Efficient Minimum-Energy Scheduling with Machine-Learning based Predictions for Multiuser MISO Systems

Lei Lei[1], Thang X. Vu[1], Lei You[2], Scott Fowler[3], and Di Yuan[2]

[1]Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Luxembourg
[2]Department of Information Technology, Uppsala University, Sweden
[3]Department of Science and Technology, Linköping University, Sweden
Emails: {lei.lei;thang.vu@uni.lu, lei.you; di.yuan@it.uu.se, scott.fowler@liu.se}

***Abstract*—We address an energy-efficient scheduling problem for practical multiple-input single-output (MISO) systems with stringent execution-time requirements. Optimal user-group scheduling is adopted to enable timely and energy-efficient data transmission, such that all the users' demand can be delivered within a limited time. The high computational complexity in optimal iterative algorithms limits their applications in real-time network operations. In this paper, we rethink the conventional optimization algorithms, and embed machine-learning based predictions in the optimization process, aiming at improving the computational efficiency and meeting the stringent execution-time limits in practice, while retaining competitive energy-saving performance for the MISO system. Numerical results demonstrate that the proposed method, i.e., optimization with machine-learning predictions (OMLP), is able to provide a time-efficient and high-quality solution for the considered scheduling problem. Towards online scheduling in real-time communications, OMLP is of high computational efficiency compared to conventional optimal iterative algorithms. OMLP guarantees the optimality as long as the machine-learning based predictions are accurate.**

***Index Terms*—Energy minimization, optimization, machine learning, MISO, resource scheduling.**

I. Introduction

According to Cisco's annual visual network index reports, numerous users in the system will be increasingly hungry for large amount of data and highly demanding on quality-and-timely services [1]. Reducing the duration in data queuing, data transmission, and terminals' waiting time is important for improving the overall quality-of-experience (QoE) of the users [2]. To fully cater to these service requirements, the next-generation communication system is expected to support low-latency services, users' high data demand, and low energy consumption [2].

Advanced resource optimization is crucial to enable these high-performance requirements. On one hand, performing sophisticated optimal resource optimization can largely boost the system performance. One the other hand, the high computational complexity in resource allocation and the stringent execution-time constraints in real-time networks, may result in several issues in applying optimal algorithmic solutions in practical systems. Many resource optimization problems in wireless networks are combinatorial optimization problems with high computational complexity, e.g., [3]–[5]. Their optimal/suboptimal solutions may not be practical for real-time systems as immense computational capabilities and time can be demanded, especially for the large-scale instances [6]. However, the time limit to complete a decision-making process in online optimization, could be required to execute within seconds or milliseconds [2].

In general, once an optimization problem is proved to be hard to solve, it is difficult to expect that a heuristic solution can meanwhile achieve satisfactory performance and with very low complexity to support online optimization [5]. In most cases, a scheduler may have to make a trade-off between the algorithm's computational complexity and the solution quality, which is a dilemma in the conventional algorithm development [7]. Thus, being aware of the shortcomings of conventional optimization approaches, we are motivated to explore new avenues in solution development for enabling real-time resource optimization. Machine learning is promising to provide a powerful alternative to design optimization algorithms for complex and highly dynamic systems. As an approach in the toolset for wireless network optimization, it has received considerable research attention recently [8]–[10]. In [8], the authors apply support vector machine approaches to optimize the transmit antenna selection. In [9], the authors considered deep reinforcement learning in resource management. In [10], deep-learning based approaches were applied in cache-enabled heterogeneous networks.

In this paper, we explore the benefits of applying machine-learning based predictions in optimal minimum-energy scheduling. We address an emerging issue for applying optimal scheduling solution in real-time systems. We investigate how to cope with the high complexity in resource optimization while retaining fast execution in real-time applications. In solution development, we leverage the power of machine learning and optimal optimization algorithms, aiming at devising competitive solutions for efficient scheduling. In the proposed method, i.e., optimization with machine-learning predictions (OMLP), we embed the machine-learning approach into the developed algorithmic solution, in order to accelerate the optimization process for dealing with the

978-1-5386-3180-5/18/$31.00

scheduling problem. The developed OMLP provides a viable path to apply complicated algorithms to practical real-time systems. The proposed OMLP method demonstrates promising performance in improving computational efficiency and in approximating global optimality. In addition, it provides a new dimension to deal with the trade-off between the solution quality and the computational complexity.

