QoS-Aware Power Management with Deep Learning

Junxiu Zhou and Xian Liu
Department of Systems Engineering
University of Arkansas at Little Rock
Little Rock, USA

Yangyang Tao and Shucheng Yu
Department of Electrical and Computer Engineering
Stevens Institute of Technology
New Jersey, USA

Abstract-Network densification is becoming an overwhelming phenomenon in many emerging wireless communication paradigms. Although network densification may promote system metrics like the throughputs, the quality-of-service (QoS) issue needs to be carefully investigated. Commonly, the QoS-aware power management is tightly restricted by the complicated patterns of interferences among multiple active communication devices. Conventional approaches in optimizing the QoS-aware power management problem may either fail to convergence or the overall power rate is unsatisfactory. In this paper, we make an effort to solve the QoS-aware power management problem with the aid of the deep learning (DL) methodology. Recently DL has shone light on wide variety of research fields, such as image processing and natural language processing. It is our intensive interest in exploring the role that DL plays to solve the QoS-aware power management problem. In the presented extensive experimental analysis work, we show that the DL-based method can well match the solution generated by a conventional optimization procedure. It is impressed that the convergence of DL is quite fast. Moreover, the DL-based approach demonstrates the better performance when the conventional method enters the infeasible region.

Keywords—Deep learning, power management, feedforward neural networks, QoS-aware, wireless communications.

I. INTRODUCTION

Network densification is becoming a common situation in small wireless communications [1]. In the networks supporting the *device-to-device* (D2D) transmission, *machine-to-machine* (M2M) transmission, *vehicle-to-vehicle* (V2V) transmission, and provisioned with the *fog computing* (FC) or *mobile edge computing* (MEC) facilities, densification is the overwhelming phenomenon. One of the main advantages of densification is proximal communication that promotes reuse of system bandwidth. However, proximal communication increases the interference between devices. The interference will directly impact the quality-of-service (QoS). Due to the highly variable traffic patterns and irregular topology, obtaining the optimal solution for the QoS-aware power management problem in a limited time period is a big challenge. Conventional approaches are usually inadequate for this situation. Therefore it is imperative to seek some non-conventional approaches to overcome this emerging challenge. One of the promising ways to solve the problem is the *deep learning* (DL) methodology.

Nowadays, DL plays a very active role in *artificial intelligence* (AI). Indeed, DL has made impressive accomplishments in several disciplines, including computer vision [2], remote sensing [3], and wireless communications [4]. In these disciplines, models are composed of multiple processing layers. DL allows these models to learn the representations of data with multiple levels of abstractions [5]. Although the concerned datasets usually have complex structures and strong inner correlations, DL has shown its strength to effectively extract the high-level features while minimizing the human-interventions. A comprehensive investigation of DL can be found in [6].

In this paper, we apply the DL methodology to solve the QoS-aware power management problem, because of its efficiency in mapping inputs to outputs. The basic idea is to train a deep *neural network* (NN) that knows how to obtain the optimal power management results by automatically learning the potential relationship between the input variables. This deep NN will act as the representative of the conventional iterative-based numerical optimization method but with lower computation cost.

The rest of this paper is organized as follows. In Section II, the related works of this paper is introduced. In Section III, the power management system model is described. Then, in Section IV, the basic concept of DL is overviewed. Next, in Section V, the numerical experiments are conducted and experimental results are reported and discussed. Finally, the conclusion is put in Section VI.

II. RELATED WORK

Currently, there is an ever increasing interest in boosting the intelligence functions of 5G wireless systems by means of DL [7]. However, the applications of DL to network control still remain underexplored [8]. One of the fundamental issues in network control is how to manage the system resources. There were a few reported studies to address this issue.

