor-Critic

Additional requirements for adopting deep actor-critic methods for continuous relay control

- Off-policy: The policy learned $\implies$ different than the one generating the data
- Stochastic Policies: Channel randomness $\implies$ stochastic reward

Soft actor-critic (SAC): [Haarnoja, Zhou et al, ICML, 2018]

- Off-policy
- Stochastic policy
- Model-free continuous control

Vanilla SAC: Direct adoption of soft actor-critic for continuous relay motion control $\implies$ ReLU MLPs for approximating actor and critic

Spectral Bias and Instability

Spectral Bias: Inability of ReLU MLPs to capture high frequencies in low-dimensional regression [Tancik, Srinivasan et al, NeurIPS, 2020]

Actor-critic instability: if critic estimate is inaccurate $\implies$ policy updates accumulate error $\implies$ suboptimal policy

Vanilla SAC:

- Critic $\rightarrow$ ReLU MLP $\rightarrow$ low-dimensional regression via bootstrapping
- Channels are highly varying $\implies$ underlying Value Function has high frequencies

ReLU MLP for the critic $\implies$ low quality policies

The **Sinusoidal Representation Network (SIREN)** architecture was introduced in [V.Sitzmann, J.Martel et al, 2020, NeurIPS] to tackle the *Spectral Bias* of ReLU MLPs

It constitutes of:

- Dense layers
- Sinusoids as activation functions

The SIREN comes with an initialization scheme to handle the periodicity of the activations between layers:

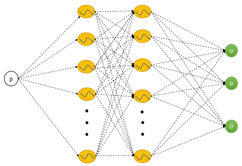


Figure: SIREN architecture - dense layers with sinusoidal activations

SIREN SAC (Our Proposition)

We propose:

1. Soft actor-critic to solve the formulated MDP for continuous relay motion control
2. SIREN for parameterizing the critic

We denote our proposed method as **SIREN SAC**

Outline

Channel Data

We simulate channel data based on a known channel model with spatiotemporal correlations [D. Kalogerias, A. Petropulu, TSP, 2018]

The log magnitude of the channel has 3 additive components:

- Pathloss
- Multipath (Gaussian i.i.d)
- Shadowing (correlation w.r.t time and space)

We perform 2 different sets of experiments

- for 2D plane (20^2)
- for 3D cube (20^3)

Experiments in 2D

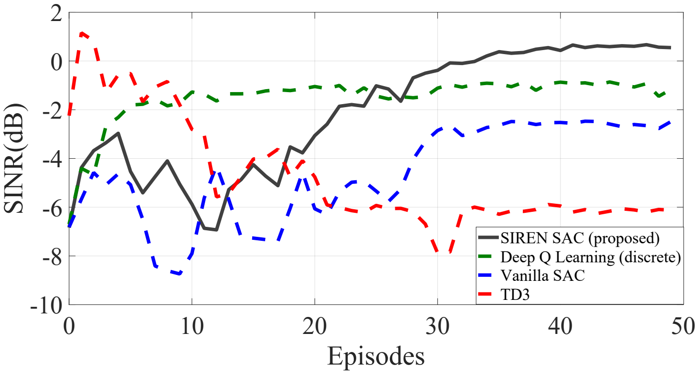


Figure: Average SINR (in db) for 50 episodes (400 slots per episode and 12 different seeds) for the 2D case - 3 relays and 1 source-destination pair

**TD3: The counterpart of soft actor-critic with deterministic policy $\Longrightarrow$ ReLU MLPs [S.Fujimotto et al, ICML, 2018]

Experiments in 3D

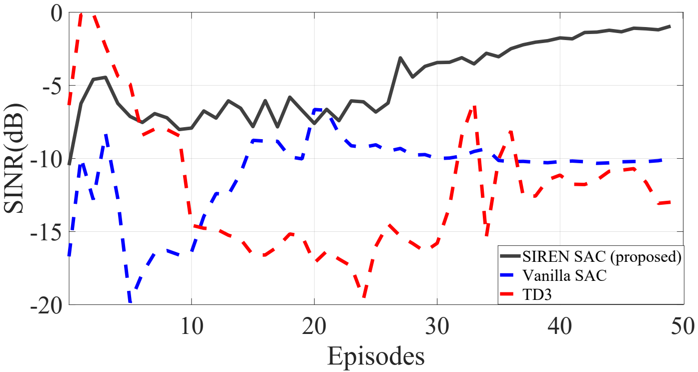


Figure: Average SINR (in db) for 50 episodes (400 slots per episode and 12 different seeds) for the 3D case - 3 relays and 1 source-destination pair

