Optimization & Decision

Masters Degree in Mechanical Engineering

Project - Part 1 [EN]

Authors:

Francisco Pinto (ISTID 089888) francisco.v.pinto@tecnico.ulisboa.pt
Henrique Branquinho (ISTID 087203) henrique.branquinho@tecnico.ulisboa.pt

Group 32

2024/2025 – 3rd Quarter

Contents

1 Introduction

The integration of renewable energy sources into the electrical grid presents challenges related to variability in power generation and demand fluctuations over time. Efficient energy management strategies are required to balance supply and demand while minimizing operational costs.

One approach involves optimizing battery dispatch to store excess renewable energy and discharge it when needed, reducing reliance on the power grid.

This project addresses the problem as a linear programming model to determine the optimal charging and discharging strategy for multiple batteries in a smart grid.

The problem is simplified to be solved using the Simplex method. The results from this approach provide insights into how an optimal energy management strategy can reduce costs and improve grid stability.

Method	**Linear**	**Nonlinear**	**Multi-Agent**	**Handles Dynamics**	**Adaptive**	**Justification**
Simplex	✓			Static only		Fast and exact for linear problems; good baseline for comparison.
DE	Partial	✓		✓		Effective for hybrid and nonlinear spaces; simple implementation.
PSO (1p1c)	Partial	✓		✓		Intuitive, flexible, and easily tuned; good for dynamic settings.
PSO (Multi)	Partial	✓	✓	✓		Captures agent-level decisions; suitable for distributed problems.
DDPG (RL)		✓	Pseudo Multi-Agent	✓	✓	Learns behaviors over time; models system as one adaptive agent.

Table 1: Comparison of Optimization Techniques for Energy Systems

2 Mathematical Formulation and Simplex Method

2.1 Problem Formulation

The objective of this project is to optimize the dispatch of a battery in a smart grid by minimizing energy costs while ensuring demand is met and battery constraints are satisfied. For simplicity, all consumers and producers are aggregated:

- P_t^{prod}: Total solar production at time t, summed across all producers.
- P_t^{load}: Total energy demand at time t, summed across all consumers.

This aggregation reduces the model's complexity and keeps the formulation linear, making it solvable using the Simplex Method.

2.2 Decision Variables

- P_t^c: Power charged into the battery at time t (kW), with $0 \leq P_t^c \leq P_t^{prod}$

- P_t^d: Power discharged from the battery at time t (kW), constrained by $0 \leq P_t^d \leq E_{t-1}$
- E_t: Energy stored in the battery at time t (kWh), with $0 \leq E_t \leq E_{\max}$
- P_t^{grid}: Power drawn from the grid at time t (kW), $P_t^{grid} \geq 0$

2.3 Objective Function

Minimize the total cost of grid energy over the horizon T:

$$\min \sum_{t=1}^{T} C_{\text{grid}} \cdot P_t^{grid} \tag{1}$$

2.4 Constraints

2.4.1 Battery Energy Balance:

$$E_t = E_{t-1} + \eta_c \cdot P_t^c - \frac{P_t^d}{\eta_d}, \quad \forall t > 0 \tag{2}$$

With initial condition:

$$E_0 = 0.5 \cdot E_{\max} = 1250 \text{ kWh} \tag{3}$$

2.4.2 Battery Operational Limits:

$$0 \leq E_t \leq E_{\max}, \quad 0 \leq P_t^c \leq P_t^{prod}, \quad 0 \leq P_t^d \leq E_{t-1}, \quad \forall t \tag{4}$$

2.4.3 Demand Satisfaction:

$$P_t^{grid} + P_t^d + P_t^{prod} \geq P_t^{load}, \quad \forall t \tag{5}$$

This guarantees that energy demand is always fulfilled by the grid, the battery, or directly from solar generation.

2.5 Solution Approach

To simplify the computation, the system models only one battery, representative of the aggregated behavior of multiple distributed units. The formulation remains a Linear Program (LP) because:

- All constraints and the objective function are linear.
- Decision variables are continuous and non-negative.
- The Simplex Method can efficiently solve the problem to global optimality.