The work in [9] outlined the application of machine learning techniques in radio resources management. They concluded that the DL-based framework would benefit the resource allocation strategy in improving the behavior of the network. Xu *et al.* [10] proposed to make use of deep reinforcement learning on solving power-efficient resource allocation problem in *cloud radio access networks* (C-RANs). They formulated the current states of all *remote radio heads* (RRHs) in the state space, the demands of associated users in the action space, and the transition power consumption as the reward function. Based on these formulations, the resource allocation problem was solved as a convex optimization problem. However, they focused on approximating the action-value function of the RANs. Zappone *et al.* [11] proposed to use deep artificial NNs to solve a global energy efficiency maximization power allocation problem in a generic interference-limited network. They tried to use DL to learn a fractional programming algorithm.

978-3-903176-15-7 © 2019 IFIP

In [12], the author proposed to use a DL model to approximate an iterative-based power management algorithm called WMMSE [13]. After conducting numerical experiments, they showed that the pre-trained NN model achieve high approximation accuracy with some fixed computational complexity. They concluded that the DL-based approach would have a great potential for real-time wireless resource management problems. The work in [12] is most relevant to our study. However, in this paper, we solve a generic power management problem with the QoS constraints. The impact of interference is also taken into account. In addition, we also analyze the performance of the DL-based approach when iterative-based algorithm fails to find an optimal solution.

The main contributions of this work are twofold. First, we formulate the QoS-aware power management problem. To solve this problem, we customize a *feedforward neural network* (FNN) based on the DL concept. In the training stage, this FNN first learns to map the input to the output through extensive training samples. In a fixed amount of time, our approach achieves a satisfactory approximation of the solution obtained from a conventional optimization procedure. In the testing stage, the established FNN is tested through a set of new samples generated with different QoS requirements. Secondly, we investigate the performance of FNN. Specifically, we find that the FNN approach is robust in the situations where the conventional optimization procedure fails.

III. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we first describe the system model for a generic network under interference, and then propose the optimization objective formula for our power management problem.

A. System Model

We consider a wireless network deployed in a region represented by a disc with radius R_c. This generic network consists of N pairs of *transmitters* (Tx) and *receivers* (Rx). Here, we follow the same number of transmitters and receivers provisioning [12], which is one of the most popular provisions of the D2D communications. We use $\mathcal{C}_T = \{1,2,\ldots,N\}$ and $\mathcal{C}_R = \{1,2,\ldots,N\}$ to denote the index sets of *transmitters* (Tx) and *receivers* (Rx), respectively.

The wireless network is illustrated in Fig. 1, where the solid lines represent the desired transmission links, while the dotted lines represent the interfering links. The main notations are listed in Table I. Other notations will be defined in the relevant context. In Table I, the term "link *ij*" means the link from Tx i to Rx j. The term "channel gain" refers to the small-scale fading. The Rayleigh fading is adopted in the analysis throughout this paper. Thus the channel gains h_{kj} and h_{kk} follow the exponential distribution. The conventions commonly used in the literature of wireless communications are also adopted, such as unit mean and the i.i.d. condition. This way, the large-scale fading due to the path-loss will be described with a power-law term. To concentrate on the key concept, the shadowing effect is included in the large-scale fading with appropriately adjusted parameters.

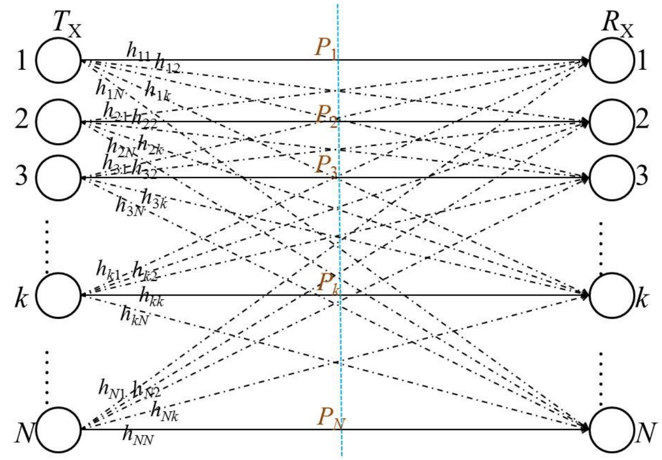

Fig. 1. Generic wireless network structure.

