Echo State Transfer Learning for Data Correlation Aware Resource Allocation in Wireless Virtual Reality

Mingzhe Chen*, Walid Saad†, Changchuan Yin*, and Mériouane Debbah‡

*Beijing University of Posts and Telecommunications, Beijing, China 100876, Emails: [chenmingzhe@bupt.edu.cn, ccyyin@ieee.org]
†Wireless@VT, Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA, Email: walids@vt.edu.
‡Mathematical and Algorithmic Sciences Lab, Huawei France R & D, Paris, France, Email: merouane.debbah@huawei.com.

Abstract—In this paper, the problem of data correlation-aware resource management is studied for a network of wireless virtual reality (VR) users communicating over cloud-based small cell networks (SCNs). In the studied model, small base stations (SBSs) with limited computational resources act as VR control centers that collect the tracking information from VR users over the cellular uplink and send them to the VR users over the downlink. However, this work is a qualitative survey that does not provide any rigorous wireless VR model. In [2], a channel of China under Grants 61671086 and 6162910, by the U.S. National Science

I. INTRODUCTION

Virtual reality (VR) can enable users to virtually hike the Grand Canyon or engage in a secret adventure as a video game hero without leaving their room. However, due to the wired connections of conventional VR devices, the users can only take a restricted set of actions which, in turn, limits the VR application space. To enable immersive VR applications, VR systems can be operated using wireless networking technologies. However, operating VR devices over wireless small cell networks (SCNs) faces many challenges that include effective image compression, tracking, and low-latency computation and communication.

The existing literature has studied a number of problems related to wireless VR such as in [1]–[4]. The authors in [1] exposed the future challenges of VR systems over a wireless network. However, this work is a qualitative survey that does not provide any rigorous wireless VR model. In [2], a channel access scheme for wireless multi-user VR system is proposed. The authors in [3] proposed an alternate current magnetic field-based tracking system to track the position and orientation of a VR user’s head. However, the recent works in [2] and [3] do not develop a VR model that can capture all factors of VR QoS and they only analyze a single VR metric such as delay or tracking accuracy. In [4], a channel of China under Grants 61671086 and 6162910, by the U.S. National Science

This work was supported in part by the National Natural Science Foundation of China under Grants 61671086 and 6162910, by the U.S. National Science Foundation under Grants IIS-1633363, CNS-1460316, and CNS-1617896 and by the ERC Starting Grant 305123 MORE (Advanced Mathematical Tools for Complex Network Engineering).

We propose a novel VR model to jointly capture the downlink and uplink transmission delay, backhaul transmission delay, and computational time thus effectively quantifying the VR delay for all users in a wireless VR network.

For the considered wireless VR applications, we analyze the allocation of resource blocks jointly over the uplink and downlink, along with the allocation of computational resources in the uplink. We formulate the problem as a noncooperative game in which the players are the small base stations (SBSs). Each player seeks to find an optimal resource allocation scheme to optimize a utility function that captures the VR delay.

To find a Nash equilibrium of this game, we propose a transfer learning algorithm based on echo state networks (ESNs) [5]. The proposed algorithm can intelligently transfer information on the learned utility across time, and, hence, allow adaptation to environmental dynamics due to factors such as changes in the users’ data correlation.

Simulation results show that the proposed algorithm can, respectively, yield 16.7% and 18.2% gains in terms of delay compared to Q-learning with data correlation and Q-learning without data correlation.

II. SYSTEM MODEL AND PROBLEM FORMULATION

Consider the downlink and uplink transmissions of an SCN servicing a set $U$ of $U$ wireless VR users and a set $B$ of $B$
SBSs. Here, the downlink is used to transmit the VR images displayed on each user’s VR device while the uplink is used to transmit the tracking information that is used to determine each VR user’s location and orientation. The SBSs are connected to a cloud via capacity-constrained backhaul links and the SBSs serve their users using the cellular band. $V_F$ represents the maximum backhaul transmission rate for all users. Here, we focus on entertainment VR applications such as watching immersive videos and playing immersive games.

In our model, the SBSs adopt an orthogonal frequency division multiple access (OFDMA) technique and transmit over a set of $V$ of $V$ uplink resource blocks and a set of $S$ of $S$ downlink resource blocks. The coverage of each SBS is a circular area with radius $r$ and each SBS only allocates resource blocks to the users located in its coverage range. We also assume that the resource blocks of each SBS will all be allocated to the associated users.

