Scalable User-Substation Assignment with Big Data from Power Grids

Bo Lyu, *Member, IEEE*, Yanhua Li, *Senior Member, IEEE*, Jie Fu, *Member, IEEE*, Andrew C. Trapp, *Member, IEEE*, Haiyong Xie, *Senior Member, IEEE*, Yong Liao, *Member, IEEE*

Abstract—The fast pace of global urbanization is drastically changing the population distributions over the world, which leads to significant changes in geographical population densities. Such changes in turn alter the underlying geographical power demand over time, and drive power substations to become over-supplied (demand $\ll$ capacity) or under-supplied (demand $\approx$ capacity). In this paper, we make the first attempt to investigate the problem of power substation-user assignment by analyzing large-scale power grid data. We develop a **S**calable **P**ower **U**ser **A**ssignment (SPUA) framework, that takes large-scale *spatial power user/substation distribution data* and *temporal user power consumption data* as input, and control the assignments between users and substations, in a manner that minimizes the maximum substation utilization among all substations. To evaluate the performance of our SPUA framework, we conduct evaluations on real power consumption data and user/substation location data collected from Xinjiang Province in China for 35 days in 2015. The evaluation results demonstrate that our SPUA framework can achieve a 20%–65% reduction on the maximum substation utilization, and 2 to 3.7 times reduction on total transmission loss over other baseline methods.

Index Terms—Big data, power grids, ADMM, power substation utilization.

1 Introduction

Electricity has become an indispensable necessity in our daily lives, powering the machines that keep our homes, businesses, schools and hospitals safe, comfortable and convenient. As the fast development of sensors, monitoring devices, such as smart meters, a large amount of power grid data are generated over time, including temporal energy consumption data, spatial user/substation distribution data, and so on. All these heterogeneous data sources offer new research and technological opportunities, and enable intelligent solutions for various applications in power grids [2], [3], [4], [5].

A power grid consists of a network of power plants and power substations that provide electricity power to a wide range of power users. Each power substation has a certain power capacity, that limits the total power demand it can serve; this capacity is typically fixed when the substation was deployed according to the regional power demand. However, the fast pace of global urbanization has dramatically changed the population distributions all over the world. For example, one study [6] reported that in 1950, 30% of the world's population was urban, which increases to 54% in 2014, in 2050 is projected to be 66%. This urbanization leads to significant changes on geographical population densities, thereby altering the underlying geographical power demand over time. For example, from large-scale power consumption data from Urumqi City, China, the rapid expansion of urban population size has driven regional power demand to the capacity limits of the nearby power substations. On the other hand, as the population density changes over time, some power substations in Xinjiang province cover power users that are 300 km away, leading to high transmission losses. We are thus motivated to investigate how to reduce substation power utilization, and prevent them from being overloaded or over-supplied.

Though none of the work in the literature has clearly proposed and addressed the power substation-user assignment problem, in operation research, various assignment problems have been investigated extensively, such as machine job scheduling problem and bin-packing problem [7], [8], [9], [10], [11]. However, these results cannot be directly applied for the power substation-user assignment problem, because of the following reasons: (1) The power grid system has unique challenges and features to be clearly and explicitly characterized as objectives and constraints in the formulation, such as the power transmission loss, power substation capacity, geographical proximity between users and substations; (2) The assignment problem we are facing involves a large amount of 6.3 million users and 783 substations, making it unsolvable even for its related linear programming (LP) relaxation. Hence, how to precisely model power grid characteristics and how to scale up the optimization solution with a provable theoretical error bound are the primary challenges in this study.

In this work, we make the first attempt to investigate the power user assignment problem in large scale power grid. The design goal is to have a scalable solution to assign each power user to one substation, while minimizing the maximum substation utilization. We develop a **S**calable **P**ower **U**ser **A**ssignment (SPUA) framework, which takes the spatial power user/substation distribution, and temporal user power consumption data as input, and performs

- *Bo Lyu, Haiyong Xie, and Yong Liao are with China Academy of Electronics and Information Technology. E-mail: blyu@csdslab.net.*
- *Yanhua Li, Jie Fu, and Andrew C. Trapp are with Worcester Polytechnic Institute (WPI), Worcester, MA 01609. E-mail: {yli15,jfu2,atrapp}@wpi.edu.*
- *This work was primarily done when Bo Lyu was visiting Worcester Polytechnic Institute (WPI).*

An earlier version of this work appeared in the Proceedings of ACM SIGSPATIAL GIS [1], Oct. 30th–Nov. 3rd 2016.

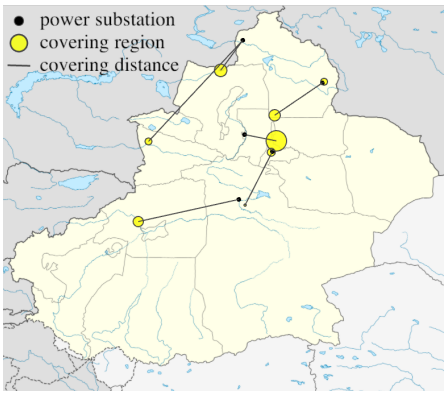


Fig. 1. Long-distance coverage

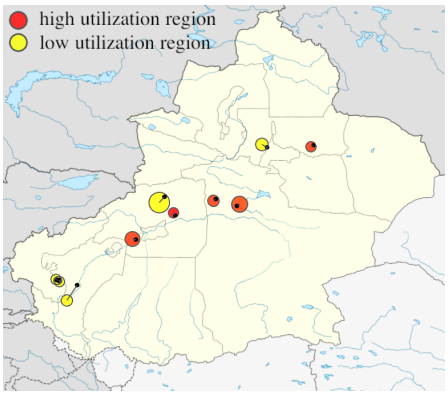


Fig. 2. Under- and over-supplied substations (peak hours)

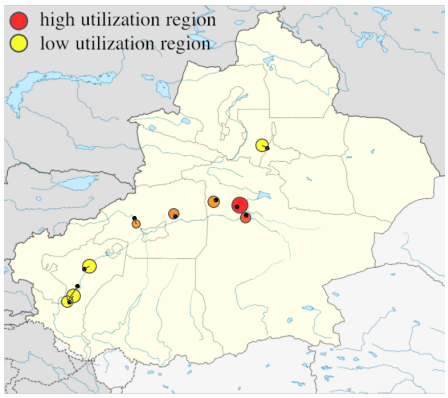


Fig. 3. Under- and over-supplied substations (valley hours)

optimal user assignment to substations to minimize the maximum substation utilization among all substations. Our main contributions are summarized as follows.

- We formulate the power user assignment problem using integer programming, which is NP-hard. We employ a 2-approximation algorithm based on linear programming (LP) relaxation to solve the problem.
- Due to the large-scale size of the power user assignment problem instances we consider, even the relaxed linear programming relaxation is unsolvable using a centralized algorithm. We propose a distributed solution using the block-splitting algorithm [12], by decomposing the large LP problem into small parallelizable sub-problems.
- To evaluate the performance of our SPUA framework, we conduct evaluations on real power consumption data and user/substation location data collected from Xinjiang Province in China for 35 days. The evaluation results demonstrate that our SPUA framework can achieve a 20%-65% reduction on the maximum substation utilization, and 2 to 3.7 times reduction on total transmission loss. Moreover, we have developed a SPUA project site [13], and publicized our system code [14] to facilitate the research community.

The rest of the paper is organized as follows. Section 2 presents the motivations, defines the problem, and overviews the key components of our framework. Section 3 provides detailed methodology of SPUA in a centralized optimization algorithm. Section 4 scales up the algorithm by developing a distributed method. Section 5 presents evaluation results on a real large-scale power consumption dataset. Related works are discussed in Section 7, and the paper is concluded in Section 8.

2 Overview

In this section, we motivate and define the power user assignment problem, describe the dataset we use, and outline the solution framework.

2.1 Motivations

A power grid consists of a network of *power plants* and *power substations*. A power plant is an industrial facility for the generation of electric power, which contains one or more generators. A power substation as a part of an electrical generation, transmission, and distribution system, transforms voltage from high to low, or the reverse. Power substations typically serve a group of power consumers. For example, by the year of 2015, there were 783 power substations in Xinjiang province in China that provide electrical power to a total of 6.3 million users for their daily power consumption of residential and industrial purposes, which covers a wide geographic region of 1.6 million of square kilometers. Due to the global urbanization and human mobility, the population size and density change geographically over time, which drives the need to upgrade the power grid network infrastructure to remedy two main issues: long distance user coverage and over- and under-supplied power substations.

Long-distance user coverage. The electrical power transmission incurs certain transmission costs. The longer the user is from the substation, the more power transmission loss [15]. Studies have shown that the power transmission loss is proportional to the transmission distance and the square of power demands. From the real data, we observe that many users are covered by a long distance power substation, rather than a nearby one. Figure 1 shows five power substations in Xijiang province that cover users who are 300 km or more away from the substation.

Over- and under-supplied power substations. A power substation when being designed and deployed has a certain capacity, namely, a maximum amount of electrical power can be provided per unit time (e.g., one hour). Over time, the power demand of some power substations may increase drastically, and exceed the substation capacity, leading to under-supplied scenario. On the other hand, the population density may decrease in the regions covered by some power substations, which would lead to over-supplied scenario, where the substation utilization becomes lower. For example, Figure 2 and 3 show the substations with highest and lowest power utilization during peak and valley hours, respectively. For those busy power substations, they are primarily located in regions with high population densities, such as downtown of Urumqi City.