2.6 Results and Analysis

The model produces:

- **Battery SoC (E_t)** evolution, showing energy storage and release over time.
- **Grid Import (P_t^{grid})** as an indicator of external energy dependency.
- **Charging / Discharging (P_t^c, P_t^d)** as indicators of energy shifting behavior.
- **Total Grid Cost**, the minimized objective value.

2.6.1 Cost Analysis

Because the grid energy cost is constant, the optimizer aims to minimize external imports by using the battery as efficiently as possible. As shown in Figure **??**, most cycles correspond to discharging periods, indicating successful shifting of solar energy to later demand hours.

3 Part 2 – Metaheuristic

3.1 Introduction

To further investigate the limitations of our simplified LP model, two metaheuristic optimization strategies were applied: Differential Evolution (DE) and Particle Swarm Optimization (PSO). These strategies are best effective with nonlinear and non-convex problems and provide better insight into the real world's energy waste restrictions and operational limits. The objective remains to minimize the total operational cost. The cost function now includes:

- Grid energy cost: $C_{\text{grid}} = 0.1\,/kWh$
- Penalty for wasted solar energy: $\lambda_{\text{waste}} = 0.2$ to $5.0\,/kWh$ (depending on method)
- Battery operational constraints and efficiencies

3.2 Decision Variables

For both DE and PSO, the main decision variables are:

$P_{c,t}$	Charging power at time t
$P_{d,t}$	Discharging power at time t
SoC_t	Battery state of charge at time t

These variables are encoded in a vector of length $2T$, where T is the number of time steps.

3.3 Constraints (Handled via Penalties)

Both metaheuristics implicitly enforce constraints using penalty terms in the objective function:

- SoC bounds: $0 \leq SoC_t \leq E_{\max}$
- Power bounds: $0 \leq P_{c,t} \leq P_{\text{production},t}$, $0 \leq P_{d,t} \leq SoC_t$
- Energy balance: Supply $\geq$ Load

3.4 Objective Function (DE)

The DE method minimizes:

$$\min_{P_c, P_d} \sum_{t=1}^{T} \left[C_{\text{grid}} \cdot P_{\text{grid},t} + \lambda_{\text{waste}} \cdot P_{\text{waste},t} + \text{Penalty}_{\text{SoC}}\right] \tag{6}$$

Where:

$$\begin{aligned}
P_{\text{grid},t} &= \max(0, P_{\text{load},t} - \eta_d P_{d,t}) \\
P_{\text{waste},t} &= \max(0, P_{\text{prod},t} - P_{c,t}) \\
\text{Penalty}_{\text{SoC}} &= \begin{cases} 10^4 \cdot |SoC_t|, & \text{if } SoC_t < 0 \text{ or } SoC_t > E_{\max} \\ 0, & \text{otherwise} \end{cases}
\end{aligned}$$

3.5 Objective Function (PSO - Aggregated 1p1c)

The PSO method in the single-agent configuration uses a strong penalization strategy to balance grid usage and renewable energy waste:

$$\min_{P_{c,t}, P_{d,t}} \sum_{t=1}^{T} \left[P_{\text{grid},t} \cdot (C_{\text{grid}} + \text{PENALIZE}_{\text{grid}}) + P_{\text{waste},t} \cdot \text{PENALIZE}_{\text{waste}}\right] \tag{7}$$

Subject to:

$$\begin{aligned}
P_{\text{grid},t} &= \max(0, P_{\text{load},t} - \eta_d \cdot P_{d,t}) \\
P_{\text{waste},t} &= \max(0, P_{\text{prod},t} - P_{c,t}) \\
\text{SoC}_{t+1} &= \min\left(\max\left(\text{SoC}_t + \eta_c P_{c,t} - \frac{P_{d,t}}{\eta_d}, 0\right), E_{\max}\right)
\end{aligned}$$