A. Data Correlation Model

1) Downlink Data Correlation Model: In VR wireless networks, multiple VR users may play the same immersive game with different locations and orientations. In this case, the cloud can exploit the data correlation between the users that are playing the same immersive game to reduce the traffic load of backhaul links. For example, when the users are watching the same immersive sports game, the cloud can extract the difference between the VR images of these users and will need to only transmit to an SBS the data that is unique to each user. However, when the VR users are playing different immersive games, the data correlation between the users is low and, hence, the cloud needs to transmit entire VR images to the associated VR users. In order to define the data correlation of VR images, we first assume that the number of pixels that user $i$ needs to construct the VR images is $N_i$ and the number of different pixels between any pair of users $i$ and $k$ is $N_{ik}$. Here, $N_{ik}$ is calculated by the cloud using image processing methods such as motion search [6]. Then, the data correlation between user $i$ and user $k$ can be defined as follows:

$$\phi_{ik} = \frac{N_{ik}}{N_i + N_k},$$

where $N_k$ is the number of pixels that user $k$ needs to construct the VR images during a period. Indeed, (1) captures the data correlation between the images of users $i$ and $j$. From (1), we can see that, when user $i$ and user $k$ are associated with the same SBS, the cloud needs to only transmit $N_i + N_j - (N_i + N_j) \phi_{ij}$ pixels to that SBS.

2) Uplink Data Correlation Model: In the uplink, the users must transmit the tracking information to the SBSs. The tracking information is collected by the sensors placed at a VR user’s headset or near the VR user. It has been shown that, for commonly used data-gathering applications, the data source can be modeled as a Gaussian field [7]. The uplink data is collected by the sensors and, hence, the uplink data can be assumed to follow the Gaussian distribution. We can assume that the tracking data, $X_i$, collected by each VR user $i$ is a Gaussian random variable with mean $\mu_i$ and variance $\sigma_i^2$. In wireless VR, observations from proximal VR devices are often correlated due to the dense deployment density. Hence, we consider the power exponential model [8] to capture the spatial correlation of VR tracking data. Here, the covariance $\sigma_{ij}$ between user $i$ and user $j$ separated by distance $d_{ij}$ is:

$$\sigma_{ij} = \text{cov}(X_i, X_j) = \sigma_i \sigma_j e^{-d_{ij}/\kappa},$$

where $\alpha$ and $\kappa$ capture the significance of distance variation on data correlation.

B. Delay Model

In our model, the VR images are transmitted from the cloud to the SBSs then to the users. The tracking information is transmitted from the users to the SBSs and processed at each corresponding SBS. In this case, the backhaul links are only used for VR image transmission and the transmission rate of each VR image from the cloud to the SBSs can be given as $V_{F_i} = \frac{1}{\tau}$. Here, we assume that the backhaul transmission rate of each user is equal and we do not consider the optimization of the backhaul transmission. In a VR model, we need to capture the VR transmission requirements such as high data rate, low delay, and accurate tracking and, hence, we consider the transmission delay as the main VR QoS metric of interest. The downlink rate of user $i$ associated with SBS $j$ is:

$$c_{ij} = s_{ij} B \log_2 (1 + \gamma_{ij,k}),$$

where $s_{ij} = [s_{ij,1}, \ldots, s_{ij,S}]$ is the vector of resource blocks that SBS $j$ allocates to user $i$ with $s_{ij,k} \in \{0,1\}$. Here, $s_{ij,k} = 1$ indicates that resource block $k$ is allocated to user $i$. \$gamma_{ij,k} = \frac{P_{ij} h_{ij,k}^2}{\sum_{i' \in R^k, i' \neq j} P_{i'j} h_{i'j,k}^2}\$ is the signal-to-interference-plus-noise ratio (SINR) between user $i$ and SBS $j$ over resource block $k$. $R^k$ represents the set of the SBSs that use downlink resource block $k$. $B$ is the bandwidth of each subcarrier, $P_{ij}$ is the transmit power of SBS $j$ which is assumed to be equal for all SBSs, $N_0^2$ is the variance of the Gaussian noise and $h_{ij,k}^2 = \phi_{ij}^2 \beta$ is the path loss between user $i$ and SBS $j$ over resource block with $\phi_{ij}^2$ is the Rayleigh fading parameter, $d_{ij}$ is the distance between user $i$ and SBS $j$, and $\beta$ is the path loss exponent. Based on (1) and (4), the downlink transmission delay at time slot $t$ is:

$$D_{ij} = \left( L_i (\phi_{ij}^{\text{max}}), s_{ij} \right) = \frac{L_i (\phi_{ij}^{\text{max}})}{c_{ij} (s_{ij}) + 1} + \frac{L_i (\phi_{ij}^{\text{max}})}{V_{F_i}},$$

where $L_i (\phi_{ij}^{\text{max}})$ is the data that user $i$ needs to construct a VR image during a period and $\phi_{ij}^{\text{max}} = \max_{k \in U^i} (\phi_{ik})$ is the maximum downlink data correlation between user $i$ and other users associated with SBS $j$. Finding the maximum data correlation allows minimizing the downlink transmission data transmitted in the downlink and that will be used construct a VR image. Here, the first term is the transmission time from SBS $j$ to user $i$ and the second term is the transmission time from the cloud to SBS $j$. We assume that $P_U$ is the transmit power of each user which is assumed to be equal for all users. The bandwidth of each uplink resource block is also $B$. In this case, the uplink rate of each user $i$ associated with SBS $j$ is:

$$c_{ij} (v_{ij}) = \sum_{k=1}^V v_{ij,k} B \log_2 (1 + \gamma_{ij,k}),$$

where $v_{ij} = [v_{ij,1}, \ldots, v_{ij,V}]$ is the vector of resource blocks that SBS $j$ allocates to user $i$ with $v_{ij,k} \in \{0,1\}$. Here, $\gamma_{ij,k} = \frac{P_{ij} h_{ij,k}^2}{\sum_{i' \in U^k, i' \neq j} P_{i'j} h_{i'j,k}^2}$ is the SINR between user $i$ and SBS $j$ over resource block $k$ with $U^k$ represents the set of users that use uplink resource block $k$. In this case, the uplink transmission
delay can be given by $\frac{K_i(\sigma_{ij}^{\text{max}})}{c_{ij}(v_{ij})}$, where $K_i$ is the data that needs to be transmitted and $\sigma_{ij}^{\text{max}} = \max_{k \in SBS_j, k \neq i} (\sigma_{ik})$ is the maximum uplink data correlation between user $i$ and other SBS $j$’s associated users. Similarly, finding the maximum data correlation allows minimizing the uplink transmission data that SBS $j$ uses to determine user $i$’s location and orientation.

In the uplink, the tracking information can be directly processed by the SBSs that have limited computational power. The computational resource of each SBS, $c$, represents its ability to compute the tracking data. Each SBS $j$ will allocate the total computational power to the associated users and, hence, $m_{ij}$ is used to represent the computational power that SBS $j$ allocates to user $i$ with $\sum_{i \in U_j} m_{ij} = m$. $U_j$ represents the set of the users associated with SBS $j$. The computational time of SBS $j$ that processes the tracking data collected by user $i$ is $\frac{K_j(\sigma_{ij}^{\text{max}})}{m_{ij}}$ and the total uplink delay can be given by:

$$D^u_{ij}(K_i(\sigma_{ij}^{\text{max}}), v_{ij}, m_{ij}) = \frac{K_j(\sigma_{ij}^{\text{max}})}{c_{ij}(v_{ij})} + \frac{K_i(\sigma_{ij}^{\text{max}})}{m_{ij}},$$

where the first term is the transmission time from user $i$ to SBS $j$ and the second term is the computation time for user $i$’s data. The computation time depends on the computational resources that SBS $j$ allocates to each user that will affect the uplink delay.

C. Utility Function Model

In order to jointly consider the transmission delay in both uplink and downlink, we introduce a method based on the framework of multi-attribute utility theory [10] to construct an appropriate utility function to capture transmission delay in both uplink and downlink. We first introduce the utility functions of transmission delay in uplink and downlink, separately. Then, we formulate the utility function based on [10].