Motivated by these observations, we aim to develop a scalable power user assignment framework, that assigns each user to a power substation by analyzing large-scale power consumption data, while maintaining low substation utilizations. Next, we define the power user assignment problem. Besides distribution automation through

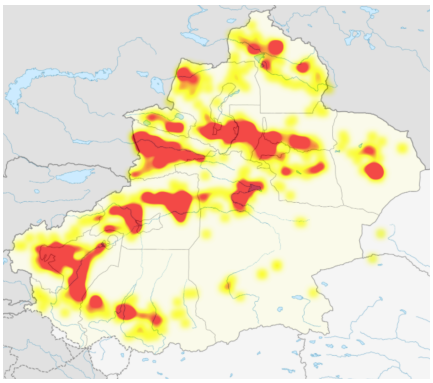


Fig. 4. User distribution

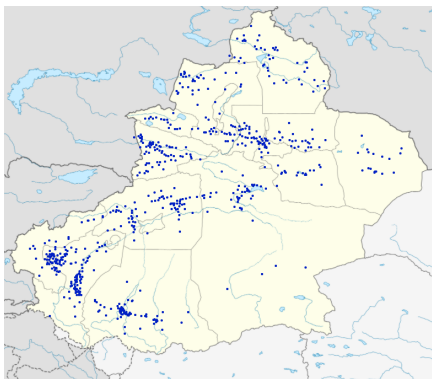


Fig. 5. Substation locations

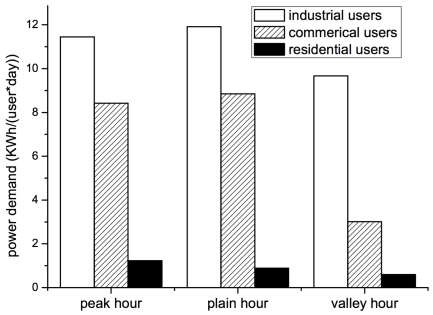


Fig. 6. Average daily power consumption

reassigning the users to substations, there are alternative methods to tackle the above two challenges, including upgrading/degrading the substation capacity or deploying/removing new power substations. However, those methods are more costly in terms of redeployment cost [16], and reassignment of users and substations are still needed after applying these methods. Thus, in this paper, we focus on the solution based on reassigning users to substations.

2.2 Problem Definition

Given a power grid system, we denote S the set of power substations, where each power substation $i \in S$ has a location in latitude and longitude, and a capacity $c_i \in C$ in kWh, indicating the maximum electrical power it can support for each hour. Moreover, the power grid system consists of a set of users denoted by U. A user could be a residential user, a factory, a commercial location, etc. Each user $j \in U$ has a location, and generates temporal power consumption data over time. Depending on the underlying power system, a power meter reading is reported after each pre-defined time interval Δt in hours, which could be 0.25 hours, 8 hours, 24 hours (a day), etc. Hence, we denote $Q_j = [Q_j(1), \cdots, Q_j(t)]$ as a power consumption sequence for user $j \in U$ from the 1st time interval to the t-th time interval. $\mathcal{Q} = \{Q_j | j \in U\}$ represents all users' power consumption sequences. $d_j = \sum_{\ell=1}^{t} Q_j(\ell)/(t \cdot \Delta t)$ is the per hour power consumption of user j. We denote $D = [d_j]$ as the list of hourly user power consumptions. Given an instance $X = [x_{ij}]$ of user-substation assignment, each x_{ij} represents a binary variable, indicating a user j is assigned to substation i, if $x_{ij} = 1$; and $x_{ij} = 0$ otherwise. Given a user $j \in U$, the total hourly power consumption for assigning it to substation $i \in S$ is $p_{ij} = d_j + \alpha d_j^2 \mathsf{dist}_{ij}$, which contains d_j the hourly power consumed by the user j, and $\alpha d_j^2 \mathsf{dist}_{ij}$ the transmission loss incurred by transmitting d_j amount of power from the substation i to user j [15]. Such transmission loss is a product of a system factor α, the (Euclidean) distance dist_{ij} (in kilometers) between station i and user j, and the square of user j's hourly power consumption d_j^2. Thus, for a substation $i \in S$, its power utilization ℓ_i is the ratio between p_{ij} the total user power demand with the operation cost by transmission loss and c_i the substation capacity, namely, $\ell_i = \sum_j p_{ij} x_{ij} / c_i$. Now, we formally define the power user assignment problem as follows which minimizes the maximum substation utilization of all power substations.

Problem definition. Given a set of substations S with capacity C and users U with their hourly power consumption D, we aim to find an optimal substation-user assignment X, so that each user is covered by exactly one power substation, and the maximum substation utilization $\ell = \max_{i=1}^{|S|} \ell_i$ is minimized.

2.3 Data Description

We use a large-scale real power grid dataset for this study, including (1) power user profiles, (2) power substation profiles, and (3) temporal user power consumption data. The datasets were collected from Xinjiang Province during March 10th – April 13th in 2015.

Power user locations. The dataset contains in total 6.3 million unique users, with their unique *user IDs*. Note that users include 6.16 million residential users and 0.14 million commercial and industrial users. Each residential user has a home address and a primary user name. In general, a residential user represents a family living in the same apartment or house. A commercial or industrial user has its business address, and the business name. Figure 4 shows the geo-distribution heatmap of all users in our datasets. Clearly, there are significant differences in user density across the entire province.

Power substation locations and capacities. At the time of data collection, there were 783 power substations deployed in Xijiang province in China. Each substation has a *substation ID*, *address* and *substation capacity*, namely, the maximum electrical power it can provide per hour. Figure 5 shows the locations of power substations. More substations with higher capacities are deployed near big cities, such as Urumqi and Turpan, to better serve the areas with high power demand and population density.

Note that the original data only contain the user and substation addresses in a standard format as [*province, city, county, township, village/road, building, unit, room*]. We parsed the addresses into locations in latitude and longitude using BAIDU Geo-Coding APIs [17], and cross-validated using Google Geo-Coding APIs [18]. There are about 25% user records with missing or incomplete addresses, which were therefore eliminated from the dataset.

Temporal user power consumption data. This dataset contains both the user-substation assignment information

and the dynamic power usage for each individual user. Each user with a user ID *uid* is uniquely assigned to a substation *sid*, represented as a tuple $\langle uid, sid \rangle$. Moreover, the dataset contains the power usage for all users over 35 days (March 10th – April 13th) in 2015. For each user, the dataset records the total daily power consumption, and the power consumptions for peak hours (9AM-1PM and 9PM–1AM), plain hours (1PM–9PM), and valley hours (1AM–9AM), respectively. Figure 6 shows the average daily power consumption over the three periods by residential, commercial, and industrial users.

2.4 System Framework

Figure 7 presents our scalable power user assignment (SPUA) framework. It takes three datasets as inputs, including power user profiles, power substation profiles, and user power consumption. The whole framework consists of three stages (highlighted as three dashed boxes): (1) user aggregation, (2) user/substation clustering, and (3) user assignment.

•**Stage 1 (User aggregation):** In a real power grid system, due to various system constraints it is not possible to assign individual users to just any substation. For example, users on the same distribution line or transformer, e.g., in the same building, or school, have to be assigned/switched to the same power substation. We in this stage aggregate power 6.3 million power users based on their locations, namely, uses with the same latitude and longitude will be grouped to an aggregated user. For each aggregated user, the power consumption dynamics are also aggregated from all the associated individual users. Then, the user assignment problem transfers to assigning the aggregated users to the substations.

•**Stage 2 (User/substation clustering):** Given a massive amount of users to assign to the substations, it is challenging to tackle such a problem in a centralized fashion. Thus, in this stage, the aggregated users and power substations are clustered into k small geographical regions, each of which contains a subset of aggregated users and power substations. Moreover, some "edge" aggregated users who are located in-between of a few clusters are identified, and they can be potentially assigned to one of the nearby clusters. Those clustered substations and users, as well as edge users, will be fed into stage 3 as input.

•**Stage 3 (User assignment):** In this stage, we first formulate the power user assignment problem as an integer linear programming problem with the objective of minimizing the maximum power utilization among all power substations. To solve this problem in a large-scale scenario, we develop a distributed approximation algorithm by applying the block splitting algorithm [12] and a 2-approximation rounding algorithm.

Table 1 provides notations used throughout the paper.

3 Methodology

In this section, we elaborate on the user aggregation stage and the centralized framework of solving user assignment problem. We also highlight the scalability challenges in applying the centralized method, that subsequently leads to our design for a distributed solution in Section 4.

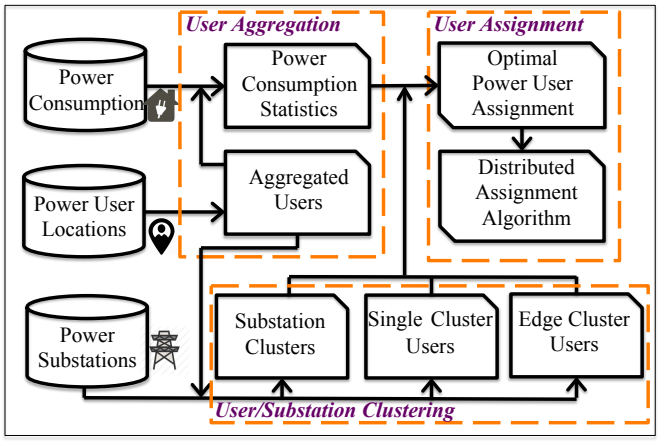


Fig. 7. Scalable power user assignment

TABLE 1
Notations and terminologies

Notations	Descriptions
S, U, U_a	Set of substations, users, aggregated users
n, m	number of substations, $n = \|S\|$, and aggregated users, $m = \|U_a\|$
x_{ij}	Indicator variable. 1: user j is assigned to substation i, 0 otherwise
$C = [c_i]$	capacity of substation i
$D_a = [d_j]$	Average hourly power demand of aggregated user j during peak hours
α	System factor, governing the transmission loss
dist_{ij}	distance between substation i and user j
ℓ_i, ℓ	Power utilization of substation i, and maximum power utilization

3.1 Stage 1: User Aggregation

Each individual power user is usually directly connected to a closest transformer, instead of a power substation, and there may be multiple hierachical transformers between a user and its substation to transform the voltage from high to low, or the reverse. Thus, when switching a user to another substation, a family of users that connect to the same transformer have to be switched together. To consider such constraints, we aggregate the power users with the same or close locations to an aggregated super user, and conduct the user assignment for aggregated users. We use a granularity of 0.0005 degrees in latitude and longitude, roughly 50 meters distance, to aggregate users. Basically, we divide the entire Xinjiang Province into small grids with equal side length of 0.0005 degrees. All residential users falling into the grid will be aggregated as a super user. It is worth mentioning that we only aggregate residential users (who tend to have lower amounts of power consumption), not commercial or industrial users. After the aggregation, we extracted $m = 21,801$ aggregated users from 6.3 million individual users. Some aggregated users contain more than 1,000 users. Then, the user assignment problem becomes assigning aggregated users to the substations. For simplicity and conciseness, we will use power users to refer to aggregated power users throughout the remainder of this paper.