3.6 Objective Function (PSO - Multi-Agent 14p51c)

In the multi-agent version of PSO, each producer charges independently, and discharge is optimized per consumer. This results in a more granular system.

$$\min_{P_{c_{i,t}}, P_{d_{j,t}}} \sum_{t=1}^{T} \left[P_{\text{grid},t} \cdot (C_{\text{grid}} + \text{PENALIZE}_{\text{grid}}) + P_{\text{waste},t} \cdot \text{PENALIZE}_{\text{waste}} + \text{Penalties}_t\right] \tag{8}$$

Subject to:

$$P_{\text{charge},t} = \sum_{i=1}^{N_{\text{prod}}} P_{c_{i,t}}$$

$$P_{\text{waste},t} = \sum_{i=1}^{N_{\text{prod}}} \max(0, P_{\text{prod}_{i,t}} - P_{c_{i,t}})$$

$$\text{SoC}_{j,t+1} = \min\left(\max\left(\text{SoC}_{j,t} + \frac{\eta_c}{N_{\text{cons}}} \cdot P_{\text{charge},t} - P_{d_{j,t}}, 0\right), \frac{E_{\max}}{N_{\text{cons}}}\right)$$

$$P_{\text{grid},t} = \max\left(0, \sum_{j=1}^{N_{\text{cons}}} P_{\text{load}_{j,t}} - \eta_d \cdot \sum_{j=1}^{N_{\text{cons}}} P_{d_{j,t}}\right)$$

3.7 Comparison with Simplex Model

The metaheuristics approach is more flexible with real-world nonlinear and complex functions, unlike the LP approach used in Part 1 which requires determinism and assumes perfect linear structures.

Due to the fact that The Simplex model rigidly follows the set procedures and linear cost function, it is efficient and fast, but does not accommodate for realism.

Key differences include:

- **Flexibility in Constraints:** Metaheuristics tolerate temporary constraint violations through penalty terms.
- **Nonlinear Capabilities:** Real-world battery behavior is nonlinear; LP cannot model such dynamics.
- **Custom Cost Functions:** DE and PSO support hybrid objectives combining grid cost, energy waste, and operational penalties.

3.8 Expected Outcome

We expect the metaheuristics to:

- Better adapt to realistic operational conditions.
- Reduce energy waste via smarter charging strategies.
- Increase computational effort slightly, in exchange for more practical feasibility.

3.9 Cumulative Cost Comparison

Figure 1 illustrates cumulative grid cost over time. As expected, the Simplex model achieves the lowest cost due to idealized assumptions. Among metaheuristics, PSO (1p1c) outperforms DE. The multi-agent PSO (14p51c) shows increased cost due to coordination complexity. The RL agent demonstrates near-optimal performance, outperforming both DE and PSO in cumulative cost.

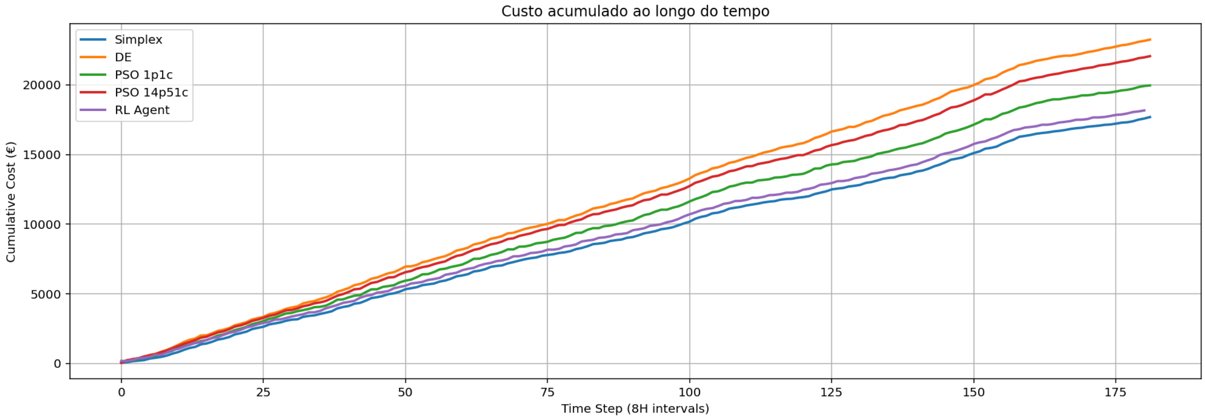

Figure 1: Cumulative cost over time for different optimization methods.