The rest of the paper is organized as follows. Section II presents the system models for downlink multi-antenna systems, group scheduling, and transmission duration. Section III formulates an energy-efficient scheduling problem with quality of service (QoS) constraints. Section IV analyzes the problem's tractability and proposes OMLP method. Numerical results are demonstrated in Section V. Conclusions are given in Section VI.

Throughout this paper, we use the following notations. Normal letters represent scalar quantities, bold face letters indicate vectors, and boldface uppercase letters designate matrices. The superscript $(.)^H$ stands for Hermitian transpose and the two-norm of a vector is denoted by $\|.\|$.

II. System Model

We consider a downlink multiple-input single-output (MISO) cellular system in which single-antenna users request data D_k from a base station (BS). The user set is denoted as $\mathcal{K} = \{1, \dots, k, \dots, K\}$. In a schedule, the BS with L antennas serves its associated users' demand $D_1, \dots, D_K$ in a common frequency channel with bandwidth B. The BS is required to satisfy all users' data demand within time limit T_{tot}. The wireless transmissions are subject to block Rayleigh fading channels, in which the channel fading coefficients are fixed within a block and are mutual independent across the users. The block duration is assumed to be long enough to complete a scheduling period.

A. Scheduling Scheme

Towards efficient data transmission, we adopt dynamic user-group scheduling and optimize the precoding vector in each group to mitigate co-channel interference. We refer to a group g as a user cluster/set, consisting of one or multiple users [11]. Let $\mathcal{K}_g$ be the set of all the included users in group g. Once a group g is scheduled, the BS's L antennas will transmit data to the users in $\mathcal{K}_g$ with positive rates and last for a certain duration t_g. Enumerating all the combinations of the user groups, provides 2^K-1 possible candidates in total. The union of all the groups is denoted by $\mathcal{G} = \{1, \dots, g, \dots, G\}$, where $G = 2^K - 1$. For example, if $K = 3$ there are seven candidate groups in total $\{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}$.

In scheduling, these user groups can be selectively scheduled in a sequential manner to deliver users' data demand, and each user's demand can be flexibly delivered in multiple groups with non-uniform data rates. The cardinality of $\mathcal{G}$ increases exponentially with K, thus G can be huge when K becomes large. To reduce the complexity, some simple schemes are adopted in previous works, e.g., time division multiple access (TDMA) which can be simply enabled by setting $G = K$ and $|\mathcal{K}_g| = 1$ in group enumeration. However, the simple scheduling schemes, e.g., TDMA, may fail to satisfy all users' QoS within T_{tot}. Thus in this work, the selection of scheduled user groups is not predefined, but is subject to optimization.

B. Transmission Model in Scheduling

Let $\mathbf{h}_k \in \mathbb{C}^{L\times 1}$ denote the channel vector from the BS's antennas to user k, which follows circular-symmetric complex Gaussian distribution $\mathbf{h}_k \sim \mathcal{CN}(\mathbf{0}, \sigma^2_{h_k}\mathbf{I}_L)$, where $\sigma^2_{h_k}$ is the parameter accounting for the path loss from the BS antennas to user k. For scheduling a group g, denote x_k^g as the modulated signal for user k in group g. In order to transmit data to the users, the BS precodes data first and then broadcasts to the users. Denote $\mathbf{w}_k^g \in \mathbb{C}^{L\times 1}$ as the precoding vector for user k in group g. The received signal at user k in group g is given as

$$y_k^g = \mathbf{h}_k^H \mathbf{w}_k^g x_k^g + \sum_{i\in\mathcal{K}_g\setminus\{k\}} \mathbf{h}_k^H \mathbf{w}_i^g x_i^g + n_k, \qquad (1)$$

where n_k is Gaussian noise with zero mean and variance σ^2. The first term in (1) is the desired signal, and the second term is the inter-user interference. The signal-to-interference-plus-noise ratio at user k is given as

$$\text{SINR}_k^g = \frac{|\mathbf{h}_k^H \mathbf{w}_k^g|^2}{\sum\limits_{i\in\mathcal{K}_g\setminus\{k\}} |\mathbf{h}_k^H \mathbf{w}_i^g|^2 + \sigma^2}.$$