Given a group of individual users who form an aggregated user, we sum up all power consumed by individual users to extract the power consumption for the aggregate user. For each aggregated user $j \in U_a$, we extract the average hourly power consumption $d_j \in D_a$ during peak hours. D_a will be used as input in the user assignment stage to determine the optimal assignment solution.

3.2 Problem Formulation

Given a set of substations S with capacity C, (aggregated) users U_a, together with the average user peak hour demand D_a, we are now in a position to formulate the power user assignment problem, with the goal of minimizing the maximum power substation utilization. Given a user $j \in U_a$, the total hourly power consumption for assigning it to substation $i \in S$ is $p_{ij} = d_j + \alpha d_j^2 \mathsf{dist}_{ij}$, which contains d_j the actual average hourly power consumption during the peak hours and $\alpha d_j^2 \mathsf{dist}_{ij}$ the transmission loss incurred by transmitting d_j amount of power from the substation i to user j [15]. Note that we use the average hourly user power demand during peak hours $D_a = [d_j]$ instead of over all 24 hours, because the highest power utilization of substations in general occurs during peak hours. The transmission loss is a product of a system factor [1] α, the (Euclidean) distance dist_{ij} (in kilometers) between station i and user j, and the square of user j's hourly power consumption in peak hours d_j^2. Thus, the substation power utilization ℓ_i is the ratio between the total user power demand with the operation cost by transmission loss p_{ij} and the substation capacity c_i, namely, $\ell_i = \sum_j p_{ij} x_{ij} / c_i$. Each $d_j \in D_a$ is extracted from the past power consumption data in the user aggregation stage. Let ℓ be the maximum substation power utilization. We denote a decision variable x_{ij} as a binary indicator variable, indicating that a user $j \in U_a$ is assigned to a station $i \in S$ when $x_{ij} = 1$, and $x_{ij} = 0$ otherwise. We aim to find the optimal assignment of all x_{ij} values that leads to the smallest possible ℓ. This problem is formally formulated as below.

$$\text{min:} \quad \ell \tag{1}$$

$$\text{s.t.:} \sum_{j \in U_a} \frac{p_{ij}}{c_i} x_{ij} \le \ell, \qquad \forall i \in S, \tag{2}$$

$$\sum_{i \in S} x_{ij} = 1, \qquad \forall j \in U_a, \tag{3}$$

$$x_{ij} = \{0, 1\}, \quad 0 \le \ell \le 1, \qquad \forall i \in S, j \in U_a. \tag{4}$$

The objective function eq.(1) is to minimize the maximum utilization ℓ for all power substations. The constraint in eq.(2) indicates the power substation capacity constraint, namely, for a substation $i \in S$, the substation power utilization ℓ_i is no more than the maximum power utilization ℓ. The validity constraint in eq.(3) indicates that any power user is covered by exactly one power substation.

Approximate Solution with LP-Rounding. The above integer linear programming (ILP) problem can be viewed as a *makespan scheduling problem with unrelated machines* or *scheduling on unrelated parallel machines* as follows. Suppose n jobs are to be assigned to m machines for scheduling, where job j costs p_{ij} units of time if scheduled on machine i. Let J_i be the set of jobs scheduled on machine i. Then $\ell_i = \sum_{j \in J_i} p_{ij}$ is the load of machine i. The maximum load $\ell = \max_i \ell_i$ to be minimized is called the makespan of the schedule. In our user assignment problem eq.(1)–(4), the makespan is the maximum power utilization of substations. The problem is NP-hard and has been extensively studied in the literature, with a variety of approximation algorithms proposed that employ LP-rounding approaches [7], [8], [9], [10]. These methods generally contain two steps, namely, LP-relaxation followed by rounding. For example, [7] proposed an approximation algorithm with a worst case error bound of $2\sqrt{n}$, where n is the number of machines (i.e., substations in our case). [8] gave a 2-approximation for this problem, and they proved that it is not possible to approximate it within a factor $(3/2\epsilon)$ for any $\epsilon > 0$, unless $P = NP$. In [9], the authors improved the bound in [8] from 2 to $2 - 1/m$. [10] provides a comprehensive study in evaluating different approximation algorithms and proposed a fast meta-heuristic algorithm without theoretical performance guarantee. In this study, we adopt the approximation solution algorithm proposed in [8] based on LP-rounding. Other algorithms can be chosen, depending on the specific requirements on the error bound and complexity. Our approximation solution algorithm consists of two steps below.

Step 1: LP Relaxation. Instead of simply relaxing the integer constraints eq.(4) to $0 \le x_{ij} \le 1$, we relax the ILP problem defined in eq.(1)–(4) into a family of linear programming problems $LP(\ell)$, where ℓ is viewed as constant in each $LP(\ell)$. Let the parameter ℓ be a "guess" of a lower bound for the actual maximum substation utilization (i.e., "makespan") ℓ^*. We perform binary search on ℓ to determine a suitable value in an outer loop. Fixing a value for ℓ enables us to enforce constraints $x_{ij} = 0$ for all substation-user pairs (i, j) for which $p_{ij}/c_i > \ell$. Define $E_\ell = \{(i, j) : p_{ij}/c_i \le \ell\}$. We can define a family of $LP(\ell)$ of linear programs, one for each value of the parameter ℓ. $LP(\ell)$ uses the variables x_{ij} for which $(i, j) \in E_\ell$ and asks if there is a feasible solution of $LP(\ell)$ below.

$$\text{min:} \quad \ell \quad \text{(constant)} \tag{5}$$

$$\text{s.t.:} \sum_{j:(i,j) \in E_\ell} \frac{p_{ij}}{c_i} x_{ij} \le \ell, \qquad \forall i \in S, \tag{6}$$

$$\sum_{i:(i,j) \in E_\ell} x_{ij} = 1, \qquad \forall j \in U_a, \tag{7}$$

$$x_{ij} \ge 0, \qquad \forall (i, j) \in E_\ell, \tag{8}$$

$$x_{ij} = 0, 0 \le \ell \le 1, \qquad \forall (i, j) \notin E_\ell. \tag{9}$$

The search space for ℓ is defined as follows. We generate a user assignment configuration, by assigning each user $j \in U_a$ to one station $i \in S$, that has the smallest p_{ij}/c_i, that is, user j is assigned to the station $i = \text{argmin}_{i \in S}\{p_{ij}/c_i\}$. Given such an assignment, let $\beta = \max_i \ell_i$ be the maximum power utilization among all substations after this assignment. With a binary search in the range of $[\beta/n, \beta]$, we

1. The multiplier α can be calculated as the conductor resistance of feeder (in ohm/km) divided by the square of nominal voltage (in volts) [15]. As the resistance of copper conductor is usually 1–4 ohm/km and the distribution voltage is 10kv or 22kv, we choose the system factor α to be within $[10^{-6}, 4 \cdot 10^{-6}]$.

find the smallest value for ℓ such that $LP(\ell)$ has a feasible solution. Let ℓ_{LP} be this value and observe that $\ell^* \geq \ell_{LP}$, i.e., the actual smallest maximum substation utilization ℓ^* is bounded from below by ℓ_{LP}. The rounding algorithm will "round" the fractional solution of $LP(\ell)$ to yield a schedule with ℓ at most $2\ell^*$.

Algorithm 1 Approximate Power User Assignment Algorithm

1: **Input:** U_a, S, D_a, α, dist_{ij};
2: **Output:** $x_{ij} \in \{0, 1\}$, ℓ;
3: **for** $j \in S_a$ **do**
4: $\quad y_{ij} = 1$, if $i = \text{argmin}_{i \in S}\{p_{ij}/c_i\}$, and 0, otherwise;
5: $\beta = \max_i \sum_{j \in U_a} p_{ij} y_{ij}/c_i$;
6: Binary search ℓ in $[\beta/n, \beta]$ for smallest ℓ that $LP(\ell)$ has a feasible solution $[x_{ij}]$;
7: Construct bipartite graph H and find perfect matching M;
8: Round in $X = [x_{ij}]$ all fractionally set jobs according to the matching M;

Step 2: Rounding LP Solutions. Algorithm 1 outlines the overall approximate power user assignment algorithm. Lines 3–6 outline the LP relaxation step. Line 7 constructs a bipartite graph $G = (U_a \bigcup S, E)$ with users and substations as the two sets of entities. Each edge $(i, j) \in E$ if and only if x_{ij} in the solution from step 1 satisfies $x_{ij} > 0$. Let $F \subseteq U_a$ be a subset of users whose x_{ij} are fractional, namely, $0 < x_{ij} < 1$. Each user that is integrally set in $[x_{ij}]$ has exactly one edge incident at it in G. Remove these users together with their incident edges from G. The resulting graph is H. Thus, an equal number of edges and vertices have been removed from G. In H, each user has a degree of at least two. So, all nodes with a degree of 1 in H must be substations. Clearly $(i, j) \in E(H)$ if $0 < x_{ij} < 1$. A matching in H is called perfect if it matches every user $j \in F$. We omit the proof that the graph H admits perfect matchings. To find a perfect matching in H, we keep matching nodes with a degree of 1 with the user it is incident to and remove them both from the graph. At each stage all nodes with degree of 1 must be substations. In the end we will be left with even cycles. Match alternating edges of each cycle. This gives a perfect matching M. In Line 8, we simply round in $[x_{ij}]$ all fractionally set users according to the matching M. Lemma 1 below provides the approximation bound of Algorithm 1, where the proof can be completed using the same idea as that in [8]. We omit it here for brevity.

