Algorithms for Fair Load Shedding in Developing Countries

Olabambo Ifeoluwa Oluwasuji, Obaid Malik, Jie Zhang and Sarvapali Dyanand Ramchurn
School of Electronics and Computer Science, University of Southampton, Southampton, UK
{oio1a14,o.malik,jie.zhang,sdr1}@soton.ac.uk

Abstract

Due to the limited generation capacity of power stations, many developing countries frequently resort to disconnecting large parts of the power grid from supply, a process termed load shedding. During load shedding, many homes are left without electricity, causing them inconvenience and discomfort. In this paper, we present a number of optimization heuristics that focus on pairwise and group-wise fairness, such that households (i.e. agents) are fairly allocated electricity. We evaluate the heuristics against standard fairness metrics in terms of comfort delivered to homes, as well as the number of times they are disconnected from electricity supply. Thus, we establish new benchmarks for fair load shedding schemes.

1 Introduction

Load shedding entails systematically and deliberately cutting off the supply to parts of the system network, so that the strain on the system is reduced and the failure of the entire system is prevented. It is very common in developing countries, especially because generation capacity is insufficient for meeting demand, and grid infrastructure is poorly maintained or obsolete. In Nigeria for instance, the total installed capacity of generating plants is under 8000MW, which is grossly inadequate for serving a population of over 170 million people [Oyedepo, 2012]. This suggests that there is a perpetual need for implementing load shedding to maintain grid stability in some developing countries. On the other hand, energy demand is increasing globally [Mahadevan and Asafu-Adjeay, 2007]. As such, load shedding will be a prevalent problem that will be relevant for the near future. It is noteworthy that the availability of electricity presents a platform for fighting poverty, improving the welfare of individuals and progressing in development. Therefore, while on the long road towards increasing generation capacity, it is absolutely necessary to develop solutions for managing load shedding events better.

Currently, solutions for load shedding involve selecting parts of the system whose consumption closely matches the deficit (i.e. the difference between electricity available for supply, and demand), then disconnecting these parts from electricity supply. For instance, artificial neural networks were used for determining the minimum amount of load to be shed in order to maintain grid stability [Mitchell et al., 2000; Hsu et al., 2005]. Following this, parts of the system that make up this load were disconnected at the supply level. By determining the minimum amount of load shedding required for maintaining grid stability, they resulted in load shedding being more optimal. However, these current approaches do not give due consideration to how fair selection processes are. As such, as far as load shedding results in the stability of the network, no priority is given to ensuring that all parts of the power system benefit from electricity allocation as equally as possible. A consequence of this is that many homes within some parts of the system may be left without supply for days or weeks. Moreover, for electricity providers, load shedding may result in revenue loss, as more load than is required may be shed when disconnecting parts of the system from supply.

In light of the above, we present a novel approach to load shedding. Our approach models homes as agents, each with its own preferences for consuming energy. We build this approach on the back of intensive and extensive research into fair-division allocation of heterogeneous resources, often termed as “cakes”, for which agents have different and conflicting interests [Varian, 1974; Robertson and Webb, 1998; Moulin, 2003; Brânzei et al., 2013]. In the research area, the objective is to allocate these heterogeneous resources to agents in a fair manner, while maximizing social welfare. As such, our model attempts to manage load shedding so that electricity is fairly allocated across agents, hence maximizing the access to electricity. In addition, our model attempts to increase revenue for suppliers through allocating electricity to agents individually. Our work advances the state of the art as follows:

1. We perform a comparative analysis on four different heuristics which consider varying, and sometimes conflicting fairness criteria. These fairness criteria include the number of times each agent is disconnected, the discomfort inflicted on individual agents by being disconnected, the number of agents disconnected and the comfort costs incurred by the power system.
2. By combining multiple sources of data, we create a dataset relevant to Nigeria from disaggregated electricity consumption data collected from Pecan Street’s Dataport1.
3. Using the data described above, we evaluate our load shed-

---
1Dataport is the largest provider of disaggregated customer energy data [Parson et al., 2015]
ding algorithms and show how they perform in optimizing utilitarian and egalitarian social welfare, as well as in minimizing envy (defined in Section 4).

Taken altogether, our heuristic algorithms establish a novel approach to load shedding algorithm design, and establish fairness benchmarks for such algorithms.