In the present model, half-duplex is assumed, i.e., a node cannot receive signals while simultaneously transmit signals. For example, at a particular moment, the link from Tx 3 to Rx 7 is different than the link from Tx 7 to Rx 3. This implies that, in general, it is not necessarily to have $b_{jk} = b_{kj}$, $b \in \{h, r, \alpha\}$

Also, $r_{kk} \neq 0$, since it represents the distance from Tx k to Rx k. σ_k^2 characterizes the *additive white Gaussian noise* (AWGN). With these elaborations, the *signal-to-interference-plus-noise ratio* (SINR) for the receiver k is expressed as:

$$u_k \triangleq \frac{h_{kk}\left(P_k/r_{kk}^{\alpha_{kk}}\right)}{\sigma_k^2+\sum_{j=1,j\neq k}^{N} h_{jk}\left(P_j/r_{jk}^{\alpha_{jk}}\right)} \quad (1)$$

where usually $1.6 < \alpha_{jk} < 9$. Note that in (1) h_{jk}, h_{kk}, r_{jk}, and r_{kk} are *random variables* (RVs), while the variables P_j and P_k are the entities to be optimized (referred to as the *decision variables* in optimization literature).

TABLE I. THE LIST OF MAIN NOTATIONS

h_{jk}	Channel gain of the interference link *jk*
h_{kk}	Channel gain of the desirable link paired with Tx k and Rx k
N	Number of Tx-Rx pairs
r_{jk}	The length of link *jk* (meter)
u_k	SINR of Rx k (dB)
$u_{k,\min}$	QoS threshold of SINR (dB)
P_k	Transmitter power of Tx k (dBm)
α_{jk}	Path-loss exponent of link *jk*
σ_k^2	Noise power of Rx k

B. Problem Formulation

In the present work, we develop a QoS-aware scheme to optimally manage the power for all transmitters. There are several ways to characterize the merit of scheme. One of the most popular ones is the *weighted sum-rate* (WSR) since it clearly describes the overall throughput of the system. Theoretically, there exist some equivalences to other metrics. Examples can be found in the literature, e.g., [14, Ch.6]. The basic WSR problem was investigated in [12, 13]. Here we adopt WSR as the objective function. Moreover, we introduce the QoS constraints as an enhancement. The augmented WSR problem is formulated as follows:

$$maximize\ \sum_{k=1}^{N} w_k \log_2(1+u_k) \tag{2}$$

$$s.t. \quad \begin{cases} u_{k,min} \le u_k \\ 0 \le P_k \le P_{k,max} \end{cases} \tag{3}$$

$$k = 1,2,\dots,N$$

where u_k is defined in (1). w_k denotes the bandwidth. $P_{k,max}$ refers to the allowed maximum power of k.

IV. Overview of Neural Network Model

In the preceding section, we described the system model and formulated the power management problem. In the literature, the conventional solving schemes for this kind of problems were based on the iterative procedures (see [13] and the references therein). Commonly, the iteration-based procedures consume considerable computational resources. Sometimes these procedures may stop at an infeasible point if the initial point was not carefully chosen. The NN-based methodology provides a different approach, where the main process is treated as an end-to-end black box.

The theoretical basis of the NN approach is due to the universal approximation theorem (UAT) [15], a landmark result in the NN discipline. The UAT indicates that, for any function with some basic measurable characteristics (such as Borel measurability), an FNN with a single layer is able to represent that function. Within this single layer, however, the number of neurons can be very large. As a result, the FNN with a single layer may fail to learn the targeted system. This is where the DL approach begins to plunge into. DL wants to reduce the number of neurons required to represent the desired function and can reduce the amount of generalization error. In the present work, we follow this approach. We will develop an FNN with multiple hidden layers to train the system to represent function.