Lemma 1. *Algorithm 1 assigns each power user in U_a to one substation in S, and the maximum substation utilization ℓ obtained by such assignment is no more than $2\ell^*$, where ℓ^* is the optimal objective value to the problem eq.(1)–(4).*

Practical issue. In fact, all of the approximation algorithms proposed in the literature [7], [8], [9], [10] for the makespan scheduling problem with unrelated machines assume that the induced linear programming problems $LP(\ell)$ defined in eq.(5)–(9) are solvable with reasonable scales. However, in our power user assignment problem, even after aggregation, we have $m = 21,801$ (aggregated) users to be assigned to $n = 783$ substations. Hence, the decision variables x_{ij}'s to be solved is at a scale of $O(n \times m) \approx 1.6 \times 10^7$. It is very hard to solve such problem with state-of-the-art LP solvers [19]. Hence, we propose a decomposition based method to tackle this issue using the block-splitting algorithm [12]. The basic idea is to decompose the entire target region into small regions, with edge users (variables) at the border lines across clusters. Then, we can solve the LP problem in each small region in parallel, followed by re-assignment of edge users to a nearby region. This process is iterated multiple rounds, until the resulting solution converges. We will elaborate on our distributed algorithm for solving $LP(\ell)$ in the next section.

4 Distributed Algorithm for $LP(\ell)$

The major difficulty in solving the LP problem $LP(\ell)$ defined in eq.(5)–(9) is that its problem size in millions of variables, making it unsolvable using a centralized LP solver. In this section, we first show how we decompose the target geographical region into smaller regions (i.e., Stage 2 in Figure 7), which enables $LP(\ell)$ to be re-organized with a sparse constraint matrix. Then, by employing the block-splitting algorithm of [12], $LP(\ell)$ can be solved in a distributed manner (i.e., Stage 3 in Figure 7).

4.1 Stage 2: User/Substation Clustering

The goal of clustering users and substations is to have a number of geographical sub-regions, that the total number of decision variables (i.e., the product of the number of users and substations) in each region is relatively small, so that the sub-problem of $LP(\ell)$ in each region has a reasonable scale size, thus solvable. We develop a two-step approach to cluster substations and users as follows.

Step 1: For substation clustering, the input is the number of desired clusters, i.e., N. Then, the k-means algorithm [20] is used to cluster the substations into N clusters. The output of the substation clustering will be a non-overlapping partition $\Pi_S = \{S_1, \ldots, S_N\}$ of the set of substations with $S = S_1 \bigcup \ldots \bigcup S_N$. A set S_k is called a *region*. Figure 8 visualizes a clustering result with $N = 15$ clusters. We use different colors and marker shapes to represent different regions.

Step 2: User clustering aims to find the primary cluster of each user, and a group of edge users, who are at the border lines across clusters, thus may be assigned to a substation from different clusters. The user clustering is based on the Euclidean distance between users and substations, denoted as dist_{ij} for user j and substation i. Each user has one and only one primary cluster. The set of users are partitioned as $\Pi_U = \{U_1, \ldots, U_N\}$, with $U_a = U_1 \bigcup \ldots \bigcup U_N$. Given the clustered substations Π_S, we can find the primary cluster for each user j as U_k, if the nearest substation is located in S_k. We can control the number of edge users, by changing n_c, which is the number of allowed nearest candidate substations (of users). When $n_c = 1$, each user can only be assigned to her nearest station, thus there will be no edge users in this case. When $n_c > 1$, each user can have those n_c nearest substations as her candidate substations. This way, a user j at the border lines across clusters may have some candidate substations not in her primary cluster, so becomes an edge user. In other words, for each user j,

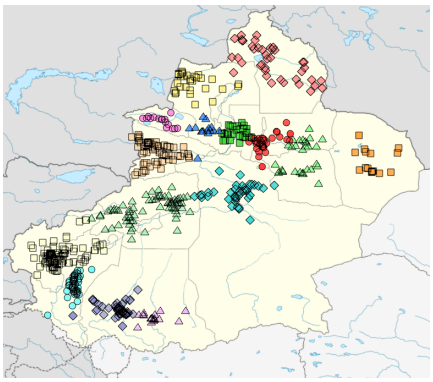


Fig. 8. Substation clustering with $N = 15$

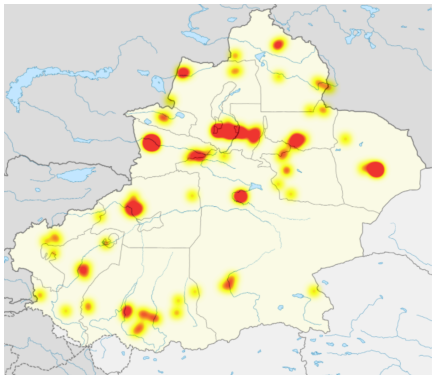


Fig. 9. Edge user distribution ($N = 15$, $n_c = 15$)

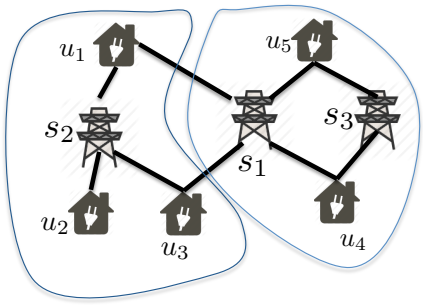


Fig. 10. Illustration of LP problem decomposition

its n_c closest connections to substations are considered as candidate assignments, and we denote $E(n_c)$ as all such substation-user pairs (i, j)'s.

Figure 9 illustrates edge user geo-distribution using a heat map with $N = 15$ and $n_c = 15$ clusters. The edge users are clearly located at the border lines across clusters.

4.2 Stage 3: Distributed User Assignment

Given the decomposition of user set $\Pi_U = \{U_1, \ldots, U_N\}$ and substation set $\Pi_S = \{S_1, \ldots, S_N\}$, we are in a position to present how we transform the $LP(\ell)$ problem into a distributed optimization problem, by decomposing the variable set, rearranging the capacity constraints, and projection for the equality constraints.

Decomposition of decision variable set $X = [x_{ij}]$. There are $n_c \times n$ candidate substation-user pairs $E(n_c)$ extracted from the user clustering stage , which determines the set of decision variables $X = [x_{ij}]$ in $LP(\ell)$ problem. Namely, if $(i, j) \in E(n_c)$, then x_{ij} is a decision variable. Otherwise x_{ij} is not a decision variable. Given these decision variables and the user set decomposition $\Pi_U = \{U_1, \ldots, U_N\}$, we decompose the decision variables $x_{ij}, 1 \le i \le n, 1 \le j \le m$ as a finite set of subsets $X = \{X_0, X_1, \ldots, X_N\}$ in the following way: 1) Initialize $X_k = \emptyset$ for $k = 0, 1, \ldots, N$; 2) For each user $j \in U_k$, if its decision variable x_{ij} has $i \notin S_k$, then x_{ij} is included in X_0; otherwise x_{ij} is included in X_k. Hence, $X_0 = \{x_{ij} | i \in S_{k_1}, j \in U_{k_2}, k_1 \neq k_2\}$, and $X_k = \{x_{ij} | i \in S_k, j \in U_k\}$ with $1 \le k \le N$. The set of variables in X_0 are called *coordinating variables* and the set of variables in X_k are called *internal variables* of region k. We write variables in X_k in vector form $\mathbf{x}_k$, for $k = 0, 1, \ldots, N$. With such a decomposition of decision variables, it is clear that given a user $j \in U_k$, each decision variable x_{ij} is either in X_k or X_0. Moreover, for each station $i \in S$, we introduce slack variables ϵ_i to make inequality constraints into equality constraints. The slack variables are considered as internal variables and included into X_k for $i \in S_k$ with $k = 1, \ldots, N$.

Rearranging capacity constraints. With the decomposition of variables, the capacity constrains eq.(6) can be rearranged in a sparse block form, as shown in Lemma 2 below.

Lemma 2. *For each $k = 1, \ldots, N$, each capacity constraint in eq.(6) can be written as*

$$\mathbf{w}^T \mathbf{x}_0 + \mathbf{v}^T \mathbf{x}_k = \mathbf{b}$$

for some $\mathbf{w}$, $\mathbf{v}$, $\mathbf{b} = [\ell, \ldots, \ell]^T \in \mathbb{R}^{|S_k|}$, and k.

Proof. For a station $i \in S_k$, the capacity constraint follows

$$\sum_{j=1}^{m} \frac{p_{ij}}{c_i} x_{ij} + \epsilon_i = \sum_{j \notin U_k} \frac{p_{ij}}{c_i} x_{ij} + \sum_{j \in U_k} \frac{p_{ij}}{c_i} x_{ij} + \epsilon_i$$
$$= \sum_{x_{ij} \in X_0} \frac{p_{ij}}{c_i} x_{ij} + \sum_{x_{ij} \in X_k} \frac{p_{ij}}{c_i} x_{ij} + \epsilon_i = \ell.$$

Note that x_{ij}'s and ϵ_i from X_k form the vector $\mathbf{x}_k$, and x_{ij}'s from X_0 form $\mathbf{x}_0$, which completes the proof. □

Projection for equality constraints. The equality constraints in eq.(7) can be viewed as a linear projection operation. For any vector $\mathbf{x} \in \mathbb{R}^{|X|}$, we can enforce (i.e., transform) it to satisfy equality constraints, by simply projecting the non-slack decision variables x_{ij}'s onto the probability simplex governed by equality constraints, $\sum_{i \in S} x_{ij} = 1$ for each user j, and projecting the slack variables onto the positive orthant. Such projection (denoted by $\mathbf{x} \in \text{Range}(\mathbf{x})$ for notational simplicity) yields the vector $\mathbf{x}$, which is feasible to equality constraints in $LP(\ell)$. Moreover, this linear projection operation can be done in polynomial time with the method of [21].