The rest of the paper is organized as follows. In Section 2, we show how we create a relevant dataset by combining multiple sources of data. Section 3 presents four heuristic household load shedding algorithms, while Section 4 analyzes the performance of the algorithms against some standard fairness metrics. Section 5 concludes.

2 Simulating Developing Country Energy Consumption Data

In order to implement and evaluate our fair load shedding schemes, we first focus on developing a realistic simulation of energy consumption that can be attributed to homes in developing countries. In particular, we focus on domestic consumption in Nigeria, where the residential sector accounts for 51.3% of consumption [Nwachukwu et al., 2014]. Because consumption data of households is not currently available for households in Nigeria as well as in most, if not all African countries, we collect readily available household consumption data of households in the USA, and adapt it to the Nigerian context. We do this based on some identified similarities between how electricity is consumed in households within both countries.

In 2010, the average consumption of an electrified household in Nigeria was 570kWh. In contrast, 11,698kWh of electricity was consumed by a USA home in the same year. A reason for the wide contrast is the difference between the average temperature of the two countries. The temperature in the USA is such that, on average, a home in the country expends energy on heating. For instance, in 2010, 41.5% of the average electricity consumed within a home in the USA was expended on heating, while 17.7% was expended on water heating. In turn, only about 16% of the average electricity consumed within homes in Nigeria is expended on cooling [Yohanna et al., 2013]. Another reason for the difference in the average electricity consumed between households in both countries is that, on average, homes in Nigeria are poorer than those in the USA. This factor directly impacts on the appliances used within a home.

Thereupon, based on disaggregated data available on Dataport, we consider the appliances commonly used in a typical home in the USA and in Nigeria. From Dataport, Figure 1 typifies a representation of the number of occurrences of each appliance category across over 700 households in the USA [Parson et al., 2015]. Conversely, studies show that the appliances typically available in an electrified home in Nigeria include lighting, televisions, electric fans, DVD players, washing machines, electric irons, air conditioners, refrigerators, sewing machine and water pumps [Oji et al., 2012; Salmon and Tanguy, 2016; Emodi et al., 2017; Monyei et al., 2018]. Hence, we extract the data for the appliances that are common to both countries from Dataport. These include the data for air conditioners, washing machines, light bulbs and refrigerators. Likewise, to factor in weather considerations, we collect the usage data of these appliances over 13 weeks of the summer (from the first week in June to the last week in August) for households in only Austin, Texas, one of the warmest states in the USA. This results in the data for 372 households. We aggregate these to make up overall household electricity consumption for each of these households. In the next section, we present four heuristic household load shedding algorithms.

3 Managing Loads at the Household Level

Our approach to load shedding proposes that shedding be planned ahead. With estimates of electricity available for supply from energy suppliers (i.e. operators) and day-ahead predictions of household consumption, information that is useful for planning load shedding a day ahead is available. Additionally, our approach proposes that load be shed at the household level, rather than at substation level, where parts of the grid are disconnected from supply. This ensures that revenue is maximized for the operator, and electricity that may be wasted is instead supplied to agents. As an example, suppose the deficit is 100kWh and a part being shed constitutes a load of 150kWh. Then, a 50kWh load that would otherwise have produced revenue for the supplier and distributed among agents is lost. In contrast, if load is shed at the household level, a far more closer match to the deficit can be consistently achieved.

Notably, our approach is based on previous research [Keelson et al., 2014; Azasoo and Boateng, 2015], where smart retrofitted household electric meters were designed for use in developing countries. The retrofits employ GSM (Global System for Mobile communication) modules as a medium of connection between individual meter and operator. This not only provides individual meters the ability for transmitting usage data securely, but also the ability for being remotely disconnected and re-connected. The retrofits are also inexpensive (less than $10 per unit if mass produced) and can be installed on existing meters. Based on these studies, we assume household-level load control. To this end, in the remain-

---

2Nigeria’s energy situation is representative of challenges in Africa.

3See also https://goo.gl/1CheJe

---

Figure 1: Number of appliance occurrences in households (Dataport).
Algorithm 1: Grouping, then selecting a group of agents for disconnection (i.e. Grouper Algorithm).

