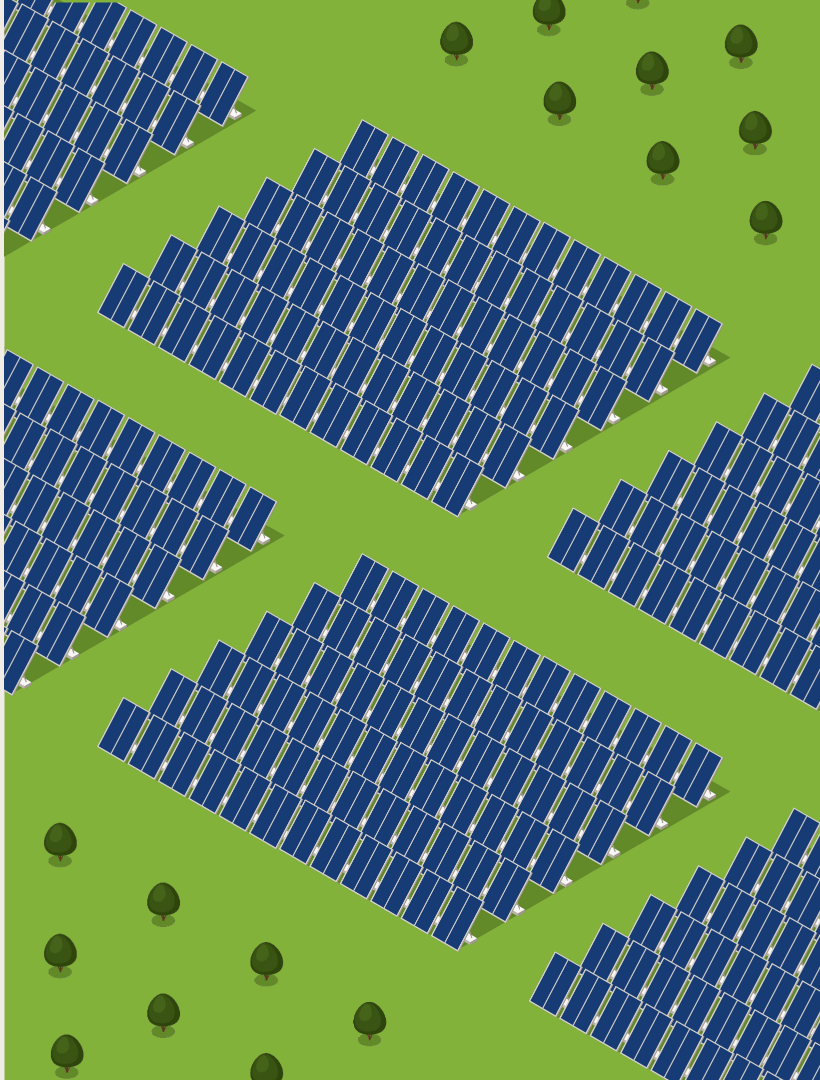
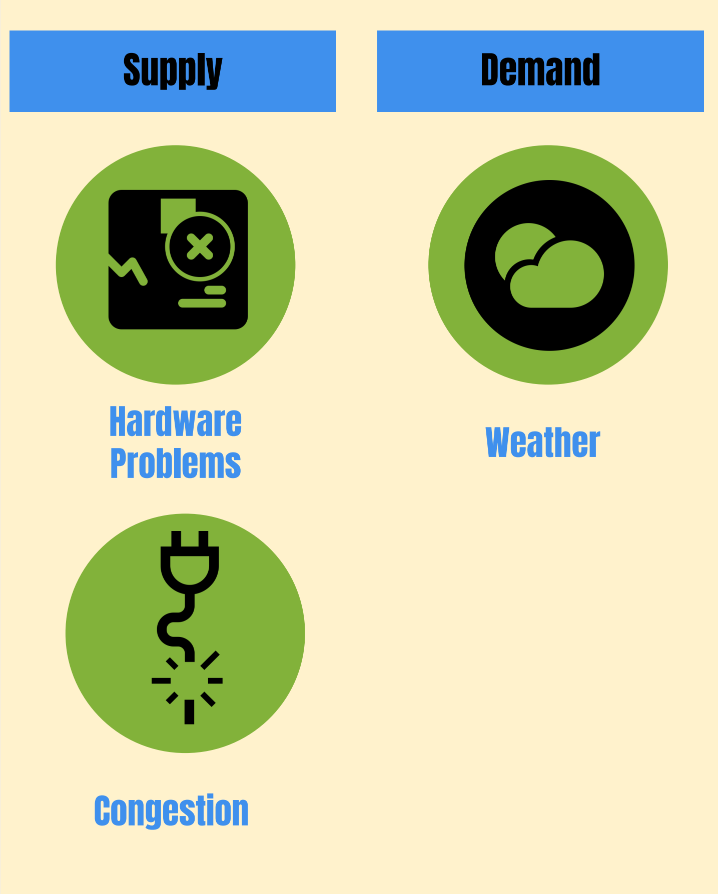
Real Time Arbitrage of Energy Prices with a Battery Network

