Multiagent Reinforcement Learning based Energy Beamforming Control

Zongqiang Pang,Liping Bai *Member, IEEE,*

***Abstract*—Ultra low power devices make far-field wireless power transfer a viable option for energy delivery despite the exponential attenuation. Electromagnetic beams are constructed from the stations such that wireless energy is directionally concentrated around the ultra low power devices. Energy beamforming faces different challenges compare to information beamforming due to the lack of feedback on channel state. Various methods have been proposed such as one-bit channel feedback to enhance energy beamforming capacity, yet it still has considerable computation overhead and need to be computed centrally. Valuable resources and time is wasted on transfering control information back and forth. In this paper, we propose a novel multiagent reinforcement learning(MARL) formulation for codebook based beamforming control. It takes advantage of the inherienntly distributed structure in a wirelessly powered network and lay the ground work for fully locally computed beam control algorithms. Source code can be found at https://github.com/BaiLiping/WirelessPowerTransfer.**

***Index Terms*—Multiagent Reinforcement Learning,MARL, Wireless Power Transfer, Beamforming**

I. Introduction

WIRELESS power transfer(WPT) can be divided into near-field WPT with inductive coupling, magnetic resonant coupling or capacitive coupling 8357386 and far-field WPT with electromagnetic power beams. [**?**] Compare to near-field WPT, the far-field option has considerable attenuation, yet the increased application of ultra low power devices such as RFID, low power sensor networks [**?**], together with various forms of joint wireless information and power transfer technology such as Simultaneous Wireless Information and Power Transfer (SWIPT) [**?**], Wirelessly Powered Communication Networks (WPCNs) [**?**], Wirelessly Powered Backscatter Communication (WPBC) [**?**] has made far-field WPT an important tool for powering those devices. Current far-field WPT technology can effectively transfer tens of microwatts of RF power to wireless devices from a distance of more than 10 meters. [**?**]

For far-field WPT to be as effective as it can be, directional RF phased array, a group of radiating elements whose phase and magnitude can be controlled to generate a directional beam pattern, [**?**] is utilized to increase the directional gain of power transfer. Digital phased control has high fidelity and is mostly used for communication systems such as 5G Antenna. However, its energy and thermal cost make it prohibitively expensive for other applications. Analog phased control utilizes RF chain to systematically shift the phase discretely. In this paper, we focus on analog phased array.

Nanjing Unversity of Posts and Telecommunications, College of Automation & College of Artificial Intelligence, Nanjing, Jiangsu,210000 China email:zqpang@njupt.edu.cn

There are two kinds of control algorithms for analog beamformer, one is adaptive beamforming, which can adjust according to various channel conditions, but it is expensive in terms of data collection and computational time. A less versatile control algorithm is switch-based control. There are set of predetermined codes for beamforming control. An exhaustive search is performed to find the "optimal code" for the given circumstances[**?**] [**?**]. The codebook exhaustive search algorithm or codebook based beam training process still has a large overhead, particularly for a multi-station scenario. In previous works reinforcement learning based solutions have been proposed where the multi-armed bandit framework [**?**] or Q-learning framework [**?**] was used to render the process more effective.

In this paper, beamforming control is formulated as a multi-agent reinforcement learning problem. Rollout algorithm proposed by Dimitri P. Bertsekas [**?**] is utilized to properly trade-off action space complexity and state-space complexity, hence reducing the learning time. This paper is arranged as the following. In section II, the system model described. In section III, the setup of multiagent reinforcement learning is introduced. In section IV, the problem of WPT is seen through the lense of multiagent reinforcement learning and the simulation result is presented.

II. Enegy Beamforming

A. Uniform Linear Array

The theories of phased array were fully formulated during the WWII era where an array of radars was deployed to detect an accurate angle of arrival [**?**]. Today, phased array hardware is widely available as a commercial product as shown in Figure **??**. Together with various forms of Space Time Signal Processing(STSP), phased array and the beamforming technology has become the enabling components for future communication networks. There are different configurations of arrays, in this paper, we only consider one dimension uniform linear array.