Transforming the problem $LP(\ell)$. After decomposing the decision variable set X, rearranging the capacity constraints, and projection for equality constraints, the LP problem $LP(\ell)$ in eq.(5)–(9) is transformed to the following matrix form:

$$\min_{\mathbf{x} \in \text{Range}(\mathbf{x})} \ell, \quad \text{subject to } A\mathbf{x} = \mathbf{b}, \tag{10}$$

where $\mathbf{x}$ is a vector obtained by stacking all $\mathbf{x}_k$, $k = 0, 1, \ldots, N$ together, $A\mathbf{x} = \mathbf{b}$ indicates the capacity constraints, and $\mathbf{x} \in \text{Range}(\mathbf{x})$ represents the feasible space for equality constraints in eq.(7). Since the objective function ℓ is a constant, solving this problem is equivalent to finding a solution $\mathbf{x}$ that is simultaneously feasible to capacity constraints $A\mathbf{x} = \mathbf{b}$ and equality constraints Range($\mathbf{x}$). From Lemma 2, we rewrite the capacity constraints $A\mathbf{x} = \mathbf{b}$ in a matrix form as follows:

$$A_{k0}\mathbf{x}_0 + A_{kk}\mathbf{x}_k = \mathbf{b}_k, \qquad \text{for } 1 \le k \le N, \tag{11}$$

$$\begin{array}{lll|l}
\text{substation } s_1: & x_{11}+x_{13}+\quad x_{14}+x_{15}+\epsilon_1=\ell & \equiv [1,1]\cdot\mathbf{x}_0+[1,1,0,0,1,0]\cdot\mathbf{x}_2=\ell & \mathbf{w}_1^\top=[1,1];\mathbf{v}_1^\top=[1,1,0,0,1,0]\\
\text{substation } s_2: & x_{21}+x_{22}+x_{23}+\epsilon_2=\ell & \equiv [0,0]\cdot\mathbf{x}_0+[1,1,1,1]\cdot\mathbf{x}_1=\ell & \mathbf{w}_2^\top=[0,0];\mathbf{v}_2^\top=[1,1,1,1]\\
\text{substation } s_3: & x_{34}+x_{35}+\epsilon_3=\ell & \equiv [0,0]\cdot\mathbf{x}_0+[0,0,1,1,0,1]\cdot\mathbf{x}_2=\ell & \mathbf{w}_3^\top=[0,0];\mathbf{v}_3^\top=[0,0,1,1,0,1]
\end{array}$$

Fig. 11. Illustration for decomposition of decision variables and rearranging capacity constraints (for example in Figure 10).

$$\left.\begin{array}{ll}
\text{user } u_1: & x_{11}+x_{21}=1\\
\text{user } u_2: & x_{22}=1\\
\text{user } u_3: & x_{13}+x_{23}=1\\
\text{user } u_4: & x_{14}+x_{34}=1\\
\text{user } u_5: & x_{15}+x_{35}=1
\end{array}\right\} = \begin{bmatrix}
1&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0\\
0&1&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&1&0&0&0\\
0&0&0&0&0&0&0&1&0&1&0&0
\end{bmatrix}\cdot\begin{bmatrix}\mathbf{x}_0\\ \mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix}=\mathbf{1}$$

$\mathbf{x}=\text{Range}(\mathbf{x})$

Fig. 12. Illustration for projection of equality constraints (for example in Figure 10).

thus A can be reorganized as the following sparse block form.

$$A=\begin{bmatrix}A_{10} & A_{11} & & & \\ A_{20} & & A_{22} & & \mathbf{0}\\ \vdots & \mathbf{0} & & \ddots & \\ A_{N0} & & & & A_{NN}\end{bmatrix},\quad \mathbf{b}=\begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\\ \vdots\\ \mathbf{b}_N\end{bmatrix}.$$

Below we include a simple example to illustrate the problem decomposition and present a distributed algorithm that integrates both block-splitting for capacity constraints and projection for equality constraints in an iterative fashion.

Illustration example. Consider a simple assignment problem in Figure 10, the decomposition generates a partition of stations: $S_1=\{s_2\}$ and $S_2=\{s_1,s_3\}$, and a partition of users $U_1=\{u_1,u_2,u_3\}$, $U_2=\{u_4,u_5\}$.

Decomposition of decision variable set. Given decision variables $x_{ij}, i=1,\ldots,3, j=1,\ldots,5$, this decomposition yields a decomposition of variables into $\mathbf{x}_0=[x_{11},x_{13}]$, $\mathbf{x}_1=[x_{21},x_{22},x_{23},\epsilon_2]$, and $\mathbf{x}_2=[x_{14},x_{15},x_{34},x_{35},\epsilon_1,\epsilon_3]$. Note again that slack variables ϵ_i with $i=\{1,2,3\}$ are incorporated as internal variables.

Rearranging capacity constraints. Without loss of generality, assuming all weights p_{ij}/c_i equal to 1, Lemma 2 can be verified by writing the capacity constraint of each substation i as shown in Figure 11.

Projection for equality constraints. The equality constrains in eq.(7) can be viewed as a linear projection operation $\mathbf{x}\in\text{Range}(\mathbf{x})$ as shown in Figure 12.

Distributed Optimization Algorithm for $LP(\ell)$. We solve the problems in eq.(10) using the block splitting algorithm based on ADMM in [22] and decomposition-based distributed synthesis in [23].

First, we introduce new variables $\mathbf{y}$, stacked by all $\mathbf{y}_k\in\mathbb{R}^{|S_k|}$ with $k=1,\ldots,N$, and let $f_k(\mathbf{y}_k)=I_{\{\mathbf{b}_i\}}(\mathbf{y}_i)$, where for a convex set C, I_C is a function defined by $I_C(\mathbf{z})=0$ for $\mathbf{z}\in C$, $I_C(\mathbf{z})=\infty$ for $\mathbf{z}\notin C$. Then, adding the term $f_k(\mathbf{y}_k)$ into the objective function enforces $\mathbf{y}_k=\mathbf{b}_k$. Suppose $\mathbf{x}_k\in\mathbb{R}^{|X_k|}$, let $g_k(\mathbf{x}_k)=\ell+I_{\text{Range}(\mathbf{x}_k)}(\mathbf{x}_k)$. The function $I_{\text{Range}(\mathbf{x}_k)}(\mathbf{x}_k)$ enforces that $\mathbf{x}_k$ is within its range.

We rewrite the LP problem in eq.(10) as follows.

$$\begin{aligned}&\min_{\mathbf{x}\in\text{Range}(\mathbf{x}),\mathbf{y}}\ \sum_{k=1}^{N}f_k(\mathbf{y}_k)+\sum_{k=0}^{N}g_k(\mathbf{x}_k)\\ &\text{subject to } \mathbf{y}_k=A_{k0}\mathbf{x}_0+A_{kk}\mathbf{x}_k,\ \text{for } k=1,\ldots,N.\end{aligned}\tag{12}$$

With this formulation, it is straightforward to apply the block splitting algorithm in [12] to solve eq.(12) in a parallel and distributed manner. See Appendix for the details.

5 Evaluations

To evaluate the performance of our scalable power user assignment (SPUA) framework, we conduct comprehensive experiments using a large scale power consumption dataset collected from Xinjiang Province in China. By comparing with baseline algorithms, the evaluation results demonstrate that SPUA can achieve a 20%-65% reduction on the maximum substation utilization, and 2 to 3.7 times reduction on total transmission loss. Below, we present our evaluation settings and results.

5.1 Evaluation settings

The dataset we use consists of 783 substations and 6.3 millions power users in Xinjiang Province in China, during March 10, 2015–April 14, 2015. All users are aggregated into 21,801 super users in near proximity. Each user is assigned to one substation in the dataset, and users' temporal power consumptions are also recorded for peak, plain, and valley hours of each day. The goal is to re-assign each aggregated user to a substation, so that the maximum substation utilization is minimized. Below, we highlight the baseline algorithms and evaluation configurations.

Baseline algorithms. We primarily compare our proposed SPUA method with three baseline algorithms, including the current user assignment (CUA), Distance-based user assignment (DBUA), and greedy method (Greedy).

(1) Current user assignment (CUA). This baseline algorithm employs the substation-user assignments observed from the real dataset.

(2) Distance-based user assignment (DBUA). This baseline algorithm simply assigns each user to its closest substation.

(3) Greedy method (Greedy). The idea behind this baseline algorithm is that we want to incrementally assign users to substations, so as to keep each substation with the relatively same utilization. It works as follows. In the first step, it assigns each substation with the closest user, and each substation has an initial utilization. The assigned users will be removed from the user set. Then, for each of the following steps, the substation with smallest utilization will be assigned with one user that is closest to it. The assignment process terminates when the user set is empty.

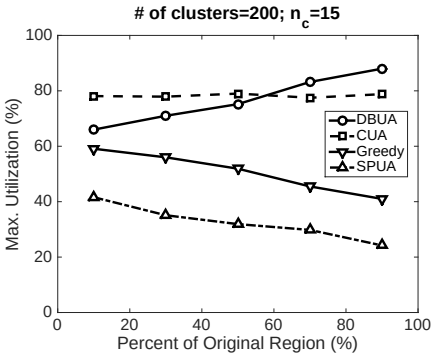


Fig. 13. Max. utilization vs problem scale

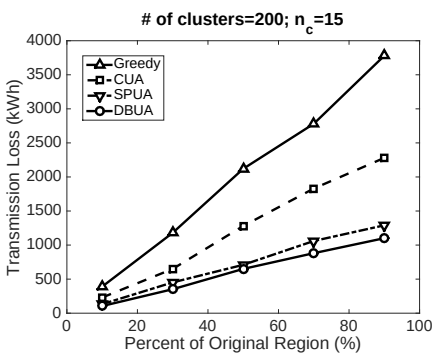


Fig. 14. Transmission loss vs problem scale

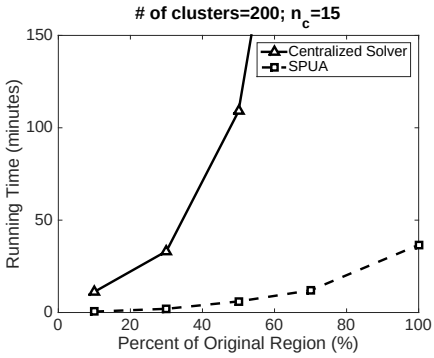


Fig. 15. Running time

TABLE 2
Evaluation configurations

% original scale	[10%, 30%, 50%, 70%, 90%]
# clusters	[100, 150, 200, 250, 300]
n_c	[5, 10, 15, 20, 25]
Assignment alg.	{SPUA, CUA, CBUA, Greedy}

Differing from these baselines, SPUA enables distributed assignment optimization, by partitioning the target area into small regions using user/substation clustering algorithm (Sec 4.1) and finding the solution with distributed user assignment algorithm (Sec 4.2).