Input : $H, l^t, C, G, L$
Output: $S^t$

1. for EACH HOUR $t \in \{1, 2, \ldots, 24\}$ do
   2. if $d^t > 0$ then
      3. $C = H$; \hfill // Set of agents available
      4. $L = L^t$; \hfill // Consumption of all agents in $C$
      5. $S^t = P = {}$; \hfill // Create empty sets
      6. $L^* = 0$; \hfill // Initialize variable
      7. while $L^* > d^t$ do
         8. random.sample $h_i \in C$; \hfill // Random selection
         9. $S^t$.add($h_i$); \hfill // Add agent to $S^t$
         10. $C$.remove($h_i$); \hfill // Remove agent from $C$
         11. $L^* = L^* + l^t_i$; \hfill // Update consumption sum
         12. $L^* = L^* - l^t_i$; \hfill // Update consumption sum
      13. if $L^* > d^t$ then
         14. $P$.add($S^t$); \hfill // Add set to set of sets
         15. $S^t = {}$; \hfill // Remove all agents in $S^t$
      16. compute($N_S \forall S^t \in P$); \hfill // Sum of selected times
      17. houses.shed($S^t \min N_S$); \hfill // Shed set with minimum

Algorithm 2: Using consumption to select agents for disconnection while minimizing the difference in the number of times all agents are selected (i.e. Consumption-Sorter Algorithm).

Input : $H, l^t, C, G, L$
Output: $S^t$

1. $C = H$; \hfill // Set of agents available for shedding
2. for EACH HOUR $t \in \{1, 2, \ldots, 24\}$ do
   3. if $d^t > 0$ then
      4. $S^t = {}$; \hfill // Create empty set
      5. $L^* = 0$; \hfill // Initialize variable
      6. $C$ descend$\_sort(l^t_i)$; \hfill // Sort using consumption
      7. while $L^* < d^t$ do
         8. select$\_i$th $h_i \in C$; \hfill // Select in order
         9. $S^t$.add($h_i$); \hfill // Add agent to $S^t$
         10. $C$.remove($h_i$); \hfill // Remove agent from $C$
         11. $L^* = L^* + l^t_i$; \hfill // Update total consumption
         12. if $C = {}$ then
            13. $C = H - S^t$; \hfill // Repopulate set
      14. houses.shed($S^t$); \hfill // Shed selected agents

In Algorithm 2, all agents in the population are made available for selection before the first shedding event (Line 1). Then, whenever there is a deficit, load shedding action is initiated (Line 2-3). At the beginning of every shedding event, an empty set $S^t$ to be populated with agents that will be disconnected is created (Line 4). A variable that represents the sum of the consumption of agents in $S^t$ is also initiated (Line 5). The selection process aims to pick agents in a decreasing order of their consumption. As such, agents are sorted in this order (Line 6). Agents are added one after the other into $S^t$, until the sum of consumption of agents in $S^t$ is enough to off-
set the deficit (Line 7-13). These agents are selected in order from the set \( C \) (Line 8), added into \( S \) (Line 9) and removed from \( C \) (Line 10). Every time an agent is selected, the sum of consumption of agents in \( S \) is updated (Line 11). Set \( C \) is re-populated with agents in the entire population \( H \) that have not already been selected, if it becomes empty in the middle of a selection process (Line 12-13). After the selection process, the agents in \( S \) are disconnected from electricity supply.

The Consumption-Sorter Algorithm attempts to maintain the similarity between the number of times households were disconnected. However, the algorithm selected agents in order of their consumption. The next algorithm is designed to be agnostic to the consumption of the agents.

**Random-Selector Algorithm**
The Random-Selector heuristic differs from the Consumption-Sorter heuristic in that it does not arrange agents based on their consumption, and so does not select agents for disconnection in any particular order. Instead, in an attempt to avoid a bias based on consumption, it randomly selects agents for disconnection during shedding events. The heuristic algorithm is described in Algorithm 3.