By Colton Lapp, Justin Poser, and Ryan Shen

JOLTON SHENERGY

New York ISO
Independent System Operator





Electricity prices are inherently fickle

- The prices we use are a combination of forecasts and estimates through the New York ISO

As renewable energy penetration increases, grid scale batteries allow

- Operators to stabilize electricity supply
- Consumers to profit off of hourly price differences

We're a contractor supplying batteries to a single generator in the New York Hudson Valley

There are a number of battery types and storage options to choose from, which vary according to the following parameters:

- Round Trip Efficiency
- Rate of discharge in an hour (Battery State of Charge)
- Maximum energy stored
- Size of the Battery
- Cost of the Battery

Optimization Formulation

Decision Variables:

Energy Prices are Available for 24 hours - Buying and Selling

Prices

Time:	Price:
12 pm	$20
1 pm	$22
2 pm	$21
…	…

BUY

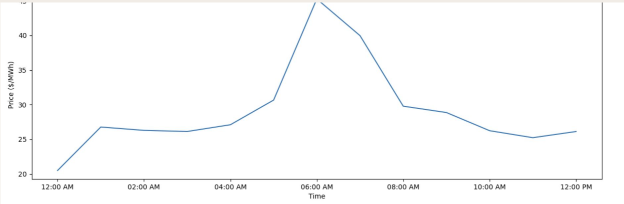
SELL

JOLTON SHENERGY

Jolton Shenergy Inc.

What to do?

Prices



Decision Variables:

Energy Prices are Available for 24 hours - Buying and Selling

Prices

Time:	Price:
12 pm	$20
1 pm	$22
2 pm	$21
...	...

BUY

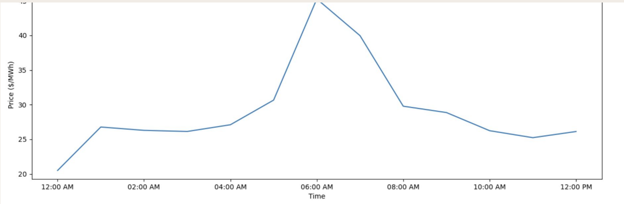
SELL

Jolton Shenergy Inc.

Buy/Sell Plan:

Time:	Decision:	Total Profit:
12 pm	Buy 100 MH	-$200
1 pm	Buy 50 MH	-$75
2 pm	Sell 150 MH	+$400
…	…	**Total: $125**

Prices



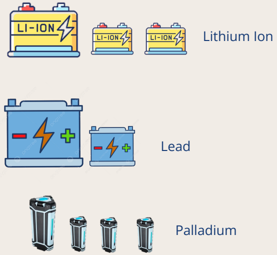
Decision Variables:

Energy Prices are Available for 24 hours - Buying and Selling

Time:	Price:
12 pm	$20
1 pm	$22
2 pm	$21
...	...