One of the main uses of NN is to optimize a performance index, e.g. minimizing a loss function. Typically the loss function is defined in terms the *mean square error* (MSE) between the output and the target. In our model, this is between the predicted power resource, i.e., the output of the NN model $f(\boldsymbol{x};\ \boldsymbol{w})$, and the optimal power P,

$$f^* = \arg\min_{f} \|P - f(\boldsymbol{x};\ \boldsymbol{w})\|^2 + \alpha\|\boldsymbol{w}^T\boldsymbol{w}\|^2 \tag{4}$$

where the first term is the MSE, while the second term refers to the *regularization penalty* added to avoid overfitting. The notation $\boldsymbol{w}$ collectively represents all the weight parameters of the NN, and α is a *hyperparameter* that balances the relative contribution of the penalty term. The DL architecture of the present work is illustrated in Fig. 2. It is a fully connected NN with one input layer, multiple hidden layers, and one output layer. We assume the network contains *L-1* hidden layers and one output layer.

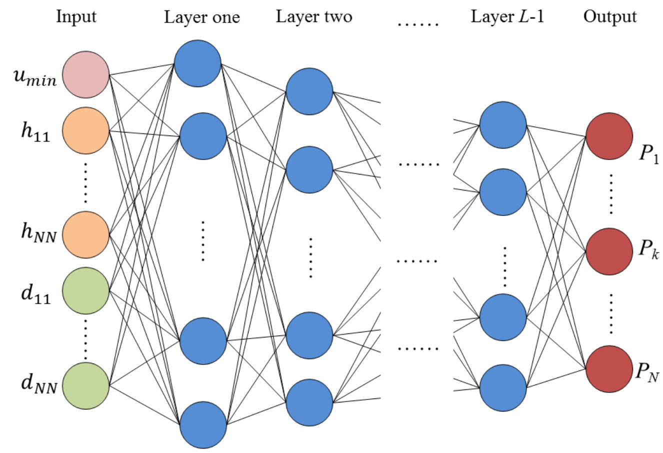

Fig. 2. The neural network architecture.

In this paper, the designed FNN uses MSE to update the overall parameters. However, the final training and testing performance evaluation are computed based on WSR in eq. (2). This is because the augmented WSR problem can reflect the performance over all interference-based links. In fact, the value of the augmented WSR directly depends on the power management results *P*s. They play the same role in evaluation the proposed method's performance. However, in order to fit the constraints of *P*s in eq. (3), we set the following for the output as mentioned in [12]:

$$\tilde{p}_k = \min(\max(p_k, P_{k,\mathrm{min}qos}), P_{k,max}) \tag{5}$$

Note, $P_{k,\mathrm{min}qos}$ represents the minimal power required to satisfy related QoS-constraint.

As shown in Fig. 2, the input to the FNN model is composed of the channel gain and the length of links, because these two sets of variables together determine the wireless network traffic pattern and topology as well as resource management decision. In our case, the augmented WSR problem is also under the constraint of different QoS constraints. In order to make the trained FNN robust under multiple QoS constraint values, we take the QoS constraint value as an additional input variable. Therefore, the total input vector x has a $(2\times N^2+N)$ size. In our numerical experiment, we set all the $u_{k,min}, k = 1,2,\dots,N$ value the same. Therefore the input has $(2\times N^2+1)$ dimension. In this way, the FNN can explore both the required augmented WSR value and the QoS constraints. Then we normalize and concatenate these three variables together to form the input to our NN model

Internally, the hidden layer i accepts a vector of inputs $\boldsymbol{h}$ which is the output of its previous layer. Then it computes an affine transformation $\boldsymbol{z_i} = \boldsymbol{W_{(i-1)}}^T\boldsymbol{h} + \boldsymbol{b}$ and applies an element-wise nonlinear activation function. In this work, for hidden layers we choose the *hyperbolic tangent activation function* $g(\boldsymbol{z}_i) = \tanh(\boldsymbol{z}_i)$ as the activation function. The reason is that the output of our NN model is constrained by eq.

(3) and only the values satisfied those conditions are meaningful.