**Algorithm 3: Selecting agents to shed while keeping the similarity between number of times all agents are selected (i.e. Random-Selector Algorithm).**

**Input**: \( H, I^t, G, L \)  
**Output**: \( S^t \)

1. \( C = H \);  
   \( // \) Set of agents available for shedding
2. **for each hour** \( t \in \{1, 2, \ldots, 24\} \)** do
3.   \( if \ d^t > 0 \) then
4.     \( S^t = \{\} \);  
5.     \( L^* = 0 \);  
6.     **while** \( L^* < d^t \) do
7.      \( random.sample \ h_i \in C \);  
8.      \( S^t.add(h_i) \);  
9.      \( C.remove(h_i) \);  
10.     \( L^* = L^* + l_i^t \);  
11.    **if** \( C = \{\} \) then
12.       \( C = H - S^t \);  
13.       \( houses.shed(S^t) \);  
14. **end**
15. **end**
16. **output**: \( S^t \)

As aforementioned, agents are selected randomly from \( C \) (Line 7). Otherwise, the description of the algorithm is similar to that of Algorithm 2.

Inasmuch as the Grouper, Consumption-Sorter and Random-Selector algorithms have, to some extent, been fair when selecting agents, they have not directly considered the comfort costs of agents when making these selections. The next heuristic aims to factor this into consideration.

**Cost-Sorter Algorithm**
The fourth heuristic uses comfort costs \( I \) (defined in Section 4.1) in selecting agents for shedding. The heuristic aims to select agents with the least comfort costs, while maintaining a parity between the number of times agents are disconnected. It is described in Algorithm 4.

In contrast to the heuristics implemented using Algorithms 2 and 3, the comfort costs of agents for the hour of load shedding are collected (Line 4). In addition, agents are sorted in an increasing order of comfort costs (Line 7). Then, they are selected for disconnection in this order (Line 9). Otherwise, the description of the algorithm is also similar to that of Algorithm 2.

Summarily, the Grouper Algorithm randomly selects households into different groups until the aggregated consumption of the households in each group is just enough to offset the deficit, then disconnects the group with the least total number of disconnections from supply. The Consumption-Sorter, Random-Selector and Cost-Sorter algorithms keep the number of times agents are disconnected as close as possible by using a queuing system to exempt a selected agent from disconnection, until all other agents have been disconnected after it. While the Consumption-Sorter Algorithm creates an order of selection using consumption, the Cost-Sorter Algorithm does the same using cost. The Random-Selector randomly selects in no particular order. We evaluate these heuristics in the section that follows.

## 4 Performance Evaluation of Heuristic Algorithms

In this section, we assess the heuristics in Section 3.1 against some fairness criteria that include the number of times each agent is disconnected, the number of agents disconnected, the individual discomfort inflicted on disconnected agents and the aggregated comfort cost incurred by the system. We do this using the utilitarian, egalitarian and envy-freeness objectives of economic model designs (defined within this section in Section 4.2). The heuristics are evaluated using the data obtained in Section 2. We begin by defining and formulating comfort costs for each agent.

### 4.1 Formulation of Comfort Costs

We define the comfort costs of agents using the amount of electricity they consume within each hour of a week, with respect to other hours of a week. We begin by learning
each agent’s normal consumption during each hour of the week. We do this based on the premise that the correlations between electricity consumed on unique days of the week are stronger than those over different days of a week over the same season, as opined by [Truong et al., 2013; Do et al., 2016]. For this reason, we assume that an agent’s consumption over a week in the same season fully represents the agent’s consumption pattern in that season. We learn each agent's normal consumption pattern from historical data of every four prior weeks, so that any changes in consumption patterns is accounted for. Thus for an agent, $h_i$, the normal weekly consumption pattern, $Z_i$, is a vector computed using the following equation.

$$Z_i = \left( \frac{1}{4} \sum_{j=1}^{4} l_{i,j=1}, \ldots, \frac{1}{4} \sum_{j=1}^{4} l_{i,j=168} \right)$$ (1)

Here, $j$ is the number of weeks. Thereafter, we normalize the vector $Z_i$, so that the normal consumption pattern of all agents falls within the range $(\epsilon, 1)$. This forms a vector of comfort costs $I_i$ for each agent. The vector provide two benefits. Firstly, they create a platform on which all agents’ consumption patterns can be uniquely quantified, without considering how much electricity the agent consumes with respect to others. The second benefit is an extension of the first, in that comparison between agents becomes possible because all agent’s comfort costs are on the same scale. Given this, we define the comfort cost, $I_i$, for an agent, $h_i$, as:

$$I_i = \max_t \left\{ \frac{Z_i}{\mathbb{E} \{ Z_i \} } \right\} = (\delta_{i=1}, \ldots, \delta_{i=168})$$ (2)

Additionally, we assume that $\delta_i$ is a cost incurred by the system in supplying a agent, $t$, or the discomfort caused an agent when disconnected at time, $t$ (i.e. $n_i \delta_i$).

