MAXIMIZING THE ENERGY ARBITRAGE REVENUE OF A PV-BESS SYSTEM UNDER DAY-AHEAD ELECTRICITY PRICES USING LINEAR PROGRAMMING

Practical Training Project - Tram Tran

PROJECT OVERVIEW

Problem

- A fixed PV-BESS-GRID system with day-ahead (DA) electricity prices are known in advance
- Simulation model determines optimal hourly power allocation under realistic operational constraints
- Objective: Maximize energy arbitrage revenue

Methods

- Uses linear programming (LP) model to optimize energy dispatch
- Implemented in Python using PuLP library with CBC solver

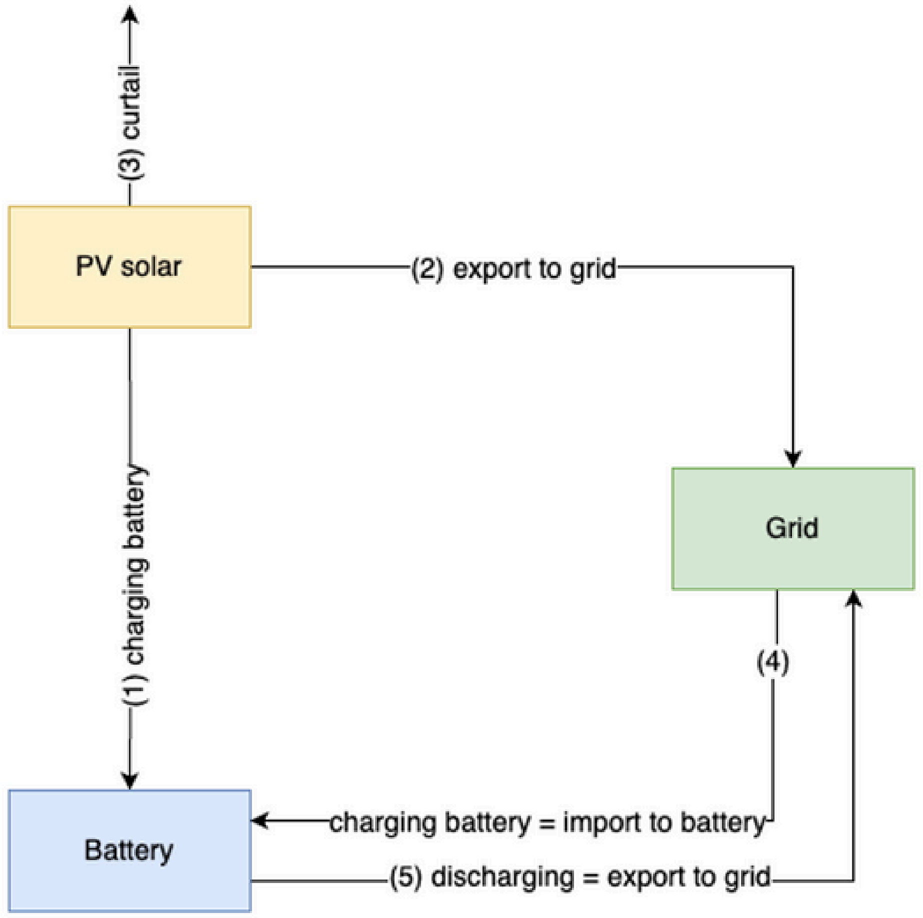
PROBLEM CLARIFICATION

Maximize revenue of PV-BESS-GRID system responding to DA electricity prices

System Configuration

- Fixed grid connection capacity (e.g., 10 MW)
- PV system with higher peak output (e.g., 20 MW)
- BESS with defined power and energy capacity (e.g., 30 MWh)