In this NN, the linear output units aim to produce a vector $\tilde{P} = \boldsymbol{W}_{L-1}{}^{T}\boldsymbol{h} + \boldsymbol{b}$ which is in turn used to produce the mean of a conditional Gaussian distribution, where $\boldsymbol{W}$ is the weight matrix and $\boldsymbol{b}$ is the bias term. Because the linear units do not saturate, they do not degenerate the gradient-based subroutine and are appropriate for a wide variety of optimization algorithms.

Our basic attempt is seeking to produce approximate power management results for input wireless network structure drawn from the same distribution as our training samples. In this work, with learning we are carefully focused on the solution of a restricted problem of interest, rather than the general problem. However, we make an effort to produce the approximate power management for some possible input vectors, especially for those where the conventional methods may fail.

V. EXPERIMENTAL RESULTS

In this section, we verify the presented approach FNN with extensive numerical experiments.

To generate the training data, we employ a conventional optimization procedure. In this work, the Fmincon in Matlab is adopted. Fmincon is a well-known solver for the constrained *nonlinear programming problems* (NLP). In this work, Fmincon is used as a representative of conventional iterative-based optimization methods, because it has all the features of its kind. That is, for large-scale problems, Fmincon may need a considerable amount of computational time. For the QoS-aware WSR problems, Fmincon inclines to stick on infeasible points when the QoS constraints are stringent.

All of the numerical experiments are implemented in MATLAB R2017a on a computer with the 64-bit operating system and 12GB RAM. The FNN is trained in the MATLAB neural network toolbox with GeForce GTX 1080 Ti GPU.

In the numerical experiments, we set the QoS thresholds to 1dB, 3dB, 5dB, 6dB and 7dB with N = 10 pairs of transceivers. The following parameters are adopted in the model: radius = 500 meters, Pmax = 21dBm, AWGN power = -143.97dBm. The layer sizes of the FNN are set to [200, 80, 80, 10]. (Note, although the number of layers in this paper is similar to that in [12], the activation functions of the NN and the structure of the NN layers are different). We use 90% of the training dataset as the training samples, while 10% as the validation samples. We use *scaled conjugate gradient descent* (SCGD) backpropagation method to update the FNN's parameters. The learning rate value is chosen as 0.01. For visualization purpose, the experimental performance is measured in terms of the *cumulative distribution function* (CDF). The CDF is computed based on the optimized power values of WSR in eq. (2).

A. DL Generalization performance

In the NN discipline, one of the challenging tasks is referred to as *generalization*. This notion is due to a fundamental axiom in NN. A good NN is capable of performing well on previously unseen inputs, not only those on which the NN was trained. This capability is called generalization. The implement ability of generalization is based on the statistics of the data. Both the observed data and the unseen data follow the identical probability distribution. In the current WAR power management problem, this is the situation in which the observed data and the unseen data of the random channel gains obey the same probability distributions. This condition is also applicable to the random link lengths. In this experiment, we explore this generalization ability of NN to address the optimal failure drawback of conventional iterative-based methods.

For the QoS-aware WAR model, we use Fmincon to locate the optimal solution. Then the obtained solutions are used as samples to train the FNN. The established FNN treats the mapping procedure as an end-to-end black box. Accordingly, we only need to train to get general values of this FNN's weights and hyperparameter. In the test stage, the whole process becomes a series of matrix manipulations, without complex parameter setting. However, in some scenarios, Fmincon exits with a premature solution or converged to an infeasible solution. These results are possibly caused by the ill initial point or the inappropriate system parameter. Therefore, in the case of Fmincon failures, we want to see if the DL-based method still performs well with respect to the WSR.