**TD3: The counterpart of soft actor-critic with deterministic policy $\Longrightarrow$ ReLU MLPs [S.Fujimotto et al, ICML, 2018]

Specifications

- Every Network (MLP or SIREN) is comprised by 3 layers
- Each layer has 200 neurons
- batch size of 100 experiences
- the size of the Experience Replay is 1e$+$6
- Adam optimizer with learning rate of 2e-4

Continuous Control Discussion

2D scenario

- Continuous control $\implies$ freedom for relay motion $\implies$ better performance than Deep Q Learning with SIREN (discrete) [Evmorfos, Petropulu et al, IEEE TSP, 2022]

3D scenario

- Continuous control $\implies$ only viable solution, because discretization induces curse of dimensionality $\implies$ Deep Q Learning cannot converge to good policies

Results Discussion

- The employment of SIRENs for Value Function approximation provides significant improvement both in SINR and in stability

- The **SIREN SAC** algorithm retains the 2D performance in the 3D case without additional complexity and tuning

- Employing SIRENs for the **TD3** provides no improvement (testament for the necessity of stochastic policies)

Outline

Conclusions

- We have posed the problem of relay motion control in a continuous model-free set up
- We have focused on off-policy deep actor-critic methods to keep the sample complexity low, which is critical for real-world deployment
- We have provided intuition on why stochastic policies are more suitable than deterministic policies for the problem and verify this with experiments
- We have proposed an adaptation of the soft actor-critic algorithm with SIRENs for Value Function approximation that provides significant boost in overall performance
- We have validated the need for continuous control for scaling to 3D motion (and for better performance in 2D)
- The proposed variation retains the performance of the 2D scenario on the 3D scenario without need for additional complexity or retuning

Reproducibility

- Code for **SIREN SAC**:
  https://github.com/SpiliosEv/SoftActorCriticSIREN3D
- Code for **Vanilla SAC**:
  https://github.com/SpiliosEv/SoftActorCriticVanilla3D
- Code for **TD3**:
  https://github.com/SpiliosEv/TwinDelayed3D

D. Kalogerias, A. Petropulu
Spatially Controlled Relay Beamforming
IEEE Transactions on Signal Processing, vol. 66, no. 24, pp. 6418-6433, 2018.

S. Evmorfos, K. Diamantaras, A. Petropulu
Reinforcement Learning for Motion Policies in Mobile Relaying Networks
IEEE Transactions on Signal Processing, vol 70., pp. 850-861, 2022.

T Haarnoja, T. Zhou, A. Abbeel, S. Levine
Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor
PMLR International Conference Machine Learning, (pp. 1862-1870).

References(2)

M. Tancik, M. Srinivasan, P. mildenhall, B. Fridovich-Keil, S, Raghavan, N. Singhal, R. Ramamoorthi, J. Barron, R. Ng
Fourier Features Let Neural Networks Learn High frequencies in Low Dimensional Domains
NeurIPS p. 7537-7547, 2020.

V. Sitzmann, J. Martel, A. Bergman, D. Lindell, G. Wetzstein
Implicit Neural Representations with periodic activations
NeurIPS p. 7462-7473, 2020.

S. Fujimoto, H. Van Hoof, D. Meger
Addressing function approximation error in actor-critic methods
PMLR International Conference Machine Learning, (pp. 1587-1596).

References(3)

Y.Haung, X. Mo, J.Xu, L.Qiu Y.Zeng
Online Maneuver Design for UAV-Enabled NOMA Systems via Reinforcement Learning,
IEEE WCNC, 2020 pp. 1-6.

V.Havary-Nassab, S.ShahbazPanahi, A.Grami
Distributed Beamforming for Relay Networks based on Second-Order Statistics of the Channel State Information
IEEE TSP, 2008 pp. 4306-4316.

Thank you!