Rented Warehouse 1

Rented Batteries at Warehouse 1:

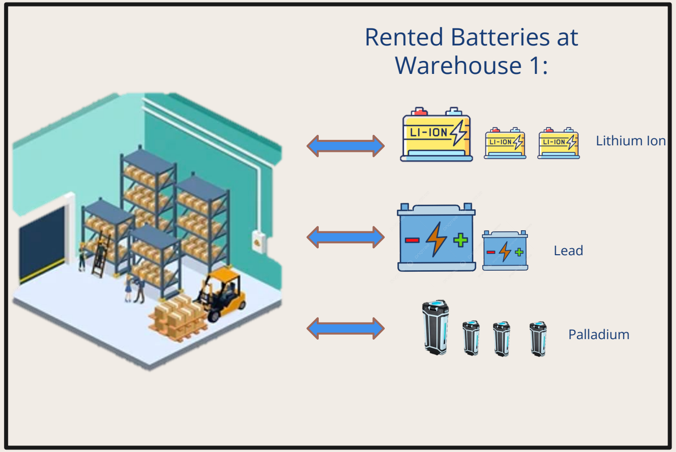


Decision Variables:

Energy Prices are Available for 24 hours - Buying and Selling

Time:	Price:
12 pm	$20
1 pm	$22
2 pm	$21
...	...

Rented Warehouse 1



Warehouse 2

Warehouse 3

... etc

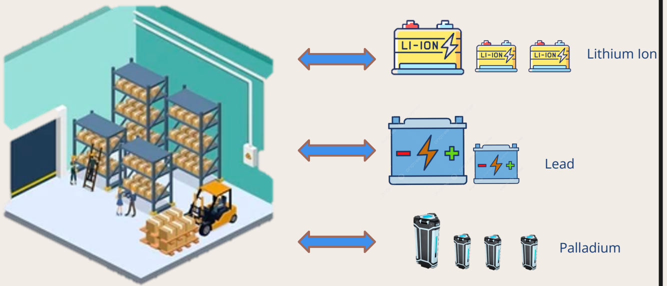
Decision Variables:

Energy Prices are Available for 24 hours - Buying and Selling

Time:	Price:
12 pm	$20
1 pm	$22
2 pm	$21
…	…

Rented Warehouse 1

Rented Batteries at Warehouse 1:



Decision Variables:

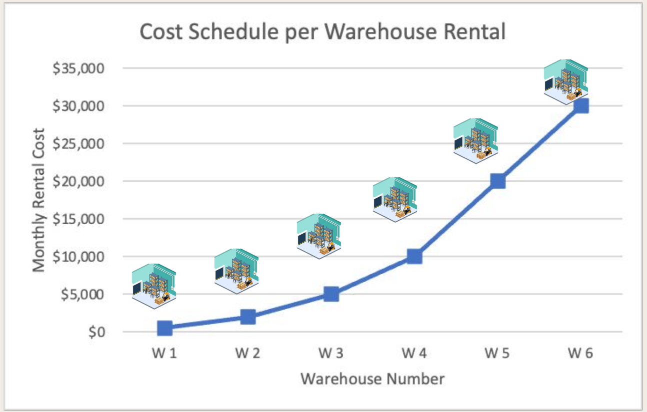
- Number of Warehouses to rent
- Number of Batteries to Rent of each type
- Amount of power to buy/sell for each battery type at each time

Warehouse 2

Warehouse 3

How to choose long term investments?

Warehouses



Batteries

$100 x10

$250 x20

$200 x15

?