The utility function of downlink transmission delay is constructed based on the normalization of downlink transmission delay, which can be given by:

$$D_{ij}(L_i(\phi_{ij}^{\text{max}}), s_{ij}) = \begin{cases} \frac{D_{ij}^{\text{max}} - D_{ij}(L_i(\phi_{ij}^{\text{max}}), s_{ij})}{D_{ij}^{\text{max}} - \gamma_D}, & D_{ij}(L_i(\phi_{ij}^{\text{max}}), s_{ij}) \geq \gamma_D, \\ 1, & D_{ij}(L_i(\phi_{ij}^{\text{max}}), s_{ij}) < \gamma_D, \end{cases}$$

where $\gamma_D$ is the maximum tolerable delay for each VR user (maximum supported by the VR system being used) and $D_{ij}^{\text{max}} = \max(D_{ij}(L_i(0), s_{ij}))$ is the maximum transmission delay. From (7), we can see that, when the downlink transmission delay is smaller than $\gamma_D$, the utility value will remain at 1. This is due to the fact when the delay meets the system requirement, the network will encourage the SBSs to reallocate the resource blocks to other users.

The utility function for the uplink transmission is:

$$U_{ij}(s_{ij}, v_{ij}, m_{ij}) = \frac{D_{ij}^{u}(K_i(\sigma_{ij}^{\text{max}}), v_{ij}, m_{ij})}{D_{ij}^{u}(K_i(\sigma_{ij}^{\text{max}}), v_{ij}, m_{ij})} \frac{D_{ij}^{u}(K_i(\sigma_{ij}^{\text{max}}), v_{ij}, m_{ij})}{\gamma_D},$$

where $\gamma_u$ is the maximum tolerable delay for the VR tracking information transmission and $D_{ij}^{u} = \max(D_{ij}(K_i(0), v_{ij}, m_{ij}))$ is the maximal uplink delay. Based on (7) and (8), the total utility function that captures both downlink and uplink delay for user $i$ associated with SBS $j$ is:

$$U_{ij}(s_{ij}, v_{ij}, m_{ij}) = \frac{D_{ij}(L_i(\phi_{ij}^{\text{max}}), s_{ij})}{D_{ij}(L_i(\phi_{ij}^{\text{max}}), s_{ij})} \frac{D_{ij}(L_i(\phi_{ij}^{\text{max}}), v_{ij}, m_{ij})}{\gamma_D}.$$
fractions of SBS $j$’s total computational resource $m_j$. We assume that each SBS $j$ adopts one action at each time slot $t$. Then, the utility function of each SBS $j$ can be given by:

$$u_j(a_j, a_{-j}) = \frac{1}{T} \sum_{t=1}^{T} \sum_{i \in A_j} U_{i,j,t} (s_{i,j}, v_{i,j}, c_{i,j}),$$

(13)

where $a_j \in A_j$ is an action of SBS $j$ and $a_{-j}$ denotes the action profile of all SBSs other than SBS $j$. Indeed, $u_j$ captures the average utility value of each SBS $j$. Let $\pi_{j,a_{ij}} = \frac{1}{T} \sum_{t=1}^{T} 1_{(a_{ij} = a_{ij})} = \Pr(a_{ij} = a_{ij})$ be the probability of SBS $j$ using action $a_{ij}$ at time $t$. Let $\pi_j = (\pi_{j,a_{ij}})_{i \in A_j}$ be the action selection mixed strategy of SBS $j$ with $|A_j|$ being the number of actions of SBS $j$. Based on the definition of the strategy, the utility function in (13) is given by:

$$u_j(a_j, a_{-j}) = \frac{1}{T} \sum_{t=1}^{T} \sum_{a \in A} U_{j,t}(a_j, a_{-j}, \pi_{j,a_{ij}}),$$

(14)

where $a \in A$ being the action set of all SBSs.

One suitable solution for this game is the mixed-strategy Nash equilibrium (NE), formally defined as follows [12]: A mixed-strategy profile $\pi^* = (\pi^*_j, \ldots, \pi^*_B) = (\pi^*_j, \pi^*_{-j})$ is a mixed-strategy Nash equilibrium if, $\forall j \in R$ and $\pi_j$, we have:

$$u_j(\pi^*_j, \pi^*_{-j}) \geq u_j(\pi^*_j, \pi_{-j}),$$

(15)

where $u_j(\pi_n, \pi_{-n}) = \sum_{a \in A} U_j(a_j, a_{-j}) \prod_{j \in B} \pi_{j,a_{ij}}$ is the expected utility of SBS $j$ when it selects the mixed strategy $\pi_j$. For our game, the mixed-strategy NE for the SBSs represents a solution of the game at which each SBS $j$ can minimize the delay for its associated users, given the actions of its opponents.