The achievable data rate of user k in group g is

$$R_k^g = B\log_2\left(1 + \text{SINR}_k^g\right), k \in \mathcal{K}. \qquad (2)$$

The data transmission at the BS is considered being continuous in a schedule. This is supported by the multiple stop-and-wait processes in LTE systems [12]. That is, once the first data packet is sent in a process, and waiting for an acknowledgment from the users, the BS will immediately start another parallel process and use the same channel to send the next data packet, instead of stopping and waiting to send the second packet until receiving the first packet's acknowledgment. Then we can define the total transmission duration as $\tau_{tot} = \sum_{g\in\mathcal{G}} t_g$, i.e., summation of the transmission duration of all the scheduled user groups.

III. Problem Formulation

Motivated by the fact that the BS is expected to empty its queued data as soon as possible or within a limited time interval, such that the occupied time-frequency resources can be released for serving the upcoming demand, then we consider an optimization problem of user-group scheduling in this section, in order to satisfy all the users' demand by minimal energy consumption, and meanwhile the data transmission can be completed within time limit T_{tot}.

We consider minimum mean square error (MMSE) precoding for each group. We collect all the channels vectors $\mathbf{h}_k$ for the users in group g, and form a $|\mathcal{K}_g| \times L$ matrix $\mathbf{H}_g$.

Under MMSE, the beamformer vector for user k in group g is of the form $\mathbf{w}_k^g = \sqrt{p_k}\hat{\mathbf{h}}_k$, where p_k is the transmit power for user k and $\hat{\mathbf{h}}_k$ is the column corresponding to user k in $\mathbf{H}_g^H(\sigma^2\mathbf{I} + \mathbf{H}_g\mathbf{H}_g^H)^{-1}$. Denote $\beta_{k,i}^g = |\mathbf{h}_k^H\hat{\mathbf{h}}_i|^2$, $\forall k, i \in \mathcal{K}_g$, by the interference factor caused to user k from the user i's beamforming vector. The minimum-energy scheduling problem is formulated in P1.

$$\text{P1: } \min_{t_g} \sum_{g\in\mathcal{G}} t_g \sum_{k\in\mathcal{K}_g} \beta_{k,k}^g p_k \tag{3a}$$

$$\text{s.t.} \sum_{g\in\mathcal{G}} t_g B \log_2\Big(1 + \frac{\beta_{k,k}^g p_k}{\sum\limits_{i\in\mathcal{K}_g\backslash\{k\}} \beta_{k,i}^g p_i + \sigma^2}\Big) \geq D_k,\ \forall k \in \mathcal{K} \tag{3b}$$

$$\sum_{g\in\mathcal{G}} t_g \leq T_{tot} \tag{3c}$$

The optimizing variables in P1 are t_g, $\forall g \in \mathcal{G}$. The optimization process determines which user groups will be scheduled to transmit users' demand and for how long. The objective (3a) is to minimize the energy consumption in data transmission, where $\sum_{k\in\mathcal{K}_g} \beta_{k,k}^g p_k$ is the total power in group g. In general, the groups with larger cardinality lead to higher sum power. In constraints (3b), the requested data demand for each user k must be delivered. In (3c), the BS should complete all the data delivery within a limited duration T_{tot}. Note that the signal processing delay is not included in the total transmission time in (3c). This is because the data transmission at the BS and the decoding processing at the receivers are carried out in parallel. When the BS completes all the data transmission in a schedule, though at the receiver side milliseconds delay can be caused due to processing the last few data packets, the term $\sum_{g\in\mathcal{G}} t_g$ typically dominates the time consumption, e.g., seconds over milliseconds.

For better understanding the problem's structure, we characterize the impact of constraint (3c) for P1 below, based on the proofs in previous works [11], [13].

Corollary 1. *Relaxing constraint (3c), TDMA is optimal for P1.*

For minimizing energy in P1, we remark that if TDMA is feasible, it is the optimal scheduling scheme for P1, e.g., when users' demand is low and can be satisfied within T_{tot} by TDMA. In this paper, we are more focused on realistic scenarios with heavier traffic demand and stricter time limits in data transmission. For these cases, TDMA fails to meet the transmission deadline. In addition, if we reduce T_{tot} to T'_{tot}, the optimal energy $E^{*\prime}$ under T'_{tot} is no less than the optimal energy E^* under T_{tot}.