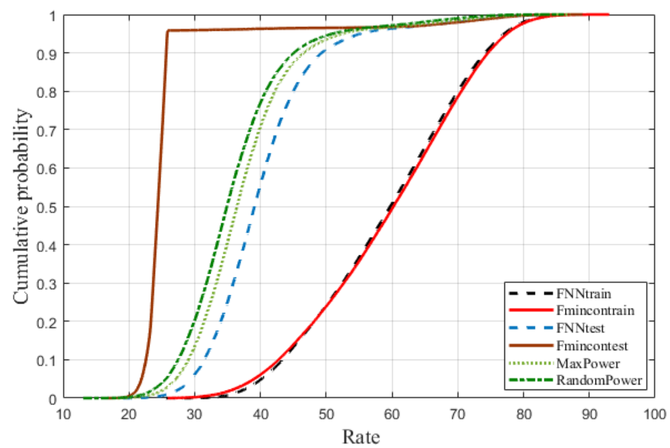

Fig. 3. Fmincon failure case test results when QoS at 7dB.

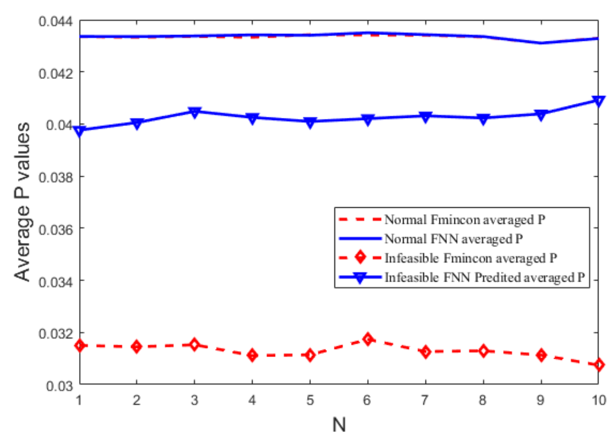

Fig. 4. Comparison the averaged power P under normal and infeasible cases.

In this experiment, we fix the QoS threshold at 7 dB with $N = 10$ pairs of transceivers. Fmincon is used to generate 100,000 samples under the normal optimal solution case as the training data set. In this case, the QoS constraint is not considered as input, therefore the inputs are N^2 dimensional vectors. Later 26367 Fmincon failure samples are obtained under the infeasible case. Only optimal solution samples are used during training. In the testing stage, we use these failure samples as input of the trained FNN to predict the power values in computing the WSR result. The test results are shown in Fig. 3 and Fig. 4.

In Fig. 3, the horizontal axis is the allowable domain for CDF, while the vertical axis is normalized from 0 to 1. As observed from Fig. 3, the predicted CDF of the WSR from trained FNN, i.e. FNNtrain, is very close to the CDF of the optimal solution generated by Fmincon, i.e. Fmincontrain, with an approximate performance of 99.69%. The Fmincon failure samples, Fmincontest, got the worst results. However, with the same input of these failure samples, the trained FNN achieves better power management performance, i.e. FNNtest. It is even better than MaxPower and RandomPower.

Fig. 4 illustrated the average values of power resources P for different links N. Similar as Fig. 3, the trained FNN achieves a similar averaged power values as the Fmincon under the optimal solution samples, indicating that the trained FNN can fit the optimal power values well. In the failure sample cases of Fmincon, that is, for some inputs, the Fmincon cannot find the optimal power management solution. The averaged power values of Fmincon, i.e. Infeasible Fmincon averaged P, are much lower than their optimal cases, i.e. Normal Fmincon averaged P. However, the trained FNN provides a better solution, i.e. Infeasible FNN Predicted averaged P.

In summary, the results presented in Figs. 3 and 4 illuminate the robust performance of FNN in approaching the optimal power management solution. Therefore, we concluded that optimal samples trained FNN allows robust predict of the power values. It is expected to obtain a high success ratio of the WSR.