3.10 Reinforcement Learning Approach

To explore adaptive control strategies, we trained a reinforcement learning (RL) agent using the Deep Deterministic Policy Gradient (DDPG) algorithm. The agent observes:

- Current SoC
- Load at time t
- Solar production at time t

It decides charging and discharging fractions, with the goal of minimizing:

$$\text{Cost}_t = P_{\text{grid},t} \cdot (C_{\text{grid}} + \text{penalty}_{\text{grid}}) + P_{\text{waste},t} \cdot \text{penalty}_{\text{waste}}$$

The RL model was trained for 100,000 time steps. The learned policy effectively minimized grid usage and solar waste, tracking closely with the Simplex solution while adapting to dynamic conditions.

3.11 Final Comparison Summary

Table 2: Final Grid Cost Comparison

Method	Cumulative Cost (€)	Comment
Simplex (LP)	˜17,000	Ideal baseline (no penalties, perfect behavior)
DE	˜22,500	Flexible but less efficient
PSO 1p1c	˜19,500	Balanced penalties and cost
PSO 14p51c	˜21,500	Higher cost due to multi-agent coordination
RL Agent	˜18,000	Near-optimal and realistic behavior

This study compared different optimization methods: Simplex, Differential Evolution (DE), Particle Swarm Optimization (PSO) both in centralized and multi-agent architectures, and a Reinforcement Learning (RL) Agent to minimize overall energy cost in a smart grid system.

The simplex method obtained the lowest cost of €17,000 as an idealized standard when perfect linear assumptions and penalties are present. Very effective, and can even be used as benchmark but lacks the ability to mimic real-world behaviors such as waste of energy or simultaneous charging and discharging.

The DE approach, despite being flexible and appropriate for non-linear optimization, obtained the highest cost of €22,500, which suggests inefficiencies in handling soft multiple constraints and dynamic energy currents.

The PSO 1p1c model presented a good balance (€19,500), combining cumulative cost with penalty structure inclusion, and was thus suitable for realistic practical scenarios which require flexibility and optimization.

The multiagent PSO (14p51c) gave a more distributed and realistic energy-sharing environment. Though with slightly higher cost (€21,500) due to coordination complexity, it proved useful for distributed system simulation and future scalability.

Finally, the RL Agent delivered a remarkable overall cost of €18,000, almost identical to the Simplex baseline. In comparison with classical optimizers, the RL model demonstrated close-to-optimal and realistic action, learning over time without model structure specification in advance. This presents good promise for RL in adaptive or real-time energy system control.