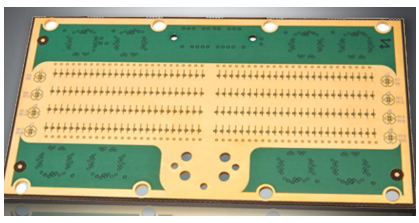

Fig. 1: Pivotal 39GHz Beamformer

B. Channel Model

Suppose there are p propogation path from transmitter to receiver. the gain for each path is denoted by α_i. The channel is modeled as a sum of each path. When Line of Sight(LoS) not available and the number of path is large, Rayleigh fading and a pleathera of channel modelling techniques can be applied to capture Non Line of Sight channel gain. In this paper, we only consider the environment with direct Line of Sight transmission and no reflection path.

$$h = \sum_{i=1}^{p} \alpha_i e^{-j2\pi \frac{d_i}{\lambda}} \quad (1)$$

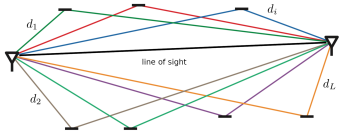

Fig. 2: Multipath Channel Model

C. beamforming codebook $\mathcal{F}$

Let Angle of Departure(AoD) be denoted as φ. Suppose the the range of adjustment for the beamformer is ζ degrees and is deicretized into N portions, each with the angle adjustment of $\frac{\zeta}{N}$ degree. For the i^{th} code in the codebook with AoD of φ_i, the beamforming vector is computed as the following:

$$\mathbf{f} := \mathbf{a}(\varphi_i) = [1, e^{jdcos(\varphi_i)}, ..., e^{jd(M-1)cos(\varphi_i)}]^T \quad (2)$$

D. System Model

The schematics of wirelessly powered communication network is shown in Figure **??**. L energy transmitting node, each equipped with M radiating elements arranged as a uniform linear array, transmit power to K energy receivers scattered in an open field.

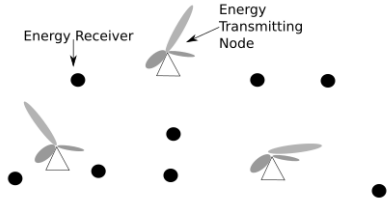

Fig. 3: Wirelessly Powered Network

One challenge for energy beamforming is lack of channel information. Time Division Duplex is a common strategy to implement joint communication and energy transfer, where the wireless power transfer happens from time 0 to P and wireless information transfer(WIT) happens from time P to T. Therefore, no information can be sent before enough energy is stored during the power transfer phase. Ideally, pilots signals should be sent and decoded systematically for channel estimation. [**?**], yet this function is not available for energy beamforming because WIT and WPT functions are realized with separate circuits [**?**], where the latter does not provide decoding capacity. Previous works have proposed methods of channel estimation with only one-bit feedback [**?**], we would adopt this minimum feedback scheme in this paper.

Because the energy beamforming signal **x** does not carry any information, it is assumed to be independent sequences with zero mean and unit variance. [**?**] Furthermore, because we consider the beamformer to be analog, x_1 to x_M are the same signal $x \in \mathbb{C}$.The power in noise is significantly weaker than the energy signal, therefore it can be ignored for practical purposes.

Let $y_{j,p}$ denote the signal received on the j^{th} receiver from the p^{th} transmitter. x_p be the signal transmitted from node p. $\mathbf{f}_p$ be the beamforming code for node p. $h_{i,p}^j$ denote the line of signt channel connecting j^{th} receiver to i^{th} radiating element from p^{th} transmitting node.

$$y_{j,p} = \begin{bmatrix} h_{1,p}^j & h_{2,p}^j & ... & h_{M,p}^j \end{bmatrix} \begin{bmatrix} f_{1,p} \\ . \\ . \\ f_{M,p} \end{bmatrix} x_p$$

In this paper, we assume that the radiating elements from the same node share the same path gain α. Let $\alpha_{j,p}$ denote path gain for line of sight channel connecting j^{th} receiver to p^{th} transmitter.$\varphi_{j,p}$ denote the Angle of Departure connecting j^{th} receiver to p^{th} transmitter. $\mathbf{a}(\varphi_{j,p})^* = [1, e^{jdcos(\varphi_{j,p}}, ..., e^{jd(M-1)cos(\varphi_{j,p})})]$. Therefore:

$$y_{j,p} = \alpha_{j,p}\mathbf{a}(\varphi_{j,p})^*\mathbf{f}_p x_p \quad (3)$$

The received signal on the j^{th} receiver is a summation of signals delievered from all the transmitters to this receiver.