Corollary 2. *If $T'_{tot} < T_{tot}$ in constraint (3c) and P1 is feasible, optimal energy $E^{*\prime} \geq E^*$.*

To meet the stricter time limit T'_{tot}, the BS has to schedule those larger-cardinality groups to allow more users to be concurrently served in data transmission. This typically leads to higher energy consumption in the optimum.

IV. Optimal Scheduling with Machine-Learning based Predictions

In this section, we first characterize the complexity and the difficulty in solving P1, then we propose OMLP method to enable an efficient solution for P1.

A. Complexity and Difficulty

Although P1 is a linear programming (LP) problem which can be solved by standard optimization tools, e.g., the simplex algorithm [14], the linearity cannot simply conclude the problem's tractability. In fact, the computational complexity and the computational time in P1 increase exponentially with K. The hardness of P1 can be referred to the minimum-energy scheduling with single-antenna BSs in [11] and the general transmitter-receiver links scheduling problem in [13]. Once the scheduling has to determine the best way of grouping subsets to transmit data, e.g., determining the optimal subsets of BSs, links, or users, the scheduling problem that includes this combinatorial aspect in general is of high computational complexity [15].

In realistic systems, stringent execution-time is imposed in real-time scheduling. If P1 is required to support online scheduling, then P1 must be solved very efficiently. However, for many-user cases, the optimal iterative algorithms may exhibit their limits in supporting online real-time network optimization. Solving such difficult problems to the satisfactory performance would require a much long span of computing time. Thus, understanding the practical limitations of the conventional approaches, we are motivated to explore new avenues in solution development. In order to accelerate the optimization process, we first investigate the major difficulties in solving P1. The first difficulty is the huge number of groups to be searched in the optimization. To determine the optimal groups, a scheduler has to go through and evaluate all the $2^K - 1$ groups. On the other hand, by the LP theory, in the optimum, the number of scheduled groups is no more than the number of constraints [13], [14]. One can observe that the number of constraints in P1 is linear with K. Then P1's outcome vector $[t_1, \ldots, t_g, \ldots, t_G]$ will be very sparse, since most of the elements/groups in the vector will be zero/inactive. Thus, to accelerate the optimization process, an effective way is to confine the searching space and let the algorithm avoid exploring those never-used groups.

However, the issue is that given a new input to P1, it is not immediately clear to know which groups are not optimal and should be excluded from the $2^K - 1$ candidates. This introduces the second difficulty which is the implicit combinatorial aspect in P1. In fact, for each user, the scheduler has to make binary decisions of determining whether a user should be transmitted in a group or entirely in TDMA. If the key information of the combinatorial part can be known in the optimization process, the computational time will be largely reduced.

B. The Proposed Algorithmic Solution: OMLP

Towards applying optimal algorithms in online scheduling, we then propose OMLP to provide high-quality and time-efficient solution. The idea is to use machine-learning based predictions to help the optimal algorithm to tackle the most difficult and time-consuming part in the optimization.

Several machine learning models can be adopted, e.g., logistic regression (LR), deep neural network (DNN) [16], to establish a prediction system in order to learn the relations between P1's input and the optimal decisions of P1 by observing the designed training set $(\mathbf{X}, \mathbf{y})$. We refer to the matrix of inputs $\mathbf{X}$ consisting of channel coefficients $|h_{lk}|^2$, $\forall l \in \{1, \ldots, L\}$, $\forall k \in \{1, \ldots, K\}$, users' demand $D_1, \ldots, D_K$, and transmission deadline T_{tot}. For generating $\mathbf{y}$, the optimal LP algorithm, e.g., the simplex algorithm, is applied to obtain the optimal solutions. The vector of outputs $\mathbf{y}$ carries the key information extracted from the optimal solutions in P1. Considering the exponential number of groups in the optimization, it prohibits the LR model or DNN simply setting its output as the optimization variables $t_1, \ldots, t_G$ in P1, mainly due to the high complexity in function approximation in learning process and the feasibility issues in P1 [16]. By our design, the output of the machine-learning prediction is to provide two types of key information. One key information is organized in a K-dimension binary vector $\boldsymbol{b} = [b_1, b_2, \ldots, b_K]$. The element b_k indicates whether user k's demand D_k is transmitted alone in the whole scheduling period ($b_k = 0$), i.e., TDMA, or can be delivered in any group/groups ($b_k = 1$). The next type of the key information is stored in another K-dimension binary vector $\hat{\boldsymbol{b}} = [\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_K]$, where $\hat{b}_k$ represents whether any of k-cardinality groups should be scheduled in the optimum ($\hat{b}_k = 1$), or not used at all ($\hat{b}_k = 0$). We organize vectors $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$ into $\mathbf{y}$ with $2K$ elements. With the training process, the weight values in the model can be improved by feeding with large-size of training sets. A maturely trained learning model is expected to provide guidance to confine the searching space of P1 such that the computational time for solving P1 can be significantly reduced for real-time applications, and the competitive performance can be achieved.

