

The key points of the methodology used in the modelling of energy arbitrage using BESS are as follows:

1. The battery optimization model maximizes profits by managing charging and discharging activities across the 3 electricity markets. Each market has distinct variables for charging and discharging (thereby making total 6 variables) to respond to the market's pricing signals.
2. Pyomo is used to construct the model, with Bonmin utilized as the solver, which can handle MINLP models.
3. To mitigate the issue of battery degradation, cycle counting is implemented within the model. This method tracks transitions/difference in a binary variable that flips between 0 (charging) and 1 (discharging) of the BESS.
4. Every switch in the binary state from charging to discharging or vice versa is counted. Frequent cycling can accelerate battery wear, thus a weighted objective is created to reduce these cycles.
5. Big M constraints along with binary variables are used to prevent charging and discharging of the BESS in the 3 markets at the same time.
6. A specific constraint enforces uniform charging and discharging rates within Market 3 throughout the day, which reflects the market with fixed daily pricing.
7. Detailed results of the model's application over three years are given in an Excel file, ([Results.xlsx](#)) showing revenues and costs for each half-hour interval. There are also visual plots shown for the financial performance and the impact of degradation strategies on BESS operation (*Figures are for first 10 days or 480 time-slots only*).
8. There is a comparative analysis of two cases, one without considering cyclic degradation and another with cyclic degradation included in the objective function. It demonstrates a significant impact on profitability where including degradation considerations lowers immediate revenue but substantially prolongs battery life.
9. The arbitrage revenue without degradation considerations is around GBP 150,000 for all the years (2018-2020) combined.
10. A public GitHub repository (https://github.com/bakulkandpal/BESS_arbitrage) gives the code model.py with the optimization model.