$$y_j = \sum_{p=1}^{L} \alpha_{j,p}\mathbf{a}(\varphi_{j,p})^*\mathbf{f}_p x_p \quad (4)$$

Let the wireless power transfer happening for duration P. Received energe on the j^{th} receiver for duration P is:

$$e_j = \int_0^P |y_j(t)|^2 dt = \int_0^P |\sum_{p=1}^{L} \alpha_{j,p}\mathbf{a}(\varphi_{j,p})^*\mathbf{f}_p x_p(t)|^2 dt \quad (5)$$

E. Wireless Power Transfer

The objective of energy beamforming control is to choose a beamforming code for each energy transmitting node such that the total received power is maximized while satisfying the minimum energy requirement of each energy receiver.

$$\begin{aligned} \underset{\mathbf{f}_p, \forall p}{\text{maximize}} \quad & \sum_{j \in \{1,2,...,L\}} e_j \\ \text{subject to} \quad & \mathbf{f}_p \in \mathcal{F}, \forall p; \\ & e_j \geq e_{min} \end{aligned}$$

III. Reinforcement Learning

A. problem setup

The impetus of reinforcement learning is that an agent can learn by interacting with the environment. In the intersection between control, optimization, and learning, the problem have different mathematical formulations. Here, we follow the problem setup proposed by Richard Sutton in his book Introduction to Reinforcement Learning. [**?**]

Agent can observe the state at each step, denoted as S_t, where t is the t^{th} step taken. For our discussion, we focus only on the subset of problems where state s is fully observable by the agent. There are action choices for each state denoted as A_t. A reward is given for each action taken at step t denoted as R_t. The terminal step is denoted as t=T. For an episodic problem, T is a finite number, for a non-episodic problem, T=∞

An episode of data is registered as an alternating sequence of state, action, and reward:

$$S_0, A_0, R_0, S_1, A_1, R_1 .....S_{T-1}, A_{T-1}, R_{T-1}, S_T, A_T, R_T$$

Gain at step t is defined as the accumulative reward the agent can get from step t onward. A discounting factor γ between 0 to 1 is introduced to incorporate the sense of time, much like how interest rate encodes time in financial systems:

$$G_t := R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + ... + \gamma^{T-t} R_T \tag{6}$$

This can be written in its recursive form, known as Bellman Equation, which is the basis for an iteratively implemented backward induction algorithm:

$$G_t = R_t + \gamma G_{t+1} \tag{7}$$

Transition matrix is intruduced to encode the stochastidy in the environmental dynamics. Transaction Matrix $\mathcal{P}$ is defined as:

$$\mathcal{P}^a_{ss'} := Pr\{S_{t+1} = s' | S_t = s, A_t = a\} \tag{8}$$

State/Action Function q(s,a) is definied as expected gain starting from state s by taking action a:

$$\begin{aligned} q(s,a) :&= \mathbb{E}\{G_t | S_t = s, A_t = a\} \\ &= \mathbb{E}\{\sum_{k=0}^{T-t} \gamma^k R_{t+k+1} | S_t = s, A_t = a\} \end{aligned} \tag{9}$$

Policy is defined as:

$$\pi(s,a) := Pr(A = a | S = s) \tag{10}$$

Optimal Policy is defined as:

$$\pi^*(s) := \arg\max_a q(s,a) \tag{11}$$

Value Function v(s) is defined as the expected gain starting from state s:

$$\begin{aligned} v(s) :&= \mathbb{E}\{G_t | S_t = s\} \\ &= \mathbb{E}\{\sum_{k=0}^{T-t} \gamma^k R_{t+k+1} | S_t = s\} \\ &= \sum_{a \in \mathcal{A}} \pi(s,a) q(s,a) \end{aligned} \tag{12}$$

B. Without Approximation

One obvious approach to learning is to statistically construct a model of the environment, which is called Model-Based Learning. The most primitive form of model-based learning is Bellman Equation based backward induction. Statistical tactics, such as maximum likelihood, Bayesian methods, etc., can be deployed to approximate the model with the least amount of sampling. However, since the environment is implicitly embedded in v(s) and q(s,a), the model building process can be circumvented entirely, hence Model-Free Learning. Depending on whether the iteration rules is policy dependent, model-free learning can be subdivided into on-policy learning and off-policy learning.