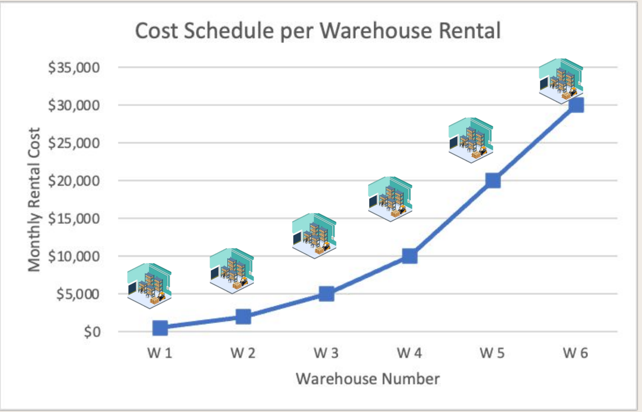
- Price data: **released daily**
- Warehouses and batteries: **rented monthly**
- How many to rent?

Investments ?

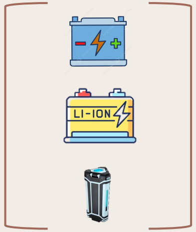
Item:	Cost
Num Warehouses: 3	$270K
Num Battery A: 10	$1000K
Num Battery B: 20	$2500K
Num Battery C: 15	$3000K
TOTAL:	**$6700K**

How to choose long term investments?

Warehouses



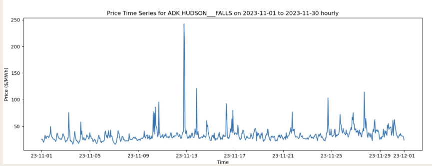
Batteries



$100	x10
$250	x20
$200	x15

Solution:

- **Look at one month historical data**
- **Optimize over that period**
- **Use the warehouse & battery numbers to guide future investment decisions**



- Optimal Warehouse #
- Optimal Battery #

Objective Function:
1st stage: investment

Backwards Looking Monthly Optimization to determine Fixed Investments

$$\max_{S_{t,b},B_{t,b},N_b,W} \sum_{t=1}^{T}\sum_{b=1}^{B}\left[\text{Arbitrage Profits}\right] - \sum_{b=1}^{B}\text{Battery Costs} - \sum_{w=1}^{W}\text{Warehouse Costs}$$

Objective Function:
1st stage: investment

Backwards Looking Monthly Optimization to determine Fixed Investments

$$\max_{S_{t,b},B_{t,b},N_b,W} \sum_{t=1}^{T}\sum_{b=1}^{B}[\text{Arbitrage Profits}] - \sum_{b=1}^{B} \text{Battery Costs} - \sum_{w=1}^{W} \text{Warehouse Costs}$$

$$= \max_{S_{t,b},B_{t,b},N_b,W} \sum_{t=1}^{T}\sum_{b=1}^{B}[(S_{t,b} - B_{t,b}) * P_t] - \sum_{b=1}^{B} \boxed{N_b} C_b - \sum_{w=1}^{W} \boxed{Z_w} C_w$$

Choose Optimal Investments

where: C_b is the cost of a battery type b

Z_w is a binary auxiliary variable equal to 1 if we rented warehouse w

C_w is the cost of warehouse w

P_t is the price of power at time t

Objective Function:
2nd stage: Real Time Arbitrage

Daily Optimization with Fixed Investments

$$\max_{S_{t,b},B_{t,b},\cancel{Z_w}} \sum_{t=1}^{T}\sum_{b=1}^{B}\left[\text{Arbitrage Profits}\right] - \cancel{\sum_{b=1}^{B}\text{Battery Costs}} - \cancel{\sum_{w=1}^{W}\text{Warehouse}}\ \text{Costs}$$

$$= \max_{S_{t,b},B_{t,b}} \sum_{t=1}^{24}\sum_{b=1}^{B}[(S_{t,b} - B_{t,b}) * P_t]$$

where: C_b is the cost of a battery type b

Z_w is a binary auxiliary variable equal to 1 if we rented warehouse w

C_w is the cost of warehouse w

P_t is the price of power at time t

Constraints

Consider:

- Space capacity of warehouses
- Charge capacity of batteries
- Charging/Discharge rate of batteries

Constraints

Consider:

- Space capacity of warehouses
- Charge capacity of batteries
- Charging/Discharge rate of batteries