Operational Scenarios



Optimization Model

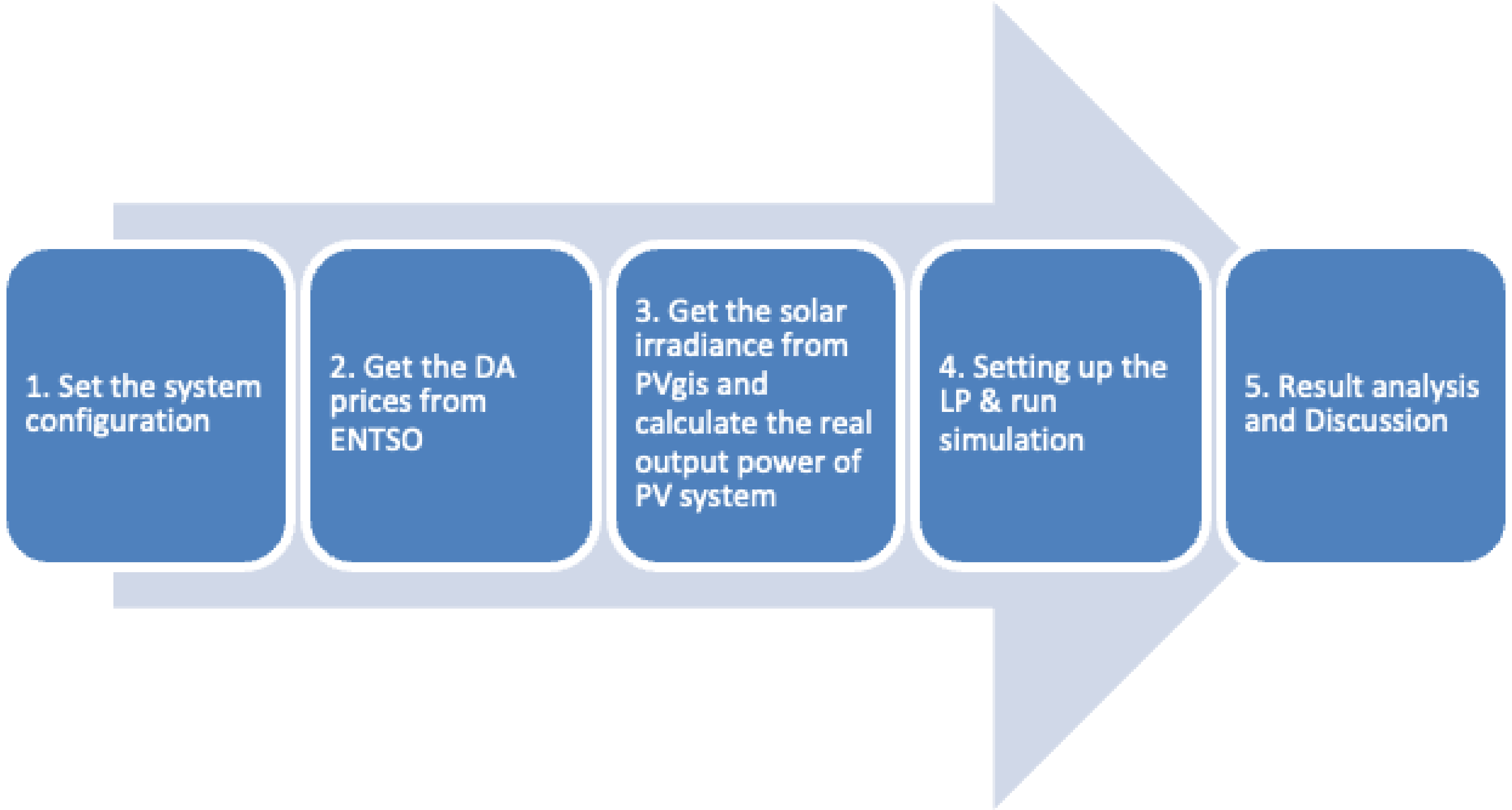
- Target: Maximum arbitrage revenue/profit
- Optimize based on
  - DA prices
  - System configuration
  - Cost, fee

Questions

- Grid fees?
- Cycle costs?
- PV costs?
- OR is it just a revenue-calculator on some sizes?

PROBLEM APPROACH

- Expand from the traditional arbitrage model (BESS-GRID) with more complex energy flows
- Consideration of system constraints and grid costs presence
- Testing with multiple battery storage capacity scenarios to find the best configuration
- Does NOT consider capital, O&M, or long-term financial analysis
- Using Linear Programing (because DA prices are known 24 hours in advance)



Overall problem solving process

ALGORITHM LOGIC

Step 1: Define system parameters

PV solar
Set location (Vaasa, Finland)
Set year (2023)
Set PV nominate power (20 MW)
Set system efficiency (80%)

BESS
Set battery size (30 MWh)
Set max charging power (10 MW)
Set max discharging power (10 MW)
Set charging and discharging efficiencies (90%)
Set battery start state of charge (SoC) (empty)

GRID
Set max power from/to grid (10 MW)

Step 2: Get DA prices from ENTSO

Get target date (eg. tomorrow)