One hindrance to the implementation of the brute force backward induction is its memory requirement. A more effective approach is to update q value and v value after one episode, one step, or n steps. They are called Monte Carlo Method, Temporal Difference Method, and $\lambda(n)$ Method respectively.

For online learning, ϵ-greedy Policy $\pi_\epsilon(s)$ is frequently deployed to balance exploration and exploitation, such that the environment can be encoded most efficiently. ϵ is initiated set to 1 and then asymptotically goes to 0 as the episode counts increases.

$$\pi_\epsilon(s,a) = \begin{cases} 1 - \epsilon + \frac{\epsilon}{|A|} & \arg\max_a q_\epsilon(s,a) \\ \frac{\epsilon}{|A|} & \text{otherwise} \end{cases}$$

C. With Approximation

When the problem gets complex, state S becomes a rather large vector and function approximation with neuro networks can be utilized to facilitate learning. Reinforcement learning as a self-sustaining mathematical framework has been refined by Rich Sutton et al. since the 1980s. Only recently, the progress made with Deep Learning has been applied to the realm of Reinforcement Learning [**?**], rendering the computation tenable with existing hardare.

Let the value function and state/action function be parameterized with $\mathbf{w} : \hat{v}(s, \mathbf{w}) \approx v(s)$ and $\hat{q}(s, a, \mathbf{w}) \approx q(s, a)$

Let the i^{th} iteration of parameter be denoted as w_i. The Loss Function $\mathcal{L}(w_i)$ is defined as the following:

$$\mathcal{L}(w_i) := \mathbb{E}\{[v(s) - \hat{v}(s, w_i)]^2\} \tag{13}$$

$$\mathcal{L}(w_i) := \mathbb{E}\{[q(s,a) - \hat{q}(s, a, w_i)]^2\} \tag{14}$$

While the real value of v(s) and q(s,a) are not knowable, it can be approximated:

$$v(s) \approx \sum_{a \in A} R(s,a) + \gamma v(s', \mathbf{w}) \tag{15}$$

$$q(s,a) \approx r + \gamma \arg\max_a q(s', a', \mathbf{w}) \tag{16}$$

The Gradient of weighing paramter **w** can be derived from **??** and **??** with the real values substituted by **??** and **??** respectively. By convention, constant is omitted. Parameter is updated following Gradient Descent:

$$\mathbf{w}_i = \mathbf{w}_{i-1} - \nabla_{w_{i-1}} \mathcal{L}(w_{i-1}) \tag{17}$$

D. Policy Gradient Methods

Policy $\pi(s)$ can be written as a function parameterized by θ with s as input and a smooth distribution overall all actions as output.By adjusting parameter θ we can adjust the distribution over action choices for different states. This style of learning is called policy gradient-based learning.

Let us register a path sequence taken by the agent as τ such that the sequence is denoted as $\{S_{\tau 0}, A_{\tau 0}, R_{\tau 0} ... S_{\tau T}, A_{\tau T}, R_{\tau T}\}$. the gain of sequence τ is defined as the gain of this entire sequence of state, action, reward:

$$G(\tau) := \sum_{t=0}^{T} \gamma^t R_t \tag{18}$$

Denote $P(\tau, \theta)$ as the probability that path τ is travesed when the policy is parameterized by θ. The Objective Function can be defined in various ways. Here we adopt the definition as the following:

$$U(\theta) = \sum_{\tau} P(\tau, \theta) G(\tau) \tag{19}$$

The objective of the policy gradient method is to find the parameter θ to maximize the objective function.