B. Approximation Performance

To verify the performance of the approximate optimized solution for FNN, we conduct experiments under all five QoS constraints. In this work, we feed all the five data sets to train one FNN. Note only the training process conducted with the help of the GPU. For each QoS, we generate 100,000 training samples composed of Fmincon's optimal solutions. Therefore each training set is a 201×100,000 size matrix. Specifically, each column contains 100-dimensional channel gain values, 100-dimensonal the length of links values and 1-dimensional QoS constraint value. Test data is a matrix of 10×10,000 size, with each column represents 10-dimensional power values. The final performance is measured and reported with respect to the WSR in eq. (2). The CDF describing the WSR obtained by different algorithms is shown in Fig. 5, Fig. 6, and Table II. Fig. 6 is one example extracted from Fig. 5, in order to provide clear illustration. As can be observed, the FNN based results are very close to the Fmincon's optimal solutions. FNN obtains better results than MaxP and RandP under all 5 QoS constraints. Table II shows the numerical results to illustrate the good approximation performance of the proposed FNN.

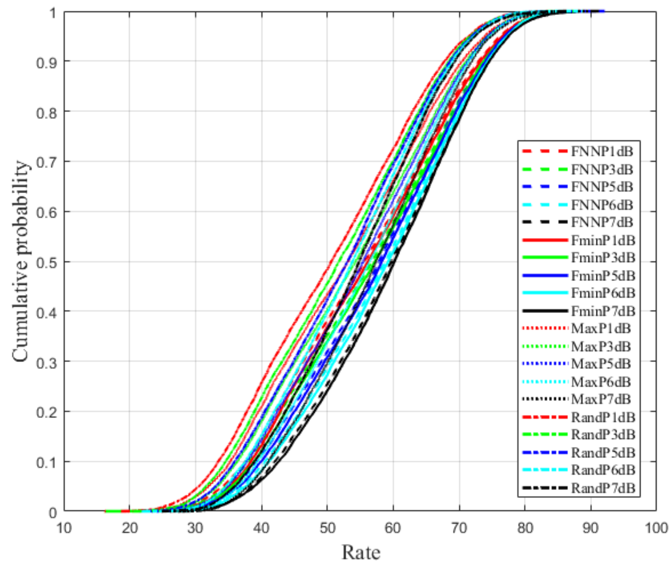

Fig. 5. Testing performance with respect to CDF.

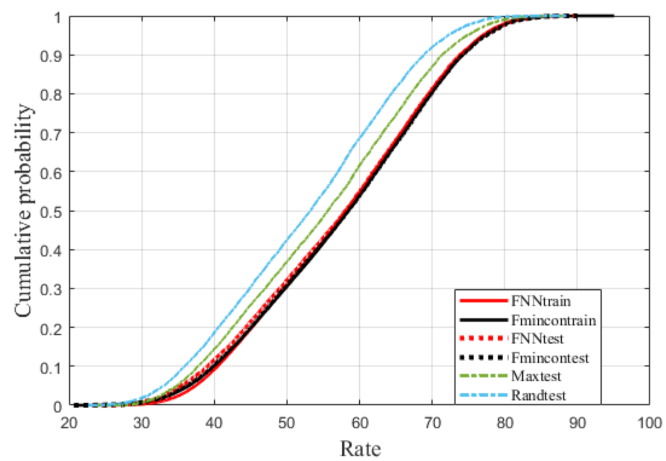

Fig. 6. Testing performance with respect to CDF under 5dB.

TABLE II. COMPARISION OF SUM-RATE ACCURACY

	1dB	3dB	5dB	6dB	7dB
FNN	98.99%	99.12%	99.20%	99.26%	99.30%
MaxP	94.62%	95.14%	95.53%	95.79%	95.97%
RandP	90.60%	91.12%	91.45%	91.66%	91.79%

In this work, the designed FNN need to not only learn to optimal the augmented WSR power management problem but also need to learn to satisfy the QoS constraint. In order to test the FNN's performance on QoS constraint, we list the QoS satisfaction rate in Table III. As illustrated in Table III, the QoS satisfaction rates of FNN are much higher under different QoS constraints. The QoS satisfaction rates of MaxP and RandP decrease as the dB increases. However, FNN can still maintain a higher level of satisfaction rates. Therefore, FNN can not only perform a good approximation of the WSR, but also perform a good approximation of the QoS constraints.