III. ECHO STATE NETWORKS FOR SELF-ORGANIZING RESOURCE ALLOCATION

Next, we introduce a transfer reinforcement learning (RL) algorithm that can be used to find an NE of the VR game. To satisfy the delay requirement for the VR transmission, we propose a transfer RL algorithm based on the neural networks framework of ESNs [16]. Traditional RL algorithms such as Q-learning typically rely on a Q-table to record the utility value. However, as the number of players and actions increases, the number of utility values that the Q-table needs to include will increase exponentially and, hence, the Q-table may not be able to record all of the needed utility values. However, the proposed algorithm uses a utility function approximation method to record the utility value and, hence, it can be used for large networks and large utility spaces. Moreover, a dynamic network in which the users’ computational resources and data correlation may change across the time, traditional RL algorithms need to be executed each time the network changes. However, the proposed ESN transfer RL algorithm can find the relationship of the utility functions when the environment changes. After learning this relationship, the proposed algorithm can use the historic learning result to find a mixed strategy NE.

The proposed transfer RL algorithm consists of two components: (i) ESN-based RL algorithm and (ii) ESN-based transfer learning algorithm. The ESN-based RL algorithm is based on our work in [4], and, thus, here, we just introduce the ESN-based transfer learning algorithm.

We first assume that, before the users’ state information changes, the mixed strategy, action, and utility of each SBS $j$ are $\pi_j$, $a_j$, and $u_j(a_j, a_{-j})$, while the strategy, action, and utility of SBS $j$, after the users’ state information changes, are $\pi'_j$, $a'_j$, and $u'_j(a_j, a_{-j})$. Since the number of users associated with SBS $j$ is unchanged, the action and mixed strategy sets of SBS $j$ will not change when the users’ state information changes. In this case, the proposed ESN-based transfer learning algorithm is used to find the relationship between $u'_j(a_j, a_{-j})$ and $u_j(a_j, a_{-j})$ when SBS $j$ only knows $u'_j(a_j, a_{-j})$. This means that the proposed algorithm can transfer the information from the already learned utility $u'_j(a_j, a_{-j})$ to the new utility $u'_j(a_j, a_{-j})$ that must be learned. The ESN-based transfer learning algorithm of each SBS $j$ consists of three components: (a) input, (b) output, and (c) ESN model, which are given by:

- **Input**: The ESN-based transfer learning algorithm takes the strategies of the SBSs and the action of SBS $j$ uses at time $t$ as input which is given by $x'_{j,t} = [\pi_1, \ldots, \pi_B, a_{j,t}]^T$.

- **Output**: The output of the ESN-based transfer learning algorithm at time $t$ is the deviation of the utility values when the users’ information changes $y'_{j,t} = u'_j(a_{j,t}) - u_j(a_{j,t})$.

- **ESN Model**: An ESN model is used to find the relationship between the input $x'_{j,t}$ and output $y'_{j,t}$. The ESN model consists of the output weight matrix $W_{out}^T \in \mathbb{R}^{1 \times N_w}$ and the dynamic reservoir containing the input weight matrix $W_{in}^T \in \mathbb{R}^{|x| \times (B+1)}$, and the recurrent matrix $W_{r} \in \mathbb{R}^{N_w \times N_w}$ with $N_w$ being the number of the dynamic reservoir units. Here, the dynamic reservoir is used to store historic ESN information that includes input, reservoir state, and output. This information is used to build the relationship between the input and output.

The update process of the dynamic reservoir will be given by:

$$\mu'_{j,t} = f(W_{r}' \mu'_{j,t-1} + W_{in}' x'_{j,t}),$$

(16)

where $f(x) = \frac{e^x - 1}{e^x + 1}$ is the tanh function. Based on the dynamic reservoir state, the ESN-based transfer learning algorithm will combine with the output weight matrix to approximate the deviation of the utility value, which can be given by:

$$y'_{j,t} = W_{out}' \mu'_{j,t},$$

(17)

where $W_{out}'$ is the output weight matrix at time slot $t$.

$$W_{r}' = W_{r}' + \lambda' (\hat{u}'_{j}(a_{j,t}) - \hat{u}_{j}(a_{j,t}) - y'_{j,t}) \mu'_{j,t},$$

(18)

where $\lambda'$ is the learning rate, and $\hat{u}'_{j}(a_{j,t})$ is the actual deviation between two utility values. In this case, the ESN-based transfer learning algorithm can find the relationship between the utility functions when the users’ state information changes and, hence, reduce the iterations of the RL algorithm to learn the new utility values. The proposed, distributed ESN-based learning algorithm performed by each SBS $j$ is summarized in Table I. The proposed algorithm is guaranteed to converge to an NE and this convergence follows from [4].