3.3 Constraints

Charge Flow: $$0 \leq \sum_{0}^{t} [L_b B_{t,b} - S_{t,b}] \leq N_b A_b \quad \forall b \in B, \quad \forall t \in T$$

Charge can't be negative or exceed capacity

Charging Rate: $$L_b B_{t,b} \leq M_b \quad \forall b \in B, \quad \forall t \in T$$

Cannot charge faster than capacity

Discharge Rate: $$S_{t,b} \leq N_b \quad \forall b \in B, \quad \forall t \in T$$

Cannot discharge faster than capacity

Warehouse Capacity: $$W * f \geq \sum_{b=1}^{B} V_b N_b$$

Cannot store more batteries than we have warehouse space

Auxiliary Vars: $$W = \sum_{z=1}^{Z} Z_w$$

Constraint for auxiliary vars

Where constants above are:

A_b : Is the charge capacity of battery type b
L_b : Is the charging efficiency of battery type b
M_b : Is the maximum charge amount in one time period for battery type b
N_b : Is the maximum discharge amount in one time period for battery type b
V_b : Is the space (square feet) taken up for battery type b
f : Is the amount of square feet provided by a warehouse

And decision variables (again) are:

N_b : The number of batteries to rent of type b
W : The number of warehouses to rent
Z_b : Whether or not to rent warehouse b specifically - binary auxiliary variable
$B_{t,b}$: The amount of power to purchase at time t for batteries b
$S_{t,b}$: The amount of power to sell at time t from batteries b

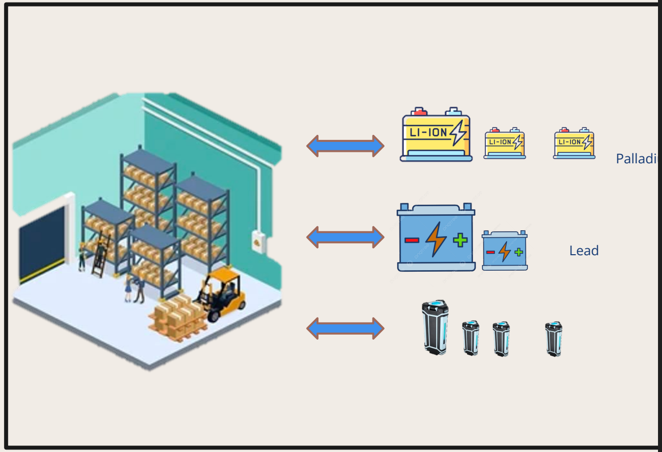
Code

Web Scrape the NY ISO for next month's prices and 30 days before

Time:	Price:
12 pm	$20
1 pm	$22
2 pm	$21
…	…

1. Optimize Based on Params for 30 days of historical data
2. Get optimal battery_counts for each type of battery
3. Optimize again with battery-counts set

Lit




```
parameters = {
    'name': 'ElectricityArbitrage',
    'generator_name': 'ADK HUDSON___FALLS',
    'date_range': None,
    'num_markets': 1,
    'battery_types': {
        'lithium': {
            'size': 22.1,
            'capacity': 100,
            'charge_loss': 0.75,
            'max_charge': 40,
            'max_discharge': 15,
            'cost': 12500
        },
        'lead': {
            'size': 20.3,
            'capacity': 350,
            'charge_loss': 0.68,
            'max_charge': 10,
            'max_discharge': 40,
            'cost': 11000
        },
        'palladium': {
            'size': .1,
            'capacity': 5,
            'charge_loss': 0.33,
            'max_charge': 5,
            'max_discharge': 5,
            'cost': 50
        }
    },
    'battery_types_used': ['lithium', 'lead', 'p
```

Create buying and selling decision variables for each time period and each type of battery.

```
for battery_type in battery_types_used:
    for key in [f'{battery_type}_buy', f'{battery_type}_sell']:
        decision_var_dict[key] = model.addVars(num_periods, vtype=GRB.CONTINUOUS, name=key, lb=0)
```