4.2 Fairness Objectives Based on Comfort Costs

The computed comfort costs serve as tools for assessing our heuristics against a set of objectives. Some predominant objectives in economic model design are the utilitarian, egalitarian and envy-freeness objectives [Mas-Colell et al., 1995; Leite et al., 2009]. Specifically, [Leite et al., 2009] define the utilitarian objective as the sum of individual utilities of agents. In our domain, we use the comfort costs of agents in calculating these utilities. In calculating these utilities, we consider chore division. Chore division is a dual version of the cake-cutting problem in which the divided resource is undesirable, so that each agent wants to get as little as possible [Petersen and Su, 2009; Dehghani et al., 2018]. In this regard, for all times an agent is disconnected during $k$ number of load sheds, an agent’s negative utility is $u_\delta = \sum_{k=1}^{k} \delta_i^k$, where $\delta_i$ is the comfort cost of the agent during the hour of shedding event $k$. To capture the performance of the heuristics for the whole system, utilitarian social welfare is defined as the addition of aggregated discomfort for $n$ agents, $\sum_{i=1}^{n} \delta_i$, where $\delta_i = \sum_{k=1}^{k} \delta_i^k$. Conversely, [Leite et al., 2009] define the egalitarian objective as the utility of the agent that is currently worst off. In our domain, we adopt the egalitarian criterion as the highest individual comfort cost incurred by the system (or highest aggregated negative utility), as defined by $g_\delta = \max_i \delta_i$. In addition, envy-freeness is a criterion of fair division that allocates resources to agents in such a way that no agent envies the allocation of another. However, agents do not have information of the allocation to others within our domain. For this reason, we adapt envy-freeness in terms of measuring the maximum difference between the comfort allocated to all pairs of agents (or maximum difference between aggregated negative utilities), as defined by $g_\delta = \max_i \{ |\delta_i - \delta_j| \}$. A fair load shedding scheme should result in the lowest possible $g_\delta$, so that if the all agents were aware of all allocations, the aggregate envy will be minimal.

Table 1 compares the utilitarian, egalitarian and envy-freeness objectives, based on comfort costs. The Grouper algorithm produces the maximum envyness, but performs admirably under the utilitarian approach to social welfare. The Random-Selector algorithms performs second best under the envy-freeness and egalitarian objectives, but generates the highest utilitarian value. The Cost-Sorter algorithm does not fulfill any of the objectives better than all others.

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Utilitarian</th>
<th>Egalitarian</th>
<th>Envy-freeness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grouper</td>
<td>49047.96</td>
<td>356.30</td>
<td>209.28</td>
</tr>
<tr>
<td>Consumption-Sorter</td>
<td>48830.27</td>
<td>174.37</td>
<td>126.29</td>
</tr>
<tr>
<td>Random-Selector</td>
<td>53072.97</td>
<td>192.04</td>
<td>137.77</td>
</tr>
<tr>
<td>Cost-Sorter</td>
<td>52803.38</td>
<td>208.69</td>
<td>149.07</td>
</tr>
</tbody>
</table>

4.3 Fairness Objectives Based on Number of Times Agents are Disconnected

In this section, we compare the heuristic algorithms based on the number of times each agent is disconnected. The utilitarian, egalitarian and envy-freeness objectives are adopted herein based on chore division also. In Section 3.1, we defined $n_i$ as 1, if an agent is disconnected at hour $t$, and $N_i$ as the aggregated number of times each agent is disconnected. As such, the utilitarian approach is described as $u_N = \sum_{i=1}^{n} N_i$ in this case. Conversely, the egalitarian approach is described as $g_N = \max_i \{ N_i \}$. Finally, for envy-freeness, we employ the definition $g_N = \max_i \{ |N_i - N_j| \}$. As with the comfort costs, any fair shedding scheme should aim to minimize these values.