The gradient of aforementioned utility function is:

$$\nabla_\theta U(\theta) = \nabla_\theta \sum_{\tau} P(\tau, \theta) G(\tau) \tag{20}$$

A mathematical sleight of hand called Importance Sampling is deployed to convert this theoretical expression of gradient into something that is algorithmically feasible.

$$\nabla_\theta U(\theta) \approx \frac{1}{N} \sum_{\tau=1}^{N} \sum_{t=0}^{T-1} \nabla_\theta ln \pi_\theta(s,a)|_{\theta_{old}} [q^{\pi_{\theta_{old}}}(s,a) - b] \tag{21}$$

We can use stochastic gradient descent(SGD) method to update θ:

$$\theta = \theta_{old} - \alpha \nabla_\theta ln \pi_\theta(s,a)|_{\theta_{old}} [q^{\pi_{\theta_{old}}}(s,a) - b] \tag{22}$$

Actor-Critic Method takes advantage of both policy gradient and function approximation to build a bootstrap structure that lead up to fast convergence. state/action function for policy $\pi_\theta(s)$ is approximated by $q^{\pi_\theta}(s, a, \mathbf{w})$.Baseline b is introduced into the bootstrap structure to foster convergence. Different algorithms define baseline differently. In advantage Actor-Critic algorithm, baseline is defined as a value function based on π_θ.Because the SGD updating process does not rely on the ordering of things, it is obvious that some of the aforementioned computations can be done asynchronously. Asynchronous Advantage Actor-Critic (A3C) is proven one of the most effective agents for renforcement learning, and is the one we will use in this paper.

E. Multiagent Reinforcement Learning

$A_t = \{A_t^1, A_t^2, ..., A_t^M\}$ M is the number of agents. The action space is cartician product of action choices available to each agent. $A_t(s) = A_t^1(s) \times A_t^2(s) \times ... \times A_t^M(s)$, which grows exponentially as the number of agents grows.

The Rollout method proposed by Dimitri Bertsekas breakdown this collective decision into its sequential components, reducing the complexity of action space while increasing the complexity of state space. It is proven that the intermediate state rollout method yields the same result as does the regular method. [**?**]

Without intermidiate state rollout, the sequences of data collected is: $...S_t, A_t, R_t, S_{t+1}...$ as shown in Figure **??**

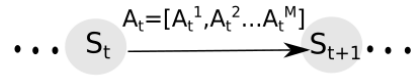

Fig. 4: Without Rollout

The intermediate states rollout technique converting action space complexity into state-space complexity by introducing intermediate states, denoted as S_t^k where k goes from 1 to M-1. The sequence of data is now: $...S_t, A_t^1, R_t^1, S_t^1, A_t^2, R_t^2, S_t^2, ..., S_t^{M-1}, A_t^M, R_t^M, S_{t+1}...$ as shown in Figure **??**

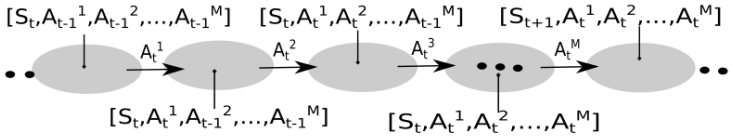

Fig. 5: With Rollout

suppose each agent has N choices. This formulation reduces the size action space from N^M to $N \times M$.

IV. Beamforming as a Multiagent Reinforcement Learning problem

A. Environment

The wirelessly powered communication network has L energy transmitting stations positioned at the corner of a 30m x 30m field. K energy receivers randomly scattered between 1m to 29m as illustrated by Figure **??**. 0.5s of energy transfer is followed with 0.5s of information transfer. Assume no energy leftover at each cycle, such that at the beginning of the next energy transfer interval, the remaining power at each energy receiver is 0.

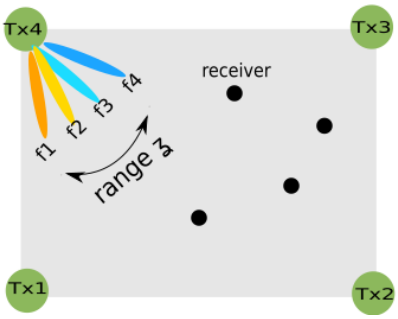

Fig. 6: Environment

Number of Energy Transmitting Nodes	L=4
Positions of Trasmitting Node	TX_1(0,0) TX_2(30,0) TX_3(30,30) TX_4(0,30)
Number of Radiation Elements per Trasmitting Node	M=64
Energy Carrier Frequency	8M Hz
Field of Energy Receivers	30m x 30m
Number of Energy Receivers	K
Energy Transfer Time	0.5s
Information Transfer Time	0.5s
Maximum Number of Steps	100

Observation Space:$\{e_1, e_2, ...e_K, c_1, c_2, c_3, c_4\}$ where e_j is the energy received at the j^{th} receiver, c_i is the codebook choice for the i^{th} energy emitting node.