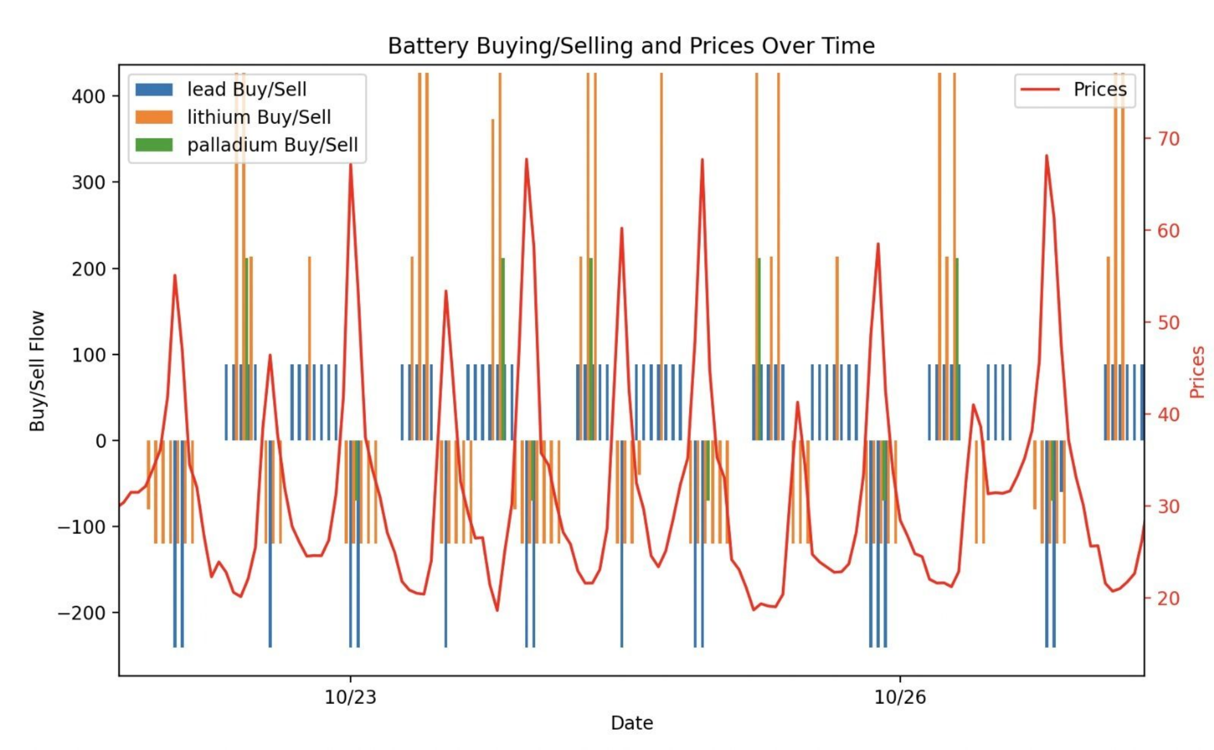
Loop over every time period, calculate a current level of charge across batteries, and check constraints using that level.

```
for p in range(num_periods):
    current_level = gp.quicksum(charge_loss * buy[p_] - sell[p_] for p_ in range(p + 1))

    if not carry_over and p % 24 == 0:
        model.addConstr(current_level <= 0, 'CarryOverConstraint')

    model.addConstr(current_level <= capacity * battery_count, f'CapacityConstraint_period_{p+1}')
    model.addConstr(current_level >= 0, f'SupplyConstraint_period_{p+1}')
    model.addConstr(buy[p] * charge_loss <= max_charge * battery_count, f'ChargeConstraint_period_{p+1}')
    model.addConstr(sell[p] <= max_discharge * battery_count, f'DischargeConstraint_period_{p+1}')
```