Get hourly prices from energy market API for that date, stored in da_prices

Make sure there is not timezone mismatch

Step 3: Calculate PV power output by get solar irradiance from PVgis

Get hourly solar irradiance data for target date from solar data API

Calculate expected PV power for each hour, stored in pv_profile

Issue: only historical data from 2023 available

Step 4: Set up LP optimization problem

Set up Variables
Set up Constraints
Set up Objectives

(see the next slide)

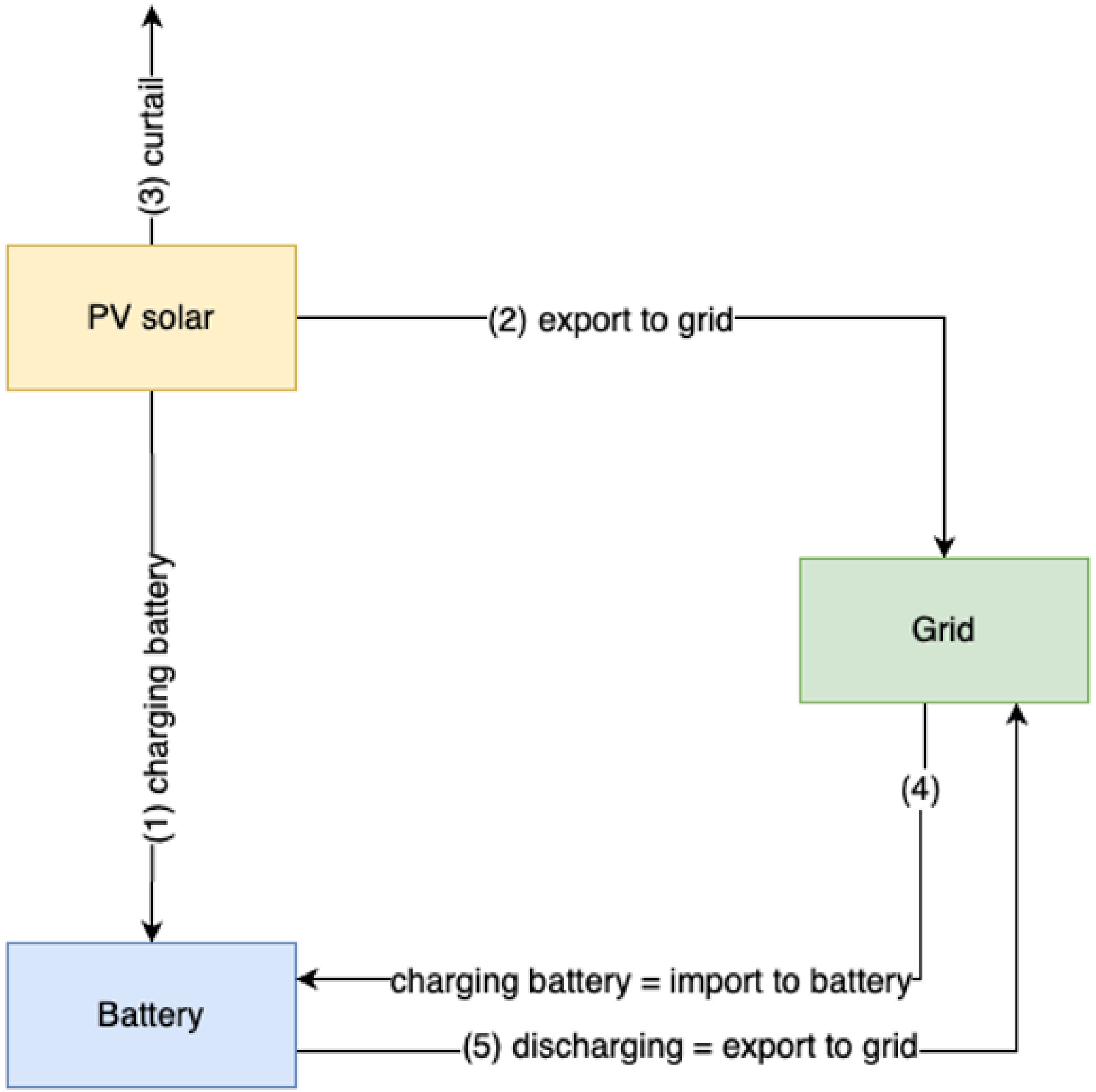
Step 5: Display results

Show results: total profit, power flows, battery SoC over the day

Print result to screen, export to Exel, plot

ALGORITHM LOGIC

Step 4: Set up Linear Programing optimization problem



Everyday, for each hour t in 0 to 23:

Variables:

Decide how much PV power to:

(1) send to battery (pv2batt)

(2) send to grid (pv2grid)

(3) curtail (curtail)

Decide how much (4) grid power to buy for charging battery (grid2batt)

Decide how much (5) battery power to send to grid (batt2grid)

Update SoC (soc)

Constraints:

1. Total PV power pv_profile= pv2batt + pv2grid + curtail
2. Battery SoC update:

*SoC at hour [t] = SoC at hour [t-1] + (pv2batt + grid2batt)*charging_efficiency - batt2grid / discharging_efficiency*

3. Charging power can't exceed max charging limit
4. Discharging power can't exceed max discharging limit
5. Battery can't charge and discharge at the same time
6. Grid power bought can't exceed max limit
7. Grid power sold can't exceed max limit
8. Battery can't discharge more energy than it has
9. End of day: SoC(23) = SoC(0)

ALGORITHM LOGIC

Step 4: Set up Linear Programing optimization problem

Objectives function: Maximize all-day profit

$$Profit = \sum_{t=0}^{T-1} [E_t^{export}.cost_{export,t} - E_t^{import}.cost_{import,t} - E_t^{batt->grid}.c_{cyc}]$$

$$cost_{import} = energy_{import} + tax_{import} + transmission_{import}$$

$$energy_{import} = DA\,prices + marginal_{import} + VAT$$

$$tax_{import} = tax_{electricity} + VAT$$

$$transmission_{import} = energy_{transmission,import} + VAT$$

$$cost_{export} = DA\,prices - marginal_{export} - (transmission_{export} + VAT)$$

```python
# --- Initialize model ---
prob = LpProblem("Energy_Optimization", LpMaximize)

# --- Decision variable ---
pv2batt = [LpVariable(f"pv2batt_{t}", lowBound=0) for t in range(T)]
pv2grid = [LpVariable(f"pv2grid_{t}", lowBound=0) for t in range(T)]
curtail = [LpVariable(f"curtail_{t}", lowBound=0) for t in range(T)]
grid2batt = [LpVariable(f"grid2batt_{t}", lowBound=0) for t in range(T)]
batt2grid = [LpVariable(f"batt2grid_{t}", lowBound=0) for t in range(T)]
soc = [LpVariable(f"soc_{t}", lowBound=0, upBound=battery_capacity) for t in range(T)]


# --- Binary variable to prevent simultaneous charging/discharging ---
mode = [LpVariable(f"mode_{t}", cat="Binary") for t in range(T)]  # 1 if discharging, 0 if charging

# --- Constraints ---
for t in range(T):
    # 1. Total PV generated power = charge to battery + sell to grid + cultail
    prob += pv2batt[t] + pv2grid[t] + curtail[t] == pv_profile[t]

    # 2. SoC is updated per hour, initial SOC is assumed
    if t == 0:
        prob += soc[t] == initial_soc
    else:
        prob += soc[t] == soc[t-1] + (pv2batt[t] + grid2batt[t]) * charge_eff - batt2grid[t] / discharge_eff

    # 3. Charging limit
    prob += pv2batt[t] + grid2batt[t] <= charge_power_limit

    # 4. Discharging limit
    prob += batt2grid[t] <= discharge_power_limit

    # 5. Do not charge + discharge at the same time (big-M method)
    prob += pv2batt[t] + grid2batt[t] <= charge_power_limit * (1 - mode[t])
    prob += batt2grid[t] <= discharge_power_limit * mode[t]

    # 6. Limit power import from grid (charging only)
    prob += grid2batt[t] <= grid_power_limit

    # 7. Limit the power exported to the grid (PV + batt)
    prob += pv2grid[t] + batt2grid[t] <= grid_power_limit

    # 8. Discharge amount <= SOC.
    if t == 0:
        prob += batt2grid[t] <= initial_soc*discharge_eff
    else:
        prob += batt2grid[t] <= soc[t-1]*discharge_eff

    # 9. At the end of the day, force the battery to discharge to initial SOC.
    prob += soc[T-1] == initial_soc


## --- Objective function: maximize arbitrage profit ---
profit = lpSum([
    (pv2grid[t] + batt2grid[t]) * (da_prices[t]-marginal_export) - grid2batt[t] * (da_prices[t]*(1+VAT_tax)+marginal_import+consumption_tax+transmission_cost) - batt2grid[t] * cycle_cost

    for t in range(T)
])

prob += profit
```


Source: Kosonen, A. (2025, March). PV system economics [Lecture slides]. BL40A2601 Wind Power and Solar Energy Technology and Business, LUT University.