As seen in Table 2, the Grouper Algorithm fails to outperform others under the egalitarian objective of social welfare and in envy-freeness. The disparity between the number of times agents are disconnected is suggested by its high envy-freeness (in Table 1). However, it fulfills the utilitarian objective better than the Random-Selector and Cost-Sorter algorithms. On the other hand, the Cost-Sorter algorithm achieves its design purpose, as it causes the least discomfort for each household disconnected (as seen in Table 3). Because of the omission technique used within the algorithms, the difference between the number of times all agents are disconnected.

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Utilitarian</th>
<th>Egalitarian</th>
<th>Envy-freeness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grouper</td>
<td>78159</td>
<td>317</td>
<td>180</td>
</tr>
<tr>
<td>Consumption-Sorter</td>
<td>74857</td>
<td>202</td>
<td>1</td>
</tr>
<tr>
<td>Random-Selector</td>
<td>86307</td>
<td>233</td>
<td>1</td>
</tr>
<tr>
<td>Cost-Sorter</td>
<td>95538</td>
<td>257</td>
<td>1</td>
</tr>
</tbody>
</table>
4.4 Other Performance Considerations

As stated in the introduction, an efficient load shedding scheme can be described as one that sheds enough load to offset the deficit, yet minimizes the difference between the deficit and the load shed. All four heuristics work by selecting agents one after the other, until the sum of consumption of the selected agents is enough to offset the deficit. Therefore, to take a closer look at the difference between the loads cut and the deficits, we present the results obtained by the heuristic used as a baseline (i.e. Grouper algorithm) for the first 50 shedding events. Figure 2 shows that the loads cut match the deficits closely. This is because shedding load at the household level gives the heuristics a finer control over the amount of load to be shed, resulting in closer match to the deficit. With regards to this, there is the suggestion that this class of heuristics is efficient.

Another consideration is to have a similar proportion of the population of agents shed, based on a factor of the number of agents cut per kWh deficit. That is, if one agent is disconnected when there is a deficit of 1 kWh, 100 agents should be disconnected when the deficit is 100 kWh. Of course, this factor will not be constant because consumption typically differs over each hour in a day. However, it is desirable that this proportion be similar, so that as much as is possible, an equal proportion of the grid that depends on the deficit is disconnected at each shedding event. Therefore, we compare the proportion of the population of agents shed by all heuristics during the first 50 individual shedding events in Figure 3.

As seen in Figure 3, the Grouper algorithm produces the best ratios of agents disconnected to kWh load shed. Conversely, the Consumption-Sorter and Cost-Sorter algorithms produce the most varying effects on the proportion of agents disconnected. This is because the algorithms select in different orders, but maintain the similarity between the number of times agents are selected. The Consumption-Sorter algorithm selects in order of reducing consumption, while the Cost-Sorter selects in order of increasing cost. However, because they both maintain the similarity between number of times agents are shed, the number of agents they select to offset the deficit during shedding events differ more.

It is noteworthy that we do not aim to determine which heuristic is best, but to present a number of heuristics that can be employed within different environments and conditions. For example, an environment that aims to disconnect a regular number of households during load shedding events may employ the Grouper or Random-Selector heuristic, while another that wants to disconnect as few houses as possible during every load shedding event may implement the Consumption-Sorter or Cost-Sorter. In addition, some components of these heuristics may be combined to a desired effect, as drawn in our conclusions below.

5 Conclusions

This paper proposed a new approach to load shedding, and presented four heuristic algorithms for shedding load at the household level. Results obtained from the implementation of the heuristics showed the extent to which they satisfied predefined fairness objectives. Although none of the heuristics fulfilled all objectives better than the other, each of them produced some desirable effects. As this is a gap in literature, the proposed class of heuristics can serve as a benchmark for designing load shedding algorithms in the future, and some qualities of individual heuristics can be adapted into designs that suit different environments, based on the desired objectives. Likewise, the heuristics can serve as a benchmark in designing solutions for allocating other scarce resources (e.g. water allocation problems addressed by [Read at al., 2014; zhen Song et al., 2016]). For future work, the fair load shedding problem can be modelled as a goal programming problem where the social welfare objectives modelled in Section 4 are used as objective functions, with constraints dependent on the system’s characteristics.
References