TABLE III. COMPARISION OF QOS SATISFACTION RATE

	1dB	3dB	5dB	6dB	7dB
FNN	99.61%	98.60%	97.35%	95.85%	95.28%
MaxP	90.75%	80.74%	75.36%	71.27%	68.98%
RandP	67.28%	53.56%	44.83%	40.06%	36.58%

TABLE IV. COMPARISION OF COMPUTATIONAL TIME (S)

	QoS	1dB	3dB	5dB	6dB	7dB
Fmincon	Avg.	0.17	0.23	0.31	0.38	0.75
	Min	0.05	0.04	0.03	0.04	0.05
	Max	2.41	3.01	3.80	3.91	4.60
FNN	Avg.	8.60e-06	7.96e-06	7.97e-06	8.25e-06	8.49e-06

A comparison of the computation times for Fmincon and FNN is given in Table IV. It shows that the average minimum time that Fmincon uses to solve the augmented WSR power management problem is about 0.042s. However, this is still far slower than the pre-trained FNN model 8.25e-06s. Note that we do not list the FNN minimal and maximal computational times because there are very close to each other and similar to their average time costs. In addition, the computational cost increases as QoS-constraint value increases in the Fmincon case. Moreover, the minimum and maximum computational costs also vary widely for Fmincon, from approximately 2s to 4s. These computational features of Fmincon, as well as similar conventional optimal methods, make power management not well applied to robust real-time systems. However, as indicated in Table IV, this can be solved by the DL based FNN method, as long as enough training samples are collected to train the DL method.

VI. CONCLUSION

In this paper, we address the QoS-aware power management problem with the NN technology. Through extensively numerical experiments: first, we have shown that the proposed FNN method has the ability to approximation the optimal solution. In this way, it achieves better performance when traditional iterative-based method converges to the infeasible solution. Second, we verified that the NN-based method achieves good approximation ability under different QoS-constraints. In addition, we have shown that the FNN method leads to dramatic reduction in the computational costs. These conclusions make robust real-time power management possible in real life.

REFERENCES

[1] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. Sukhavasi, C. Patel, and S. Geirhofer, “Network densification: the dominant theme for wireless evolution into 5G, ” IEEE Communications Magazine, vol. 52, no. 2, pp. 82-89, 2014.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems, pp. 1097-1105, 2012.

[3] L. Zhang, L. Zhang, and B. Du, “Deep learning for remote sensing data: A technical tutorial on the state of the art,” IEEE Geoscience and Remote Sensing Magazine, vol. 4, no. 2, pp. 22-40, 2016.

[4] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: a survey," arXiv preprint arXiv:1803.04311, 2018.

[5] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436-444, May 2015.

[6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, Cambridge: MIT Press, November 2016.

[7] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, advances and opportunities," IEEE Network, vol. 32, no. 2, pp. 92-99, 2018.

[8] W. G. Hatcher, and W. Yu, "A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends," IEEE Access, vol. 6, pp. 24411-24432, 2018.

[9] F. D. Calabrese, L. Wang, E. Ghadimi, G. Peters, and P. Soldati, "Learning radio resource management in 5G networks: Framework, opportunities and challenges," IEEE Communications Magazine, vol. 56, no.9, pp. 138-145, 2018.

[10] Z. Xu, Y. Wang, J. Tang, J. Wang, and M. C. Gursoy, "A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs," 2017 IEEE International Conference on Communications (ICC), pp. 1-6, 2017.

[11] A. Zappone, M. Debbah, and Z. Altman, "Online energy-efficient power control in wireless networks by deep neural networks," IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1-5, 2018.

[12] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for wireless resource management," 2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1-6, 2017.

[13] Q. Shi, M. Razaviyayn, Z. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3060-3063, 2011.

[14] S. Verdu, Multiuser Detection, Cambridge University Press, Cambridge, UK, 1998.

[15] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” Neural networks, vol. 2, no. 5, pp.359-366, 1989.