The proposed OMLP algorithm is summarized as four steps. In general, the optimization procedure in the algorithm consists of an offline machine-learning model training phase, and an online test phase. **Step 1:** In the training phase, the optimal algorithm, e.g., the simplex algorithm, is adopted to generate the optimal decisions for the training. Feeding the model by sufficient data instances, the model is able to learn the relation between the input and the output [17]. **Step 2:** In the online test phase, given a new input to a well-trained learning model, it can timely provide the required key information, i.e., the output vectors $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$. Note that in some machine-learning model, e.g., DNN, there may exist fractional elements in $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$. We then design a rounding approach to convert those fractional elements to binary. We firstly compute the mean values M and $\hat{M}$ for $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$, respectively. If any fractional $b_k > \alpha M$ or $\hat{b}_k > \alpha \hat{M}$, we set $b_k = 1$ or $\hat{b}_k = 1$, respectively, otherwise zero, where $\alpha > 0$ is a control parameter to balance the computational efficiency and the optimality. **Step 3:** Relying on the binary $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$, we delete a considerably large amount of groups from $\mathcal{G}$, forming a small-scale candidates set $\mathcal{G}^*$ for P1. Specifically, if $b_k = 0$, it means user k will be with high probability to be scheduled by TDMA, then only one group $\{k\}$ is needed for user k. As a results, in OMLP, all the relevant groups containing user k are excluded from the original set $\mathcal{G}$, i.e., $\mathcal{G} \backslash \{g \in \mathcal{G} : k \in \mathcal{K}_g,\ |\mathcal{K}_g| > 1\}$. If $\hat{b}_k = 0$, it means that all the groups with cardinality k will highly possibly not be scheduled, and they can be excluded from $\mathcal{G}$, i.e., $\mathcal{G} \backslash \{g \in \mathcal{G} : |\mathcal{K}_g| = k\}$. By doing so, the proposed OMLP is able to reduce the complexity in the online test phase for solving P1. The cardinality of the restricted set $\mathcal{G}^*$ may not necessarily be an exponential number of K. **Step 4:** We solve a small-scale P1 with the restricted set $\mathcal{G}^*$, which can be much more efficient than a large-scale one.

We remark that as long as the well-trained machine-learning model can provide accurate predictions, the proposed solution grantees global optimality. This is because OMLP in fact equivalently transforms a large-scale optimization task to a small-scale one by precisely excluding non-optimal groups without loss of any optimality. In case the prediction is not completely accurate in $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$, the designed control parameter α can be used to improve the prediction accuracy. For instance, suppose $M = 0.5$, $\alpha = 1$, and accurate b_k should be one in the optimum, however, due to imperfect estimation we read $b_k = 0.49$ from the learning model's output. In OMLP, b_k will be rounded to zero as $b_k < \alpha M$, meaning that any groups involved by user k will be excluded from the optimization process for solving P1, though at least one group among them is clearly optimal. For this case, we can scale down α, say 0.9, then b_k will be set to 1 as $b_k = 0.49 > \alpha M = 0.45$, and all the groups containing user k will be searched in OMLP, and the solution quality can be guaranteed.