GitHub Link

You can find the code on GitHub at:
https://github.com/FranciscoVPinto/OD_Proj

Annex: Optimization Plots

Simplex - Aggregated 1p1c

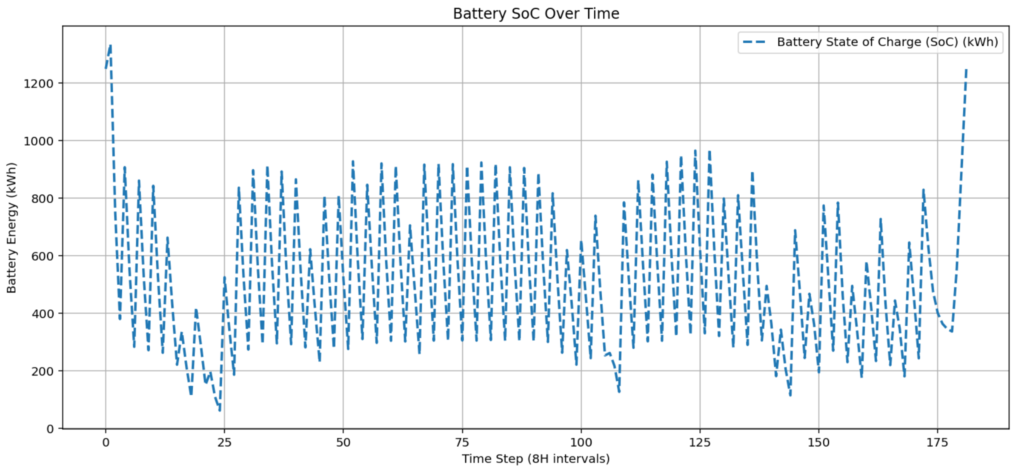

Figure 2: Battery State of Charge Over Time (Simplex)

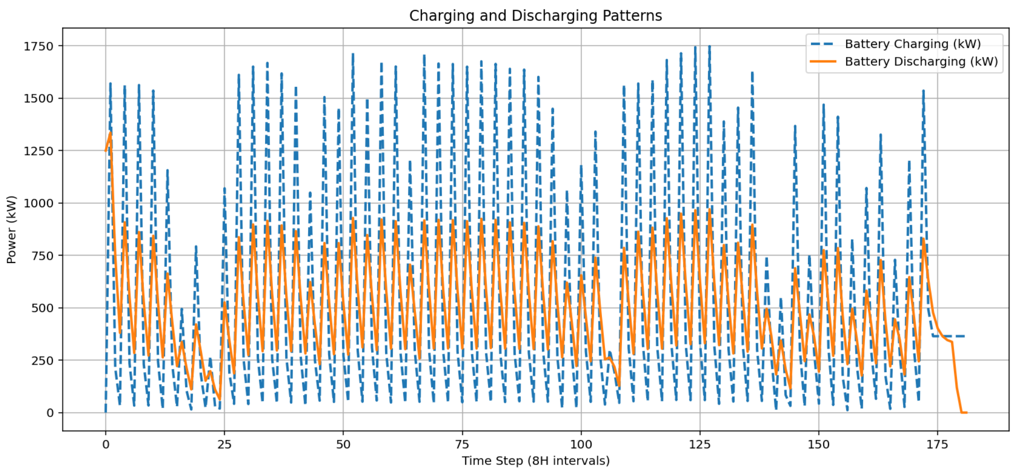

Figure 3: Total Charge and Discharge of Battery (Simplex)

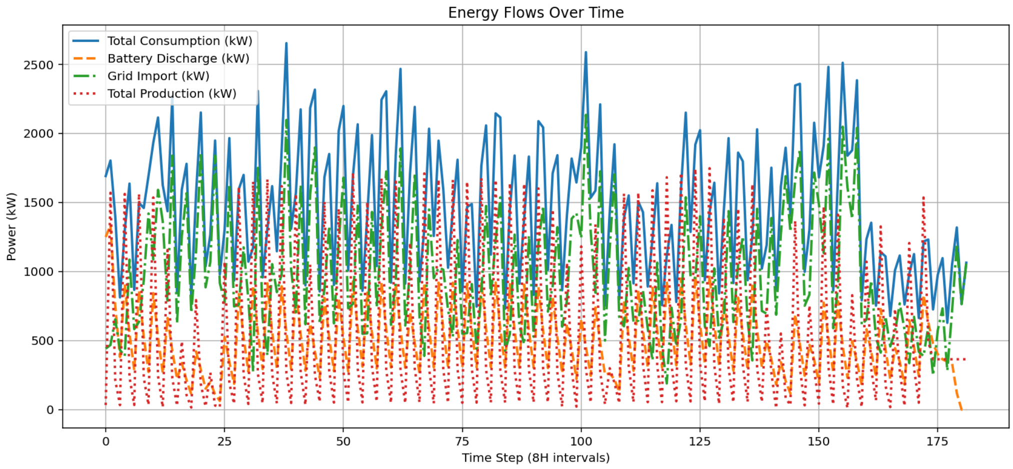

Figure 4: Energy Flow (Simplex)