ALGORITHM LOGIC

The same algorithm logic is then also applied to test various battery capacities to determine the optimal capacity that maximizes profit.

RESULT ANALYSIS

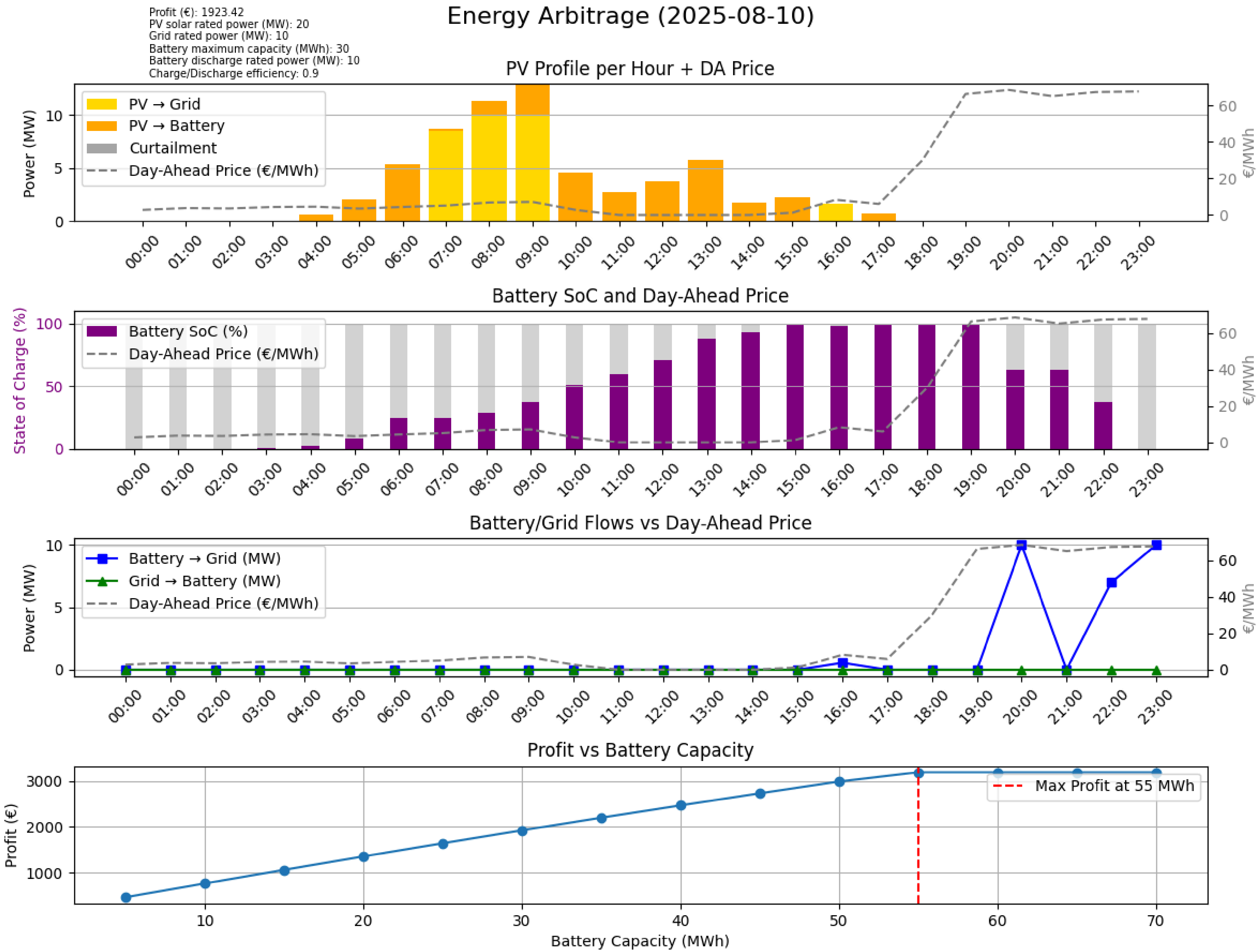
Sample date: 2025-08-10

Parameter	Value
Objective (Profit) (€)	1923.42
PV solar rated power (MW)	20
Grid rated power (MW)	10
Battery maximum capacity (MWh)	30
Battery discharge rated power (MW)	10
Battery charge rated power (MW)	10
Charge/Discharge efficiency (%)	0.9