Table I
SIMULATION PARAMETERS

Parameter	**Value**
Cell radius	300 m
Carrier frequency	2 GHz
Channel bandwidth, B	1 MHz
Number of users, K	5-25
Path loss	COST-231-HATA
Shadowing (Log-normal)	8 dB standard deviation
Fading	Rayleigh block fading
Noise power spectral density	-173 dBm/Hz
Prediction model	LR, DNN

V. PERFORMANCE EVALUATION

In this section, we first provide numerical results to illustrate the features of the optimal solutions in P1 in order to verify the rationale of the proposed OMLP. Then we show the effectiveness of the OMLP approach, in terms of computational time reduction and the performance gap between OMLP and the optimum. The simulation parameters are summarized in Table I.

A. Characterizations of the Optimal Scheduling Solutions in P1

The computational efficiency of OMLP depends on the number of zero elements in vectors $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$. More zeros elements in the vectors, more groups are with very low probability to be scheduled in the optimum. By the design in OMLP, these groups will not be considered in the optimization, thus it enables high computational efficiency. To verify the idea of OMLP in reducing the candidate groups, in Fig. 1 and Fig. 2, we first demonstrate the characterizations of the scheduled groups by observing the optimal solutions. For illustration, 15-user scenarios are adopted, and in both figures we generate one thousand instances and show the average performance over the normalized time limit T_{tot}. We remark that the variation of T_{tot} should be limited within a time interval $[T_{feas}, T_{tdma}]$ corresponding to the normalized interval $[1, 10]$ in both figures, where T_{feas} is the minimum duration to keep the problem feasible (refer to the left red-dotted line), and T_{tdma} is the maximum transmission duration (refer to the right red-dotted line). If $T_{tot} < T_{feas}$ the problem will be infeasible, and if $T_{tot} \geq T_{tdma}$ TDMA will be optimal for P1.

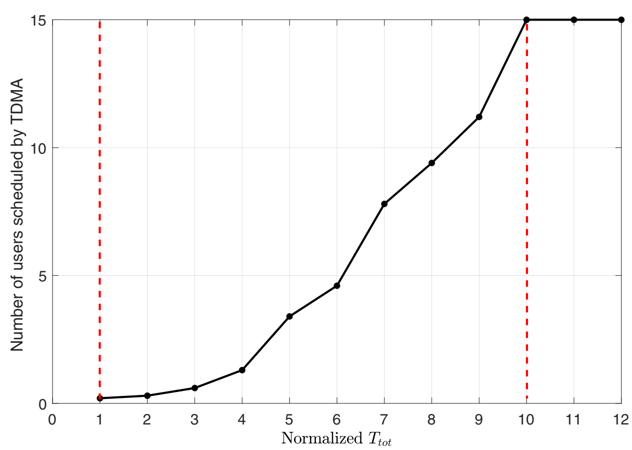

Figure 1. The average number of the users scheduled by TDMA in the optimum.

From the results in Fig. 1, when T_{tot} becomes large, TDMA transmission will be preferred to more users in the optimum. Since the predicted vector $\boldsymbol{b}$ learns this behavior in the training phase, then correspondingly, the more zero elements in $\boldsymbol{b}$ will be predicted when T_{tot} increases, which means a large amount of groups will be excluded in the restricted set $\mathcal{G}^*$. On the other hand, if T_{tot} becomes small, almost all the users in Fig. 1 are scheduled in groups instead of TDMA. In this case, $\boldsymbol{b}$ will have marginal influence in reducing the number of candidate groups. However, from Fig. 2, we observe that in average only very few types of cardinality are actually used in the optimum. Overall, around 1 to 4 cardinality types out of 15 are scheduled. For example, if the three groups $\{1\}, \{1, 2\}, \{1, \ldots, 15\}$ are optimal, then the scheduled groups are with 3 types of cardinality. The accurate prediction should be $\hat{b}_1 = \hat{b}_2 = \hat{b}_{15} = 1$, and the other elements in $\hat{\boldsymbol{b}}$ are zero. As a supplementary mechanism, most of the elements in vector $\hat{\boldsymbol{b}}$ will be predicted as zero. By jointly using the information from both $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$, the size of $\mathcal{G}^*$ can be confined in a moderate level. Thus, OMLP is promising to enable high computational efficiency.