DE - Aggregated 1p1c

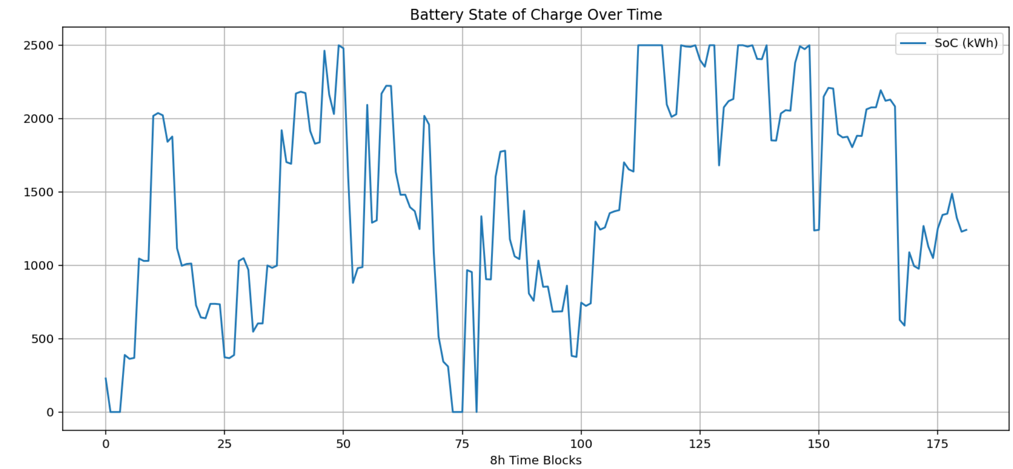

Figure 5: Battery State of Charge Over Time (DE)

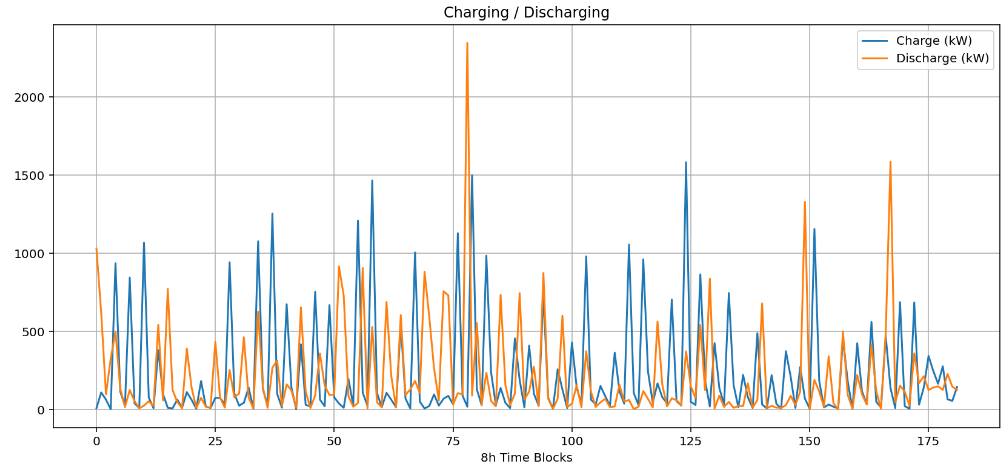

Figure 6: Total Charge and Discharge of Battery (DE)