Battery Capacity (MWh)	Profit (€)
5	467.01
10	766.75
15	1063.38
20	1355.72
25	1642.56
30	1923.42
35	2199.06
40	2467.16
45	2729.71
50	2985.99
55	3188.25
60	3188.25
65	3188.25
70	3188.25

Time	Day-Ahead Price (€/MWh)	Solar Irradiance (W/m²)	PV Power (MW)	PV → Grid (MW)	PV → Battery (MW)	PV Curtailment (MW)	Grid → Battery (MW)	Battery → Grid (MW)	Battery SoC (%)
2025-08-10 00:00	2.79	0	0	0	0	0	0	0	0
2025-08-10 01:00	3.78	0	0	0	0	0	0	0	0
2025-08-10 02:00	3.57	0	0	0	0	0	0	0	0
2025-08-10 03:00	4.36	2.43	0.03888	0	0.03888	0	0	0	0.11664
2025-08-10 04:00	4.53	37.07	0.59312	0	0.59312	0	0	0	1.896
2025-08-10 05:00	3.5	126.95	2.0312	0	2.0312	0	0	0	7.9896
2025-08-10 06:00	4.38	334.61	5.35376	0	5.35376	0	0	0	24.05088
2025-08-10 07:00	5.12	546.24	8.73984	8.5092267	0.23061333	0	0	0	24.74272
2025-08-10 08:00	6.8	706.9	11.3104	10	1.3104	0	0	0	28.67392
2025-08-10 09:00	7.13	807.85	12.9256	10	2.9256	0	0	0	37.45072
2025-08-10 10:00	2.77	286.14	4.57824	0	4.57824	0	0	0	51.18544
2025-08-10 11:00	0.03	171.82	2.74912	0	2.74912	0	0	0	59.4328
2025-08-10 12:00	0	235.03	3.76048	0	3.76048	0	0	0	70.71424
2025-08-10 13:00	0.01	362.5	5.8	0	5.8	0	0	0	88.11424
2025-08-10 14:00	0.02	107.38	1.71808	0	1.71808	0	0	0	93.26848
2025-08-10 15:00	1.25	140.24	2.24384	0	2.24384	0	0	0	100
2025-08-10 16:00	8.29	103.99	1.66384	1.66384	0	0	0	0.559224	97.9288
2025-08-10 17:00	6	43.15	0.6904	0	0.6904	0	0	0	100
2025-08-10 18:00	30.03	3.04	0.04864	0.04864	0	0	0	0	100
2025-08-10 19:00	66.4	0	0	0	0	0	0	0	100
2025-08-10 20:00	68.61	0	0	0	0	0	0	10	62.96296333
2025-08-10 21:00	65.24	0	0	0	0	0	0	0	62.96296333
2025-08-10 22:00	67.46	0	0	0	0	0	0	7	37.03703667
2025-08-10 23:00	67.79	0	0	0	0	0	0	10	0

RESULT ANALYSIS



- Accuracy of the data?
  - Compare to other methods of getting DA price and solar irradiance data
- Compliance with the system's technical constraints?
  - Check if all the system constraints are satisfied
- Reasonableness of the energy distribution strategy?
  - SoC vs Electricity price
  - PV power behavior
  - Impact of grid cost
  - Best battery size?

DISCUSSION

Benefits

Combines PV generation with battery storage for realistic modeling
Uses actual day-ahead (DA) prices for higher accuracy
Linear Programming provides reliable results with simpler implementation than more complex optimization methods

Limitations

LP only applicable when all objectives/constraints are linear; real-world factors can cause nonlinearity.
Relies on fixed system and cost assumptions; parameter variability would need advanced algorithms and more computing power.
Solar irradiance based on historical data, not real-time measurements.

Future work

Add variable PV sizes, efficiencies, and dynamic components.
Incorporate load demand profiles for realistic consumption patterns.
Extend to other storage technologies (e.g., fuel cells with hydrogen revenue).
Include full economic indicators (LCOE, WACC, IRR, NPV) and CAPEX/OPEX for investment analysis.

MY LEARNING OUTCOMES

Arbitrage concept

This is the first time I heard about arbitrage

Problem solving

How to convert real world problem → math problem
→ computer language
How to evaluate the result
What is the root cause
“What if" questions

Energy system modeling

How to set up and run a simulation
How to evaluate the simulation result
What is the benefits, limitations of a simulation model
Potential expansions

Optimization problem & Solver tools

Not just Excel!
There are a variety of methods (LP, MINLP, Monte Carlo, Reinforcement learning...) and LP is just the simplest way.

THANK YOU

Please feel free to give feedback.