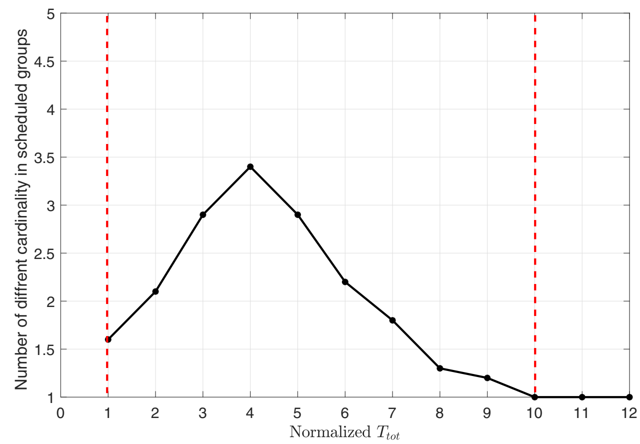

Figure 2. The average number of different cardinality in the optimal groups.

B. The Performance of Computational Time of OMLP

To illustrate the computational efficiency of the proposed OMLP in its online test phase, we compare the CPU time (in seconds) in computations between OMLP and two optimal iterative algorithms, i.e., the simplex algorithm and the column generation algorithm. The former is a conventional algorithm for optimally solving LP, and the latter is proposed to improve the former's computational efficiency with guaranteed optimality. To provide a fair comparison, we implement and evaluate the three algorithms in a unified platform Matlab. All the three algorithms are applied to solve P1, and in OMLP both LR and DNN approaches are adopted in the machine-learning component. The averaged computational time per instance are shown in Table II.

Table II
COMPARISON IN COMPUTATIONAL TIME

Cases	Simplex	Column Generation	OMLP in Test Phase
$K = 5$	0.091	0.087	0.065
$K = 15$	1.357	1.019	0.138
$K = 20$	187.3	121.4	0.291
$K = 25$	>3600	>3600	0.402

From the results, for 5-user to 15-user cases, all the three algorithms can solve P1 very efficiently. For the cases of $K > 15$, the CPU time in OMLP keeps at the same magnitude as before, whereas the time in the other two algorithms exponentially increases with the number of users. As can be foreseen, the computational efficiency of the proposed solution OMLP is insensitive with the increase of the input size.

We remark that the high computational-efficiency of OMLP displayed in Table II, does not mean that we reduce the total computational efforts in solving P1, like the most of heuristic algorithms. By our design, the majority of the complexity in OMLP is not disappeared but just shifted from the online test phase to the offline training phase, such that the computational complexity in the online test phase is moderate thus OMLP can keep high efficiency in computations.

C. Performance of OMLP in Approximating Optimum

Next, in Fig. 3, we show the algorithm OMLP's ability in approaching the optimum with respect to the training progress. We compare the energy obtained in OMLP with the optimal energy to derive the performance gaps. The values in y-axis demonstrate how close of OMLP to the optimum. For instance, "0.9" in Fig. 3 means in average 10% performance gap between the energy in OMLP and the optimum. Based on the completely/relatively accurate information of $\boldsymbol{b}$ and $\hat{\boldsymbol{b}}$, most of the optimal groups can be kept in the restricted set $\mathcal{G}^*$, thus in average OMLP can achieve close-to-optimum performance. By relying on a well-trained machine-learning model, the average energy gaps between OMLP and the optimum are less than 11%. By adopting small α, algorithm OMLP can be even closer to the optimum, but may pay more computational time than the case of $\alpha = 1$.

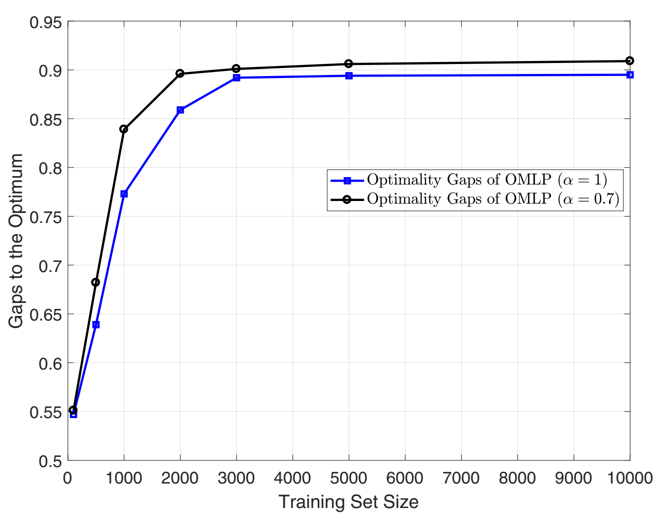

Figure 3. Average gaps between the optimum and OMLP.