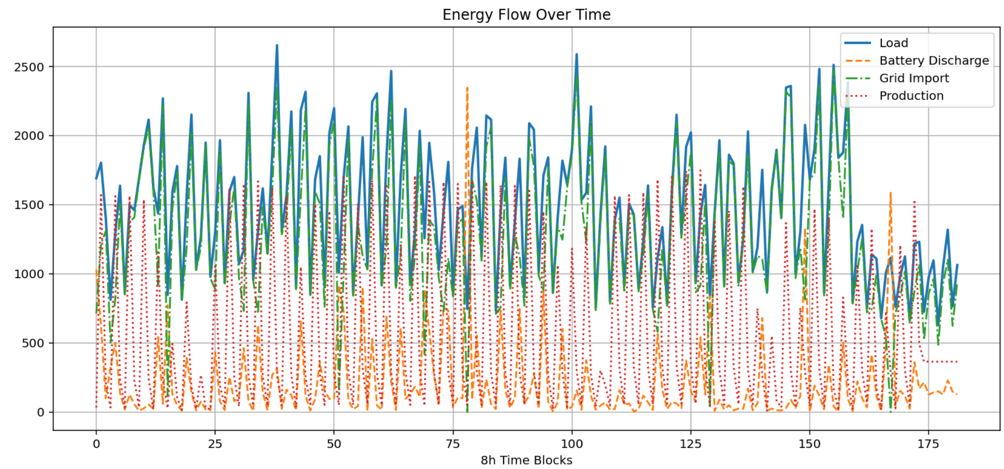

Figure 7: Energy Flow (DE)

PSO - Aggregated 1p1c

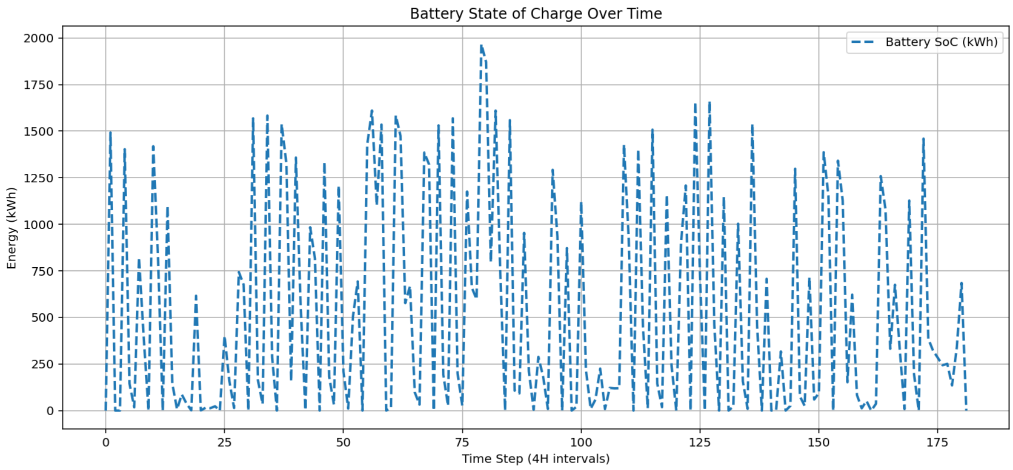

Figure 8: Battery State of Charge Over Time (PSO 1p1c)