Evaluation configurations. We evaluate our proposed SPUA using two performance metrics, including maximum substation utilization (max. utilization) and total transmission loss (in kWh). We also evaluate the convergence rate for SPUA method. We conducted three sets of evaluations as follows, to evaluate the scalability, stability, and practicality of SPUA method.

(1) Scalability. In this set of evaluations, we change the problem scale by choosing sub-regions with varying sizes, i.e., from 10% to 90% size of the entire dataset. For each size, e.g., 10%, we randomly generate 100 sub-regions, and take the average of the result from each region, to reduce the effect of randomness. Through the evaluations, we aim to understand how different methods perform for different sizes of the power user assignment problem.

(2) Stability. As proven in block-splitting paper [12], the decomposition of the problem does not affect much on the final result. In our power user assignment problem, we will examine how the results hinge on the numbers of clusters and edge users.

(3) Practicality. Finally, we will conduct case studies to look into the specific regions, and understand how our SPUA method improves user assignments.

Table 2 lists configurations used in our evaluation. All the experiments were run on a cluster which consists of three servers with Intel Xeon 2.4 GHz, 48-core CPU and 64 GB RAM running Linux. We used TORQUE Resource Manager to schedule massive jobs between cluster servers. The distributed optimization algorithm is implemented in MATLAB. The decomposition and other operations are implemented in Perl. Our project code package is available for download at [14].

5.2 Scalability Evaluation

Figures 13 shows the comparison results on the maximum substation utilization when applying our SPUA (200 clusters and 15 candidate nearest substations per user) and the baseline methods (i.e., DBUA, CUA and Greedy). We observe that our SPUA method has the lowest maximum substation utilization comparing all baseline methods, with a significant improvement ranging from 20% (over Greedy) to 65% (over DBUA at the scale of 90% original region size). As the size of the sub-region increases from 10% to 90%, the maximum substation utilization decreases with our SPUA method and Greedy method. The reason is that a larger underlying sub-region generally contains a larger number of users and substations, thus allows larger flexibility for SPUA and Greedy to assign and shift users across substations, leading to lower maximum substation utilization. Since the user assignment with CUA (from the data) does not change with the sub-region scale, the maximum substation utilization stays the same over sub-region sizes as well. On the other hand, the maximum substation utilization of DBUA increases with the sub-region size, because DBUA aims to assigns users to the nearest substation, without considering the substation utilization at all. Hence, the larger size the sub-region is, the worse substation utilization it has.

Similarly, when looking at the total transmission loss (in kWh), our SPUA always achieves lower total transmission loss over CUA and Greedy methods (as shown in Figure 14), with 2 to 3.7 times reduction. Notice that DBUA method has a slightly lower (about 30–190kWh) total transmission loss (per hour) than SPUA method, which is because DBUA is designed by nature to assign the nearest substations to users, thus leading to the lowest total transmission loss. However, comparing to the significant improvement (up to 65% reduction) of maximum substation utilization over DBUA method (from Figure 13), such a small increase on transmission loss is completely reasonable.

Running time and convergence. With a large number of decision variables, the original power user assignment problem as defined in eq.(1)–(4) do not scale up well. On the other hand, our SPUA solves this problem using a block-splitting algorithm with a theoretical guarantee that the maximum substation utilization obtained is no more than twice of the optimal solution of the original problem. Figure 15 shows the running time of solving eq.(1)–(4) with a centralized (optimization) solver vs our distributed SPUA.

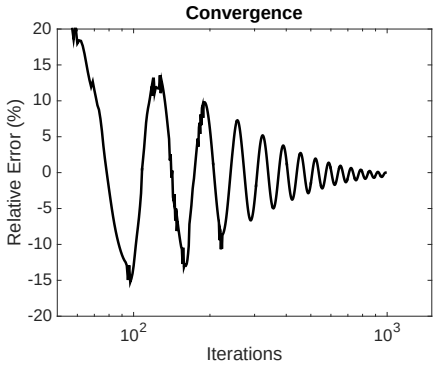


Fig. 16. Convergence of SPUA

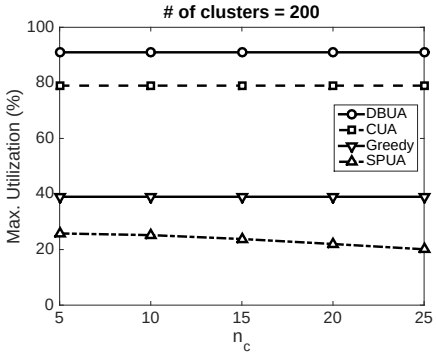


Fig. 17. Max. utilization vs n_c

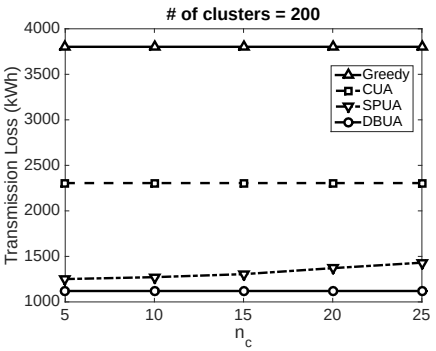


Fig. 18. Transmission loss vs n_c

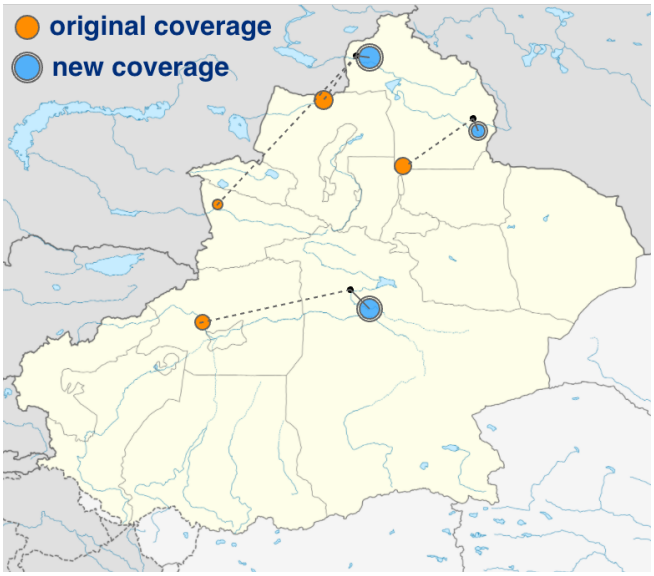


Fig. 19. Reduced covering distance

Fig. 20. Balanced substation utilization

It clearly shows that as the problem scales up, the running time increases drastically (for centralized solver), which is because the number of variables increases exponentially. By dividing and synchronizing the problem into small subproblems, our SPUA has much lower running time. For example, when 60% of the original region is considered, our SPUA has lower running time in two orders of magnitude than the centralized solver. Note that heuristic baseline algorithms, such as Greedy, CUA, and DBUA have low running time, but there is no performance guarantee on the obtained results, which leads to poor system performance in maximum substation utilization and transmission loss as shown in Figure 13–14. Figure 16 shows the convergence process of SPUA of one instance for the entire Xinjiang region, with 200 clusters and n_c as 15. The relative error of the objective value, i.e., the maximum substation utilization, fluctuates over $1,129$ iterations, taking in total 36.6 minutes before the convergence.

5.3 Stability Evaluation

We change the parameters including n_c, the number of candidate nearest substations per user and the number of clusters, to examine if SPUA method can consistently produce stable results. Figures 17–18 show that as we increase n_c, the maximum substation utilization and total transmission loss stay relatively the same. The maximum substation utilization (resp. total transmission loss) slightly decreases (resp. increases) while n_c increases, because a larger n_c allows more candidate substation-user assignments (with longer distances), thus leading to slightly lower maximum substation utilization (resp. more total transmission loss). Note that as the performances of baseline algorithms CUA, DBUS, and Greedy do not change over n_c, nor the number of clusters, we present their maximum substation utilization and transmission loss as constants to show the consistent high performance of our SPUA method. Again, DBUA assigns all users to their nearest substations, and it has slightly lower (about 100–300 kWh reduction of) transmission loss (in Figure 18), while sacrificing the maximum substation utilization. When we increase the number of clusters while keeping the same n_c, the maximum substation utilization and total transmission loss do not change for all four methods. This is because the number of candidate substation-user assignments are fixed given n_c, and so SPUA performs equally well with a varying number of clusters. We omit this set of results for brevity.

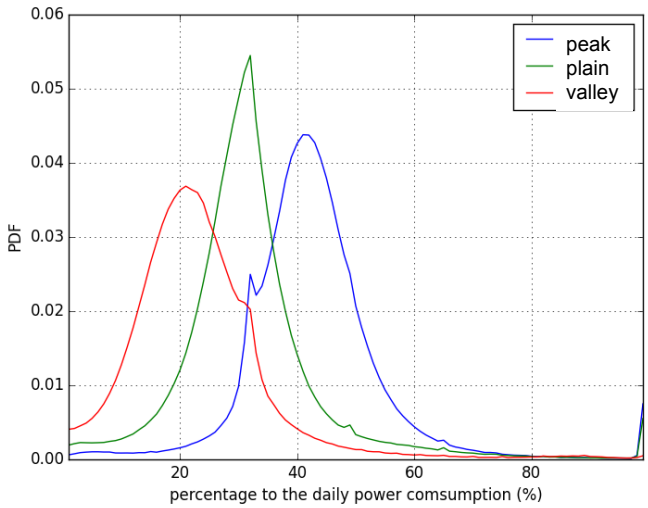


Fig. 21. Temporal dynamics of power consumption

5.4 Practicability Evaluation with Case Study

We look into the user assignment results obtained by SPUA vs the current assignment from the data (i.e., CUA). Figure 19 visualizes three substations with particularly long distance coverage in the existing user assignment. The black dots are the substations, and the orange circles are the current covering regions. Due to the high transmission loss, SPUA method re-assigns users from orange to blue circles, which are nearer in proximity.

Comparing to Figure 2, Figure 20 illustrates that SPUA balances the substation utilization across substations to circumvent the over- and under-supplied problems. For over-supplied substations, SPUA either merges some of them, or expands their coverage to achieve higher utilization. For under-supplied substations, SPUA reduces the covering range to decrease the substation utilization.