VI. Conclusions

We considered applying machine-learning based methods in optimal minimum-energy scheduling. We formulated a resource scheduling problem with various QoS requirements, aiming at providing timely services to satisfy users' demand by consuming less transmission energy. To deal with the practical issues of the high complexity in resource optimization and the stringent execution-time requirements in real-time operations, we developed a solution based on optimization with machine-learning predictions to enable high-quality and time-efficient solutions. Numerical results demonstrate the promising performance of the proposed method in improving computational efficiency and optimality approximation. The computational time is insensitive to the input size. The developed algorithm is able to efficiently provide close-to-optimum solutions when the accuracy of the machine-learning based predictions is improved. The global optimum can be achieved if the predictions are precise.

VII. Acknowledgments

The work has been supported by the European Research Council (ERC) project "Actively Enhanced Cognition based Framework for Design of Complex Systems (AGNOSTIC)". The work of Dr. S. Fowler has been supported by the EU Horizon 2020 MSCA project DECADE (645705). The work of Dr. L. Lei has been supported by the FNR CORE project ROSETTA (11632107).

References

[1] Cisco, "Visual networking index: global mobile data traffic forecast update, 2016-2021," Feb. 2017, white paper at Cisco.com.

[2] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065-1082, June 2014.

[3] L. Lei, D. Yuan, C. K. Ho and S. Sun, "Power and channel allocation for non-orthogonal multiple access in 5G systems: tractability and computation," in *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8580-8594, Dec. 2016.

[4] L. Lei, D. Yuan and P. Värbrand, "On Power Minimization for Non-orthogonal Multiple Access (NOMA)," in *IEEE Communications Letters*, vol. 20, no. 12, pp. 2458-2461, Dec. 2016.

[5] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, 1993.

[6] M. Cardone, D. Tuninetti and R. Knopp, "On user scheduling for maximum throughput in K-user MISO broadcast channels," IEEE International Conference on Communications (ICC), 2015, pp. 4205-4210.

[7] F. Calabrese, L. Wang, E. Ghadimi, G. Peters, P. Soldati, "Learning radio resource management in 5G networks: framework, opportunities and challenges," arXiv, Online available: https://arxiv.org/pdf/1611.10253.pdf, 2017.

[8] J. Joung, "Machine Learning-Based Antenna Selection in Wireless Communications," in IEEE Communications Letters, vol. 20, no. 11, pp. 2241-2244, Nov. 2016.

[9] E. Ghadimi, F. D. Calabrese, G. Peters, and P. Soldati, "A reinforcement learning approach to power control and rate adaptation in cellular networks, IEEE International Conference on Communications (ICC), 2017.

[10] L. Lei, L. You, G. Dai, T. X. Vu, D. Yuan, and S. Chatzinotas, "A deep learning approach for optimizing content delivering in cache-enabled HetNet," IEEE International Symposium on Wireless Communication Systems (ISWCS), 2017.

[11] L. Lei, D. Yuan, C. K. Ho, and S. Sun, "Optimal cell clustering and activation for energy saving in load-coupled wireless networks," in *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6150–6163, Nov. 2015.

[12] S. Sesia, I. Toufik, and M. Baker, *LTE: the UMTS long term evolution*, Wiley, 2011.

[13] V. Angelakis, A. Ephremides, Q. He and D. Yuan, "Minimum-time link scheduling for emptying wireless systems: solution characterization and algorithmic framework," in *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1083-1100, Feb. 2014.

[14] K. Murty, *Linear programming*, Wiley, 1983.

[15] A. Pantelidou and A. Ephremides, "The scheduling problem in wireless networks," Journal of Communications and Networks, vol. 11, no. 5, pp. 489499, Oct. 2009.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[17] V. Sze, Y. Chen, T. Yang, and J. Emer, "Efficient processing of deep neural networks: A tutorial and survey," arXiv, Online available: http://arxiv.org/abs/1703.09039, Mar. 2017.