Reward: If $e_j < e_min$, reward is deducted by 50 points each. If $e^n_{total}ew > e^n_{total}ew$, reward is increased by 100 points. If $e^n_{total}ew < e^n_{total}ew$, reward is deduced by 300 points.

B. A3C agent

Layers of Actor Network	3
Layers of Critic Network	3
Learning Rate for Actor	α_a=0.1
Learning Rate for Critic	α_c=0.1
Discount Rate	γ=0.9
Action Function	Softmax

C. Simulation Result

V. Conclusion

In this paper, we demonstrated the possibility to formulate WPT as a multiagent reinforcement learning problem, this lays the groundwork for further study towards fully locally computed control algorithms for wirelessly powered communication networks. Instead of group actions of all agents together, a multiagent rollout approach sees things sequentially, action taken by one agent becomes part of the state of another. This framework deduces the dimension of action space from exponential growth to multiplicative growth, and it can be applied to other problems. The most recent incarceration of beamforming technology is a passive reflective surface, or Intelligent Reflective Surface(IRS), where the reflective components are in the thousands. The multiagent approach proposed in this paper could be applied to IRS control as well, which should be a fruitful topic of future studies.

References

[1] Z. Popovic, "Near- and far-field wireless power transfer," in *2017 13th International Conference on Advanced Technologies, Systems and Services in Telecommunications (TELSIKS)*, 2017, pp. 3–6.

[2] T. Hiramoto, K. Takeuchi, T. Mizutani, A. Ueda, T. Saraya, M. Kobayashi, Y. Yamamoto, H. Makiyama, T. Yamashita, H. Oda, S. Kamohara, N. Sugii, and Y. Yamaguchi, "Ultra-low power and ultra-low voltage devices and circuits for iot applications," in *2016 IEEE Silicon Nanoelectronics Workshop (SNW)*, 2016, pp. 146–147.

[3] T. D. Ponnimbaduge Perera, D. N. K. Jayakody, S. K. Sharma, S. Chatzinotas, and J. Li, "Simultaneous wireless information and power transfer (swipt): Recent advances and future challenges," *IEEE Communications Surveys Tutorials*, vol. 20, no. 1, pp. 264–302, 2018.

[4] S. Bi, Y. Zeng, and R. Zhang, "Wireless powered communication networks: an overview," *IEEE Wireless Communications*, vol. 23, no. 2, pp. 10–18, 2016.

[5] K. Han and K. Huang, "Wirelessly powered backscatter communication networks: Modeling, coverage and capacity," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1–6.

[6] B. D. Van Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[7] Junyi Wang, Zhou Lan, Chang-woo Pyo, T. Baykas, Chin-sean Sum, M. A. Rahman, Jing Gao, R. Funada, F. Kojima, H. Harada, and S. Kato, "Beam codebook based beamforming protocol for multi-gbps millimeter-wave wpan systems," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 8, pp. 1390–1399, 2009.

[8] D. J. Love, R. W. Heath, and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2735–2747, 2003.

[9] V. Va, T. Shimizu, G. Bansal, and R. W. Heath, "Online learning for position-aided millimeter wave beam training," *IEEE Access*, vol. 7, pp. 30 507–30 526, 2019.

[10] M. Cui, G. Zhang, and R. Zhang, "Secure wireless communication via intelligent reflecting surface," *IEEE Wireless Communications Letters*, vol. 8, pp. 1410–1414, 2019.

[11] D. P. Bertsekas, "Multiagent rollout algorithms and reinforcement learning," *ArXiv*, vol. abs/1910.00120, 2019.

[12] T. K. Sarkar, R. Mailloux, A. A. Oliner, M. Salazar-Palma, and D. L. Sengupta, *A History of Phased Array Antennas*, 2006, pp. 567–603.

[13] M. Biguesh and A. B. Gershman, "Training-based mimo channel estimation: a study of estimator tradeoffs and optimal training signals," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 884–893, 2006.

[14] J. Xu and R. Zhang, "Energy beamforming with one-bit feedback," *IEEE Transactions on Signal Processing*, vol. 62, no. 20, pp. 5370–5381, 2014.

[15] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[16] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *ArXiv*, vol. abs/1312.5602, 2013.