IV. SIMULATION RESULTS

For our simulations, we consider an SCN deployed within a circular area with radius $r = 100$ m. $U = 25$ users and $B = 4$ SBSs are uniformly distributed in this SCN area. The rate requirement of VR transmission is $25.32$ Mbit/s [4]. The detailed
In this paper, we have proposed a novel resource allocation framework for optimizing delay for wireless VR services with data correlation. We have formulated the problem as a noncooperative game and proposed a novel transfer learning algorithm based on echo state networks to solve the game. The proposed learning algorithm can use the existing learning result to directly find the optimal resource allocation when the users' state information changes and, hence, can quickly converge to a mixed-strategy NE. Simulation results have shown that the proposed algorithm has a faster convergence time than Q-learning and guarantees low delays for VR services.

V. CONCLUSION



![Fig. 2. Convergence of the proposed algorithm and Q-learning.](image)

In this paper, we have proposed a novel resource allocation framework for optimizing delay for wireless VR services with data correlation. We have formulated the problem as a noncooperative game and proposed a novel transfer learning algorithm based on echo state networks to solve the game. The proposed learning algorithm can use the existing learning result to directly find the optimal resource allocation when the users’ state information changes and, hence, can quickly converge to a mixed-strategy NE. Simulation results have shown that the proposed algorithm has a faster convergence time than Q-learning and guarantees low delays for VR services.

### REFERENCES

[1] E. Baštug, M. Bennis, M. Médard, and M. Debbah, “Towards interconnected virtual reality: Opportunities, challenges and enablers,” IEEE Communications Magazine, vol. 55, no. 6, pp. 110–117, Jan. 2017.

[2] J. Ahn, Y. Yong Kim, and R. Y. Kim, “Delay oriented VR mode WLAN for efficient wireless multi-user virtual reality device,” in Proc. of IEEE International Conference on Consumer Electronics, Las Vegas, NV, USA, March 2017.

[3] M. Singh and B. Jung, “High-definition wireless personal area tracking using AC magnetic field for virtual reality,” in Proc. of IEEE Virtual Reality, Los Angeles, California, USA, March 2017.

[4] M. Chen, W. Saad, and C. Yin, “Virtual reality over wireless networks: Quality-of-service model and learning-based resource management,” available online: arxiv.org/abs/1703.04209, Mar. 2017.

[5] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, “Machine learning for wireless networks with artificial intelligence: A tutorial on neural networks,” available online: arxiv.org/abs/1710.02913, Oct. 2017.

[6] J. F. Yang, S. C. Chang, and C. Y. Chen, “Computation reduction for motion search in low rate video coders,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 12, no. 10, pp. 948–951, Oct. 2002.

[7] N. Cressie and C. K. Wikle, Statistics for spatio-temporal data, John Wiley & Sons, 2015.

[8] M. C. Uryan, O. B. Akan, and I. F. Akyildiz, “Spatio-temporal correlation: theory and applications for wireless sensor networks,” Computer Networks, vol. 45, no. 3, pp. 245–259, June 2004.

[9] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, “Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience,” IEEE Journal on Selected Areas on Communications (JSAC), Special Issue on Human-In-The-Loop Mobile Networks, to appear, 2017.

[10] A. E. Abbas, “Constructing multiattribute utility functions for decision analysis,” INFORMS Tutorials in Operations Research, pp. 62–98, Oct. 2010.

[11] M. Chen, W. Saad, C. Yin, and M. Debbah, “Echo state transfer learning for data correlation aware resource allocation in wireless virtual reality,” available online: http://resume.walid-saad.com/pdf/extendedsilomarpaper.pdf, May 2017.

[12] G. Bacci, S. Lasaulce, W. Saad, and L. Sanguinetti, “Game theory for networks: A tutorial on game-theoretic tools for emerging signal processing applications,” IEEE Signal Processing Magazine, vol. 33, no. 1, pp. 94–119, Jan. 2016.