6 Discussions

User-substation assignment in real world power system design is a complex task, with many practical challenges and trade-offs to consider. In this section, we discuss two design challenges as further extensions to our SPUA framework, including temporal dynamics of power consumption and objective function choices.

6.1 Temporal dynamics of power consumption

So far, we characterize each user $j \in U$ using the average hourly power consumption d_j (see Sec 2.2), and assign users to substations based on d_j's. However, users' power consumption may change over time. During rush hours vs non-rush hours, different user types (commercial, industrial, and residential users) may exhibit different power consumption patterns. Figure 21 shows the distributions of power consumption in peak, plain, and valley hours from our dataset. For each user in each day, we calculate the percentage of power consumption ratios in peak, plain, and valley hours over the daily usage. Clearly, during peak hours, more power is consumed than plain and valley hours. Moreover, power consumption percentages of each time interval follows roughly a Gaussian distribution. Such variations indicate that for some users and days, the temporal dynamics matter. To incorporate this observation, we can characterize each user $j \in U$ using the maximum hourly power consumption d_j^{MAX} rather than the average hourly power consumption d_j. The intuition is that if the hour with the highest power consumption is guaranteed to not exceed the utilization limit, other hours are automatically guaranteed. However, d_j^{MAX} for some users may not occur very often, thus using d_j^{MAX} in SPUA may significantly over-estimate the actual power consumption for most of the time. All in all, there is a trade-off between using maximum vs average hourly power consumption (d_j^{MAX} vs d_j) in the user-substation assignment problem. In practice, the two can be (linearly) combined to characterize users in SPUA framework.

6.2 Objective function choices

Alternative objective functions. The maximum substation utilization $\ell_{MAX} = \max_{i=1}^{|S|} \ell_i$ is used as the design objective in SPUA. The intuition is that minimizing the maximum substation utilization can provide an upper bound guarantee for the substation utilization. In real world system design, it may be worthwhile to also guarantee a lower bound $\ell_{MIN} = \min_{i=1}^{|S|} \ell_i$ on substation utilization to avoid over-supplied scenarios, or minimize the average substation utilization $\ell_{AVE} = \frac{1}{|S|} \sum_{i=1}^{|S|} \ell_i$. In general, ℓ_{AVE} may lead to lower performance user-substation assignment, since it provides no guarantee on the individual substation utilization. For example, by minimizing ℓ_{AVE}, some substations may be overloaded, while others may be significantly under-loaded. On the other hand, jointly optimizing both objectives ℓ_{MIN} and ℓ_{MAX} is promising in practice, which leads to an extension to our SPUA framework with multiple objectives (as discussed below).

Multiple design objectives. When solving the user-substation assignment problem, multiple design objectives may be considered. For example, in SPUA (discussed so far) maximum substation utilization ℓ_{MAX} is considered as the single objective, while the transmission loss as an objective is included as the constraint eq.(2). To capture multiple design objectives, for example, ℓ_{MIN}, ℓ_{MAX}, and transmission loss, our SPUA framework can be extended using a multi-objective bi-level optimization [24] technique. Bi-level optimization is a special optimization with one problem embedded within another, which thus can jointly optimize two problems simultaneously. To include multiple (more than two) objectives, each problem can take a linear combination of multiple objectives in the objective function. Clearly, when multiple objective functions are jointly considered, it is impossible to obtain a solution where all objectives are optimized. The Bi-level optimization employs a game-theoretical framework (a.k.a. Stackelberg game [24]) that leads to an equilibrium of all sub-problems.

7 Related Works

In the literature, we are the first to investigate the scalable user assignment problem in power grids using large scale

power consumption data. In this section, we discuss two topics that are closely related to our work: (1) data driven research for power grids, and (2) power grid planning.

Data Driven Research for Power Grids. Power grids generate large amount of big data from various sources, such as (1) energy consumption data measured by the smart meters, (2) energy market pricing and bidding data, (3) management, control and maintenance data for devices and equipment in the power generation, transmission and distribution networks. All of these heterogeneous power grid data enable intelligent solutions for various applications in power grids [2], [3], [4], [5]. For example, [2] explored temporal patterns in electricity consumption time-series data using a real-world, large-scale dataset and showed that usage behavior patterns can be identified at different times-of-day or days-of-the-week. [3] investigated how to classify household items such as televisions, kettles and refrigerators based only on their electricity usage profile. All these patterns arising from smart grid data can be used to smooth the profile of the existing peaks in the demand curve, or at least reduce the peak-to-average ratio. Moreover, to examine the energy consumption data to identify potential energy fraud, machine learning techniques were used to model consumers energy consumption behavior under normal conditions [4]. [5] employs energy sharing techniques to preserve user privacy. However, none of the existing works address the user assignment problem in power grid networks. In this work, we employ real power consumption data to identify and solve the issues with the current substation-user assignment.

Power Grid Planning. In the scope of power grid planning, the closest works to ours is the optimal substation planning, which involves substation site selection, substation size and service areas determination. In a classic reference, [25] presented a distribution substation planning model and a heuristic combinational optimization algorithm to solve the problem. In [26], the proposed planning problem was formulated as a Mixed Integer Linear Programming (MILP) problem, aiming at minimizing the total cost, subject to voltage drops and substation capacities. [27] proposed a mixed-integer linear programming approach to solving the optimal fixed/switched capacitors allocation problem in radial distribution systems with distributed generation. In this paper, we investigate the power substation-user assignment problem, which is a less studied topic in power grid planning. Due to the large number of decision variables, solving this user assignment problem in a centralized manner is not feasible, which motivates us to design a distributed optimization approach using block-splitting algorithm [12].

8 Conclusion

In this paper, we study the problem of how to judiciously assign each power user to a substation, such that the maximum substation utilization is minimized. We develop a data-driven scalable power user assignment (SPUA) framework that takes heterogeneous power grid data as inputs, including temporal power consumption data and spatial power user/substation distribution data, and performs optimal user assignment via a scalable distributed algorithm. To evaluate the performance of our SPUA framework, we conduct extensive evaluations using a large-scale power consumption data with user and substation locations. The evaluation results demonstrate that our SPUA framework can achieve a 20%–65% reduction on the maximum substation utilization, and 2 to 3.7 times reduction on total transmission loss over other realistic baselines. This observation motivates us to further investigate various power grid planning problems, including the power plant and substation deployment, as well as roll-out strategies of substation-user assignment. We leave these problems for our future work.

9 Acknowledgments

Yanhua Li was supported in part by NSF CRII grant CNS-1657350 and a research grant from Pitney Bowes Inc.

Appendix

Distributed optimization: To facilitate the understanding of distributed synthesis method developed in Section 4, we describe the ADMM [28] for the generic convex constrained minimization problem $\min_{z\in\mathbf{C}} g(z)$ where function g is closed proper convex and set $\mathbf{C}$ is closed nonempty convex.

The block splitting algorithm implemented to solve eq.(10) works as follows. Note that the subscripts i, j here are the indices of variables, not the user and station indices. Initialize all variables to zero vectors with proper dimensions at $t = 0$. At the t-th iteration, for $i = 1, \ldots, N, j = 0, \ldots, N$,

$$
\begin{aligned}
\mathbf{y}_i^{t+1/2} &:= \mathsf{prox}_{f_i}(\mathbf{y}_i^t - \tilde{\mathbf{y}}_i^t) = \mathbf{b}_i,\\
(\mathbf{x}_0^{t+1/2}, \mathbf{x}_j^{t+1/2}) &:= \mathsf{prox}_{g_j}(\mathbf{x}_0^t - \tilde{\mathbf{x}}_0^t, \mathbf{x}_j^t - \tilde{\mathbf{x}}_j^t),\\
&:= \mathsf{proj}_{g_j}(\mathbf{x}_0^t - \tilde{\mathbf{x}}_0^t, \mathbf{x}_j^t - \tilde{\mathbf{x}}_j^t),\\
(\mathbf{x}_{ij}^{t+1/2}, \mathbf{y}_{ij}^{t+1/2}) &:= \mathsf{proj}_{ij}(\mathbf{x}_j^t - \tilde{\mathbf{x}}_{ij}^t, \mathbf{y}_{ij}^t + \tilde{\mathbf{y}}_i^t),\\
\mathbf{x}_j^{t+1} &:= \mathsf{avg}(\mathbf{x}_j^{t+1/2}, \{\mathbf{x}_{ij}^{t+1/2}\}_{i=1}^N),\\
(\mathbf{y}_i^{t+1}, \{\mathbf{y}_{ij}^{t+1}\}_{j=0}^N) &:= \mathsf{exch}(\mathbf{y}_i^{t+1/2}, \{\mathbf{y}_{ij}^{t+1/2}\}_{j=0}^N),\\
\tilde{\mathbf{x}}_j^{t+1} &:= \tilde{\mathbf{x}}_j^t + \mathbf{x}_j^{t+1/2} - \mathbf{x}_j^{t+1},\\
\tilde{\mathbf{y}}_i^{t+1} &:= \tilde{\mathbf{y}}_i^t + \mathbf{y}_i^{t+1/2} - \mathbf{y}_i^{t+1},\\
\tilde{\mathbf{x}}_{ij}^{t+1} &:= \tilde{\mathbf{x}}_{ij}^t + \mathbf{x}_{ij}^{t+1/2} - \mathbf{x}_j^{t+1},
\end{aligned}
$$

where $\mathsf{prox}_{f_i}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left(f(\mathbf{x}) + (\rho/2)\|\mathbf{x} - \mathbf{z}\|_2^2\right)$ is the *proximal operator* of f_i with parameter $\rho > 0$ that enforces the constraints are satisfied, proj_{g_j} denotes the projection of non-slack decision variables in X_0 and X_j onto a probability simplex and the slack variables onto non-negative orthant, proj_{ij} denotes projection onto $\{(\mathbf{x}, \mathbf{y}) \mid \mathbf{y} = A_{ij}\mathbf{x}\}$, avg is the elementwise averaging [2]; and exch is the exchange operator, defined as below. $\mathsf{exch}(\mathbf{z}, \{\mathbf{z}_j\}_{j=1}^N)$ is given by $\mathbf{y}_{ij} := \mathbf{z}_j + (\mathbf{z} - \sum_{j=1}^N \mathbf{z}_j)/(N+1)$ and $\mathbf{y}_i := \mathbf{z} - (\mathbf{z} - \sum_{j=1}^N \mathbf{z}_j)/(N+1)$. Note that the computation in each iteration can be parallelized.

