Appliance Scheduling as a QUBO Problem
(v2: Upgraded Formulation)

Rudrax

November 1, 2025

1 Our Objective

Our goal is to create a scheduling algorithm for a set of deferrable household appliances over a 24-hour period. This schedule must achieve two objectives:

1. Minimize the total electricity cost for the household.
2. Minimize the peak energy demand on the grid, while maximizing the use of on-site renewable energy [1].

Minimizing the peak *net* load is crucial for grid stability and aligns perfectly with the hackathon's "Demand Response" theme.

2 Input Data & Parameters (The "Givens")

We'll use data from January 1, 2024, provided by the hackathon organizers [2]. The data is in 15-minute intervals, so a 24-hour period gives us $T = 96$ time steps (from $t = 0$ to $t = 95$). From this dataset, we will extract three key arrays of length 96:

- **Baseline Load (B_t):** This is the non-deferrable power consumption of the house. We'll use the *Power Consumption (kW)* column.
- **Energy Price (C_t):** This is the cost per kWh at each time step. We'll use the *Electricity Price (USD/kWh)* column.
- **Renewable Production (R_t):** This is the on-site, locally generated power. We define it as the sum of *Solar Power (kW)* and *Wind Power (kW)* [2].

3 Deferrable Loads (The "Problem Space")

The dataset provides the grid's state, not the appliances we need to schedule. We must define these ourselves.

Let's define $N = 3$ appliances for our proof-of-concept:

1. **Appliance 1: EV Charger**
   - Power Draw (P_1): 9 kW

- Duration (L_1): 16 steps (4 hours)
- Window: 10:00 PM - 6:00 AM (e.g., $t = 88$ to $t = 95$ and $t = 0$ to $t = 23$)

2. **Appliance 2: Washing Machine**
   - Power Draw (P_2): 1.5 kW
   - Duration (L_2): 4 steps (1 hour)
   - Window: 8:00 AM - 10:00 PM (e.g., $t = 32$ to $t = 87$)

3. **Appliance 3: Dishwasher**
   - Power Draw (P_3): 1.8 kW
   - Duration (L_3): 6 steps (1.5 hours)
   - Window: 8:00 PM - 5:00 AM (e.g., $t = 80$ to $t = 95$ and $t = 0$ to $t = 19$)

4 Decision Variables (The "Knobs")

This is the core of our quantum model. We need to decide when each appliance starts. Let's define a set of binary decision variables $x_{i,t}$:

- $x_{i,t} = 1$ if appliance i starts running at time step t.
- $x_{i,t} = 0$ otherwise.

We only consider $x_{i,t}$ where t is a valid start time within the appliance's allowed window.

5 The QUBO Formulation (The "Math")

Our goal is to build a cost function (Hamiltonian) H that, when minimized, gives us the best schedule. We'll formulate this as a QUBO (Quadratic Unconstrained Binary Optimization) problem [3].

The total Hamiltonian is a weighted sum of three objective terms. Note: This (v2) formulation corrects a notational redundancy from our first draft, as pointed out by our mentor. The coefficients A, B, C are applied only in the final sum.

$$H = A \cdot H_{\text{cost\_term}} + B \cdot H_{\text{peak\_term}} + C \cdot H_{\text{constraint\_term}}$$

Here's how we build each part:

5.1 Cost Term ($H_{\text{cost\_term}}$)

Minimize the cost of running our deferrable appliances. First, we pre-calculate the cost $C_{i,t}$ of starting appliance i at time t:

$$C_{i,t} = P_i \cdot \sum_{k=t}^{t+L_i-1} C_k$$

The total cost objective is a linear sum over our decision variables:

$$H_{\text{cost\_term}} = \sum_{i,t} (C_{i,t} \cdot x_{i,t})$$

5.2 Peak Load Term ($H_{\textbf{peak\_term}}$)

This term is upgraded based on mentor feedback to explicitly integrate renewable energy. Instead of minimizing the *total* load, we minimize the *net load* (consumption minus renewable production). This directly incentivizes the algorithm to run appliances when renewable energy is abundant [1].

Total deferrable load at time t is $L_{\text{deferrable}}(t)$:

$$L_{\text{deferrable}}(t) = \sum_i \left(P_i \cdot \sum_{k=t-L_i+1}^{t} x_{i,k} \right)$$

The **Net Load** at time t, $L_{\text{net}}(t)$, is the total consumption (Baseline + Deferrable) minus the renewable production (R_t):

$$L_{\text{net}}(t) = (B_t + L_{\text{deferrable}}(t) - R_t)$$

The peak penalty term minimizes the square of this net load:

$$H_{\text{peak\_term}} = \sum_t \left(L_{\text{net}}(t)^2 \right) = \sum_t \left((B_t + L_{\text{deferrable}}(t) - R_t)^2 \right)$$

Expanding this $L_{\text{deferrable}}(t)^2$ term provides the quadratic $x_{i,k} \cdot x_{j,l}$ terms essential for a QUBO.

5.3 Constraint Term ($H_{\textbf{constraint\_term}}$)

Each appliance must run exactly once. For each appliance i, the sum of its start-time variables must be 1:

$$\sum_t x_{i,t} = 1$$

We enforce this with a squared penalty term [3]:

$$H_{\text{constraint\_term}} = \sum_i \left(\left(\sum_t x_{i,t} \right) - 1 \right)^2$$

5.4 Final Formulation (The Complete Model)

Our complete cost function to minimize is the weighted sum of these three terms, where A, B, C are the hyperparameters we will tune:

$$H = A \cdot H_{\text{cost\_term}} + B \cdot H_{\text{peak\_term}} + C \cdot H_{\text{constraint\_term}}$$

This QUBO structure is perfectly suited for the Quantum Approximate Optimization Algorithm (QAOA) [4].

References

[1] Hackathon Organizers (2025). *Challenge Title: Optimizing Energy Demand Response.* Official Hackathon Documentation.

[2] Hackathon Organizers (2025). *smart_grid_dataset_2024.csv*. Provided Hackathon Dataset.

[3] Glover, F., Kochenberger, G., & Du, Y. (2019). A Tutorial on Formulating and Using QUBO Models. *arXiv preprint arXiv:1811.11538*.

[4] Farhi, E., Goldstone, J., & Gutmann, S. (2014). A Quantum Approximate Optimization Algorithm. *arXiv preprint arXiv:1411.4028*.

[5] Energy Storage Scheduling: A QUBO Formulation for Quantum Computing. In *2021 IEEE International Conference on Environment and Electrical Engineering (EEEIC)*. DOI: 10.1109/EEEIC50896.2021.9584518.