Optimization Results

First Stage

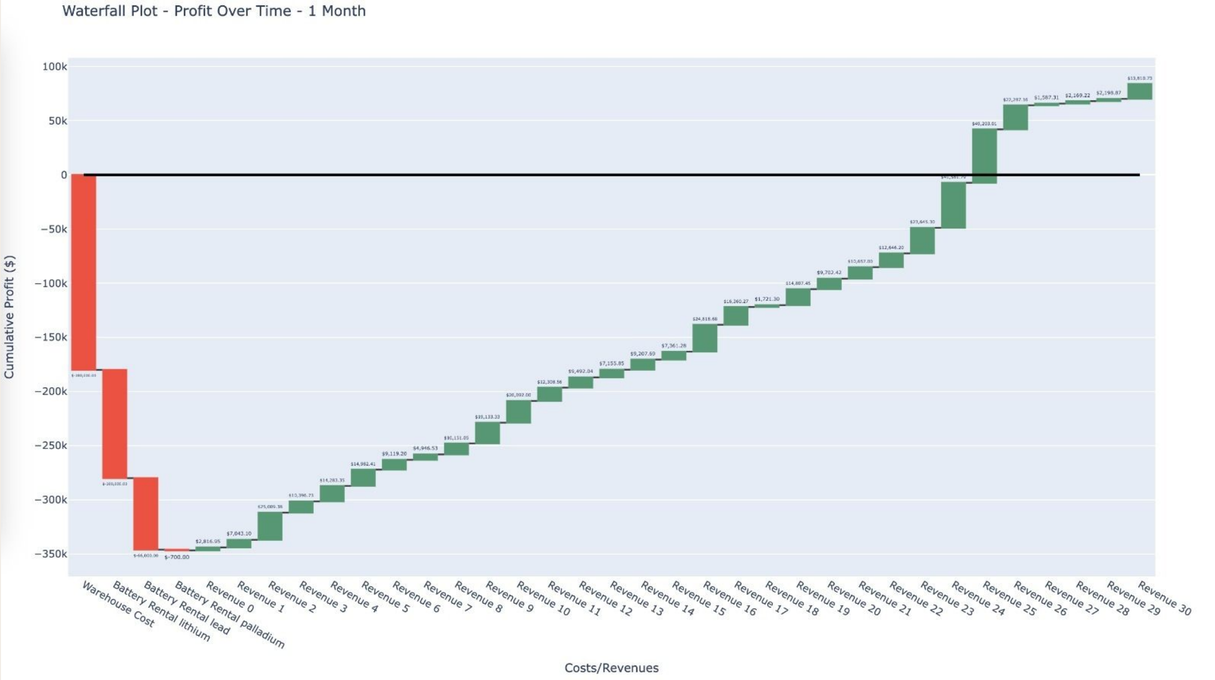
Warehouses:	3 / 5
Lithium batteries:	8
Lead Batteries:	6
Palladium Batteries:	14



Optimization Results

123% of investment returned

Warehouses:	3 / 5	- $180K
Lithium batteries:	8	-$100K
Lead Batteries:	6	-$66K
Palladium Batteries:	14	-$700
Profit from Arbitrage:		$429K
TOTAL:		$83K



OPTIGUIDE

For optiguide, we alter the conditions that affect cost and revenue

What if every 14 periods, we can't sell electricity to the grid?

```
for p in range(num_periods):
    if p%14 == 0:
        model.addConstr(sell[p] = 0, f'DischargeConstraint_period_{p+1}')
```
```

## What if we have a minimum charge at all times of 20%?

```
Answer Code:
for p in range(num_periods):
    model.addConstr(charge_loss * buy[p_] - sell[p_] for p_ in range(p + 1) >= 0.2 * capacity)
""""
```

What if you added a vanadium flow battery and knew the specs as follows .... ?

```
Answer Code:
parameters['battery_types']['vanadium'] = {
            'size': 15,
            'capacity': 200,
            'charge_loss': 0.8,
            'max_charge': 140,
            'max_discharge': 140,
            'cost': 550}
parameters['battery_types_used'].append('vanadium')
```

Disclaimer:
Due to our web scraper + optiguide bugs, we don't have working optiguide output at the moment
```

The end