Stopping criterion. The algorithm takes parameters ρ, ϵ^{rel}, and ϵ^{abs}: $\rho > 0$ is a penalty parameter to ensure the constraints are satisfied, $\epsilon^{rel} > 0$ is a relative tolerance and $\epsilon^{abs} > 0$ is an absolute tolerance. The choice of ϵ^{rel} and ϵ^{abs}

2. Since for some i, j, $\mathbf{x}_{ij}^{t+1/2} = 0$, in the elementwise averaging, these $\mathbf{x}_{ij}^{t+1/2}$ will not be included.

depends on the scale of variable values. In our study, we used $\rho = 0.5$, $\epsilon^{abs} = \epsilon^{rel} = 10^{-4}$ for our SPUA method throughout our evaluations. The algorithm is ensured to converge with any choice of ρ and the value of ρ may affect the convergence rate.

At each iteration, we compute two values $r^{t+1} = z^{t+1/2} - z^{t+1}$ and $s^{t+1} = -\rho(z^{t+1} - z^t)$, where $z^* = (x^*, y^*)$ for $* \in \{t+1/2, t+1\}$. Variables r^{t+1} and s^{t+1} can be viewed as primal and dual residuals in the algorithm. The algorithm terminates when both residuals are small, i.e.,

$$\|r^{t+1}\| \leq \epsilon^{\text{pri}} \text{ and } \|s^{t-1}\| \leq \epsilon^{\text{dual}}$$

where ϵ^{pri} and ϵ^{dual} are tolerances that are pre-defined functions of an relative tolerance $\epsilon^{rel} > 0$ and an absolute tolerance $\epsilon^{abs} > 0$ using the method in [12](Section 3.2). The iteration terminates when the stopping criterion for the block splitting algorithm is met. The solution can be obtained $\mathbf{x}^* = (\mathbf{x}_0^{t+1/2}, \ldots, \mathbf{x}_N^{t+1/2})$.

REFERENCES

[1] B. Lyu, S. Li, Y. Li, J. Fu, H. Xie, and Y. Liao, "Scalable user assignment in power grids: A data driven approach," in *GIS'16: 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2016.

[2] C. Chelmis, J. Kolte, and V. K. Prasanna, "Big data analytics for demand response: Clustering over space and time," in *IEEE Big Data*. IEEE, 2015, pp. 2223–2232.

[3] J. Lines, A. Bagnall, P. Caiger-Smith, and S. Anderson, "Classification of household devices by electricity usage profiles," in *IDEAL*. Springer, 2011, pp. 403–412.

[4] V. Ford, A. Siraj, and W. Eberle, "Smart grid energy fraud detection using artificial neural networks," in *CIASG*. IEEE, 2014, pp. 1–6.

[5] Y. G. Zhichun Huang, Ting Zhu and Y. Li, "Shepherd sharing energy for privacy preserving in hybrid ac-dc microgrids," in *ACM e-Energy*, 2016.

[6] G. K. Heilig, "World urbanization prospects the 2011 revision," *United Nations, Department of Economic and Social Affairs (DESA)*, 2012.

[7] E. Davis and J. M. Jaffe, "Algorithms for scheduling tasks on unrelated processors," *JACM*, vol. 28, no. 4, pp. 721–736, 1981.

[8] J. K. Lenstra, D. B. Shmoys, and É. Tardos, "Approximation algorithms for scheduling unrelated parallel machines," *Mathematical programming*, vol. 46, no. 1-3, pp. 259–271, 1990.

[9] E. V. Shchepin and N. Vakhania, "An optimal rounding gives a better approximation for scheduling unrelated machines," *Operations Research Letters*, vol. 33, no. 2, pp. 127–133, 2005.

[10] L. Fanjul-Peyro and R. Ruiz, "Iterated greedy local search methods for unrelated parallel machine scheduling," *European Journal of Operational Research*, vol. 207, no. 1, pp. 55–69, 2010.

[11] R. E. Korf, "A new algorithm for optimal bin packing," in *AAAI/IAAI*, 2002, pp. 731–736.

[12] N. Parikh and S. Boyd, "Block splitting for distributed optimization," *Mathematical Programming Computation*, vol. 6, no. 1, pp. 77–102, 2014.

[13] W. S. Team, "Spua project website," http://urban.cs.wpi.edu/SmartGrid/index.html.

[14] ——, "Spua project code downloading site," http://urban.cs.wpi.edu/SmartGrid/index.html#vis.

[15] S. M. Mazhari and H. Monsef, "Dynamic sub-transmission substation expansion planning using learning automata," *Electric Power Systems Research*, vol. 96, pp. 255–266, 2013.

[16] C. E. Commission, "The value of distribution automation," Navigant Consulting, Inc., Tech. Rep., 2009.

[17] *Baidu Geocoding API*, http://http://lbsyun.baidu.com.

[18] *Google Geo-Coding API*, https://developers.google.com/maps/documentation/geocoding/start.

[19] *Benchmark of commercial LP solvers*, http://plato.asu.edu/ftp/lpcom.html.

[20] Wikipedia, "K-means clustering," https://en.wikipedia.org/wiki/K-means_clustering.

[21] W. Wang and M. Carreira-Perpinan, "Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application," *CoRR*, vol. abs/1309.1541, 2013.

[22] N. Parikh, S. P. Boyd *et al.*, "Proximal algorithms." *Foundations and Trends in optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[23] J. Fu, S. Han, and U. Topcu, "Optimal control in markov decision processes via distributed optimization," in *CDC*, 2015.

[24] Wikipedia, "Multi objective bilevel optimization," https://en.wikipedia.org/wiki/Bilevel_optimization.

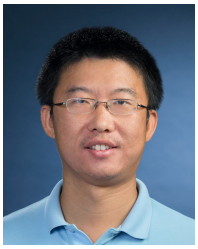
[25] H. Dai, Y. Yu, C. Huang, C. Wang, S. Ge, J. Xiao, Y. Zhou, and R. Xin, "Optimal planning of distribution substation locations and sizes—model and algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 18, no. 6, pp. 353–357, 1996.

[26] T. El-Fouly, H. Zeineldin, E. El-Saadany, and M. Salama, "A new optimization model for distribution substation siting, sizing, and timing," *International Journal of Electrical Power & Energy Systems*, vol. 30, no. 5, pp. 308–315, 2008.

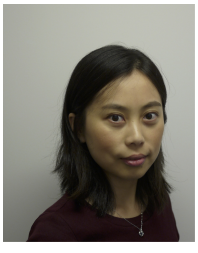
[27] J. F. Franco, M. J. Rider, M. Lavorato, and R. Romero, "Optimal allocation of capacitors in radial distribution systems with distributed generation," in *ISGT*. IEEE, 2011, pp. 1–6.

[28] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

Bo Lyu received Ph.D. degrees in information and communication from Beijing University of Posts and Telecommunications, Beijing in China in 2014. He works as a researcher in the Innovation Center of China Academy of Electronics and Information Technology. He was a visiting scholar in the Department of Computer Science at Worcester Polytechnic Institute (WPI) in Worcester, MA in 2016. His research interests are urban computing and data visualization.



Yanhua Li (S'09-M'13-SM'16) received two Ph.D. degrees in electrical engineering from Beijing University of Posts and Telecommunications, Beijing in China in 2009 and in computer science from University of Minnesota at Twin Cities in 2013, respectively. He has worked as a researcher in HUAWEI Noahs Ark LAB at Hong Kong from Aug 2013 to Dec 2014, and has interned in Bell Labs in New Jersey, Microsoft Research Asia, and HUAWEI research labs of America from 2011 to 2013. He is currently an Assistant Professor in the Department of Computer Science at Worcester Polytechnic Institute (WPI) in Worcester, MA. His research interests are big data analytics and urban computing in many contexts, including urban network data analytics and management, urban planning and optimization.



Jie Fu received the B.S. and M.S. degrees from Beijing Institute of Technology, Beijing, China, in 2007 and 2009, respectively, and the Ph.D. degree in mechanical engineering from the University of Delaware in 2013. She is currently an Assistant Professor at the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute. Her research interests include adaptive and optimal control, hybrid systems, and formal methods.

Andrew C. Trapp completed his PhD in Industrial Engineering from the University of Pittsburgh in 2001 being supported through a doctoral Graduate Assistance in Areas of National Need (GAANN) fellowship in sustainable engineering. He is presently an Assistant Professor of Operations and Industrial Engineering at Worcester Polytechnic Institute (WPI) in Worcester, MA. His research focus is on using advanced analytical techniques, in particular mathematical optimization, to find optimal decisions to problems arising from a diverse cross-section of sectors such as data mining, sustainability, humanitarian operations, and healthcare. He develops new theory, models, and computational solution approaches to tackle such problems. He has published in leading optimization journals such as Operations Research, European Journal of Operational Research, INFORMS Journal on Computing, Annals of Operations Research, IIE Transactions, and Discrete Optimization.

Haiyong Xie received his Ph.D. and M.S. degree in Computer Science from Yale University in 2008 and 2005, respectively, and his B.S. degree from USTC in 1997. He is now the executive director for the Cyberspace and Data Science Laboratory, China Academy of Electronics and Information Technology. His research interest includes cyber-physical systems, content-centric networks, software-defined networks, network data analytics.

Yong Liao received his B.S. degree in 2001 from University of Science and Technology of China, his M.S. from Chinese Academy of Sciences in 2004, and Ph.D. from University of Massachusetts at Amherst in 2010. After graduating from UMass Amherst, he was Software Engineer at Microsoft Redmond, Washington; Senior Member of Technical Staff in the CTO group of Narus Inc. (a Boeing company), Sunnyvale, California; and Senior Principal Data Scientist at Symantec Corp., Mountain View, California. Currently he is a Principal Research Scientist at China Academy of Electronics and Information Technology, Beijing, China. His research interests include big data framework and analytics, network forensics, security and privacy.