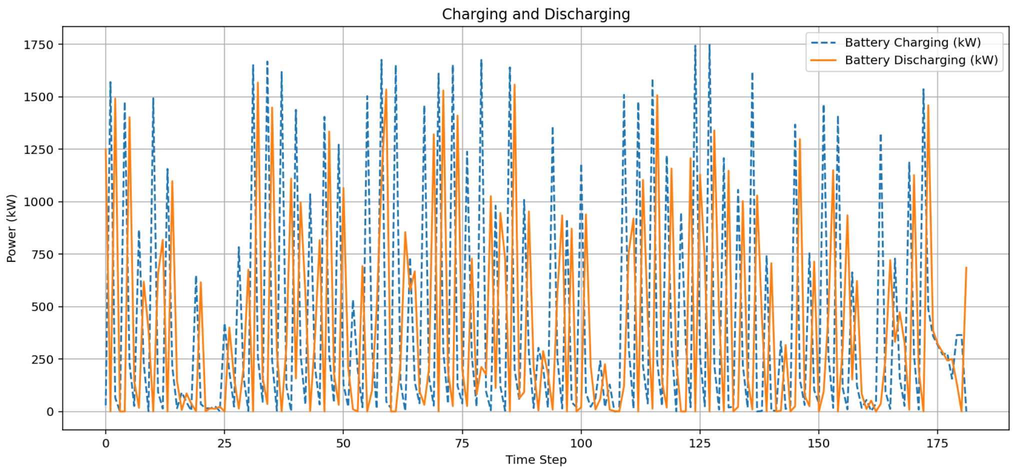

Figure 9: Total Charge and Discharge of Battery (PSO 1p1c)

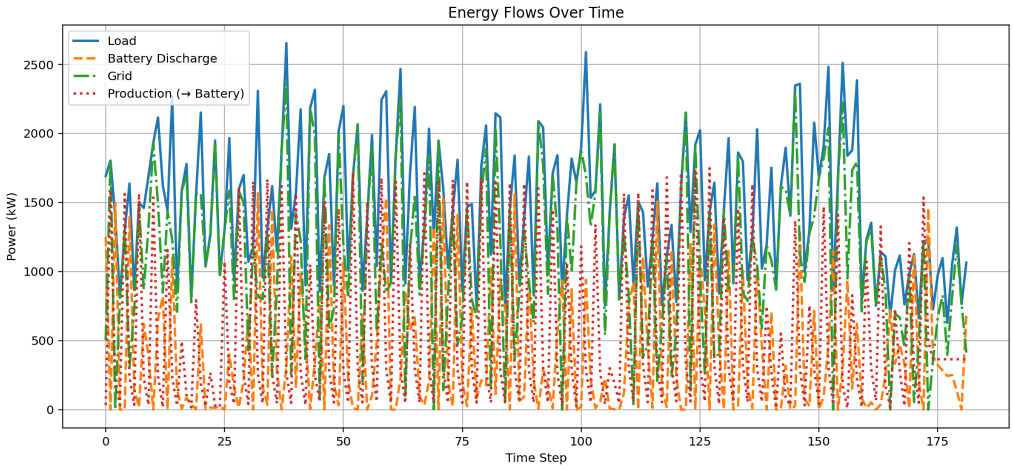

Figure 10: Energy Flow (PSO 1p1c)

PSO - Multi-Agent 14p51c

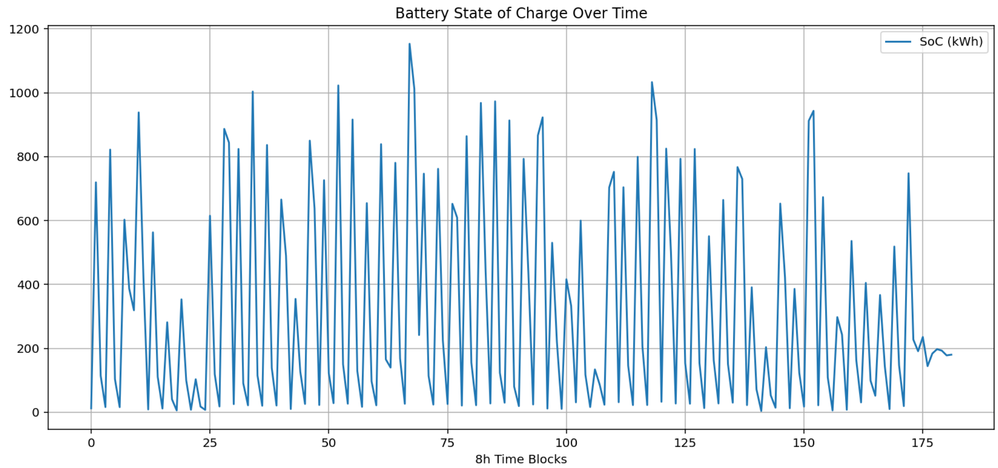

Figure 11: Battery State of Charge Over Time (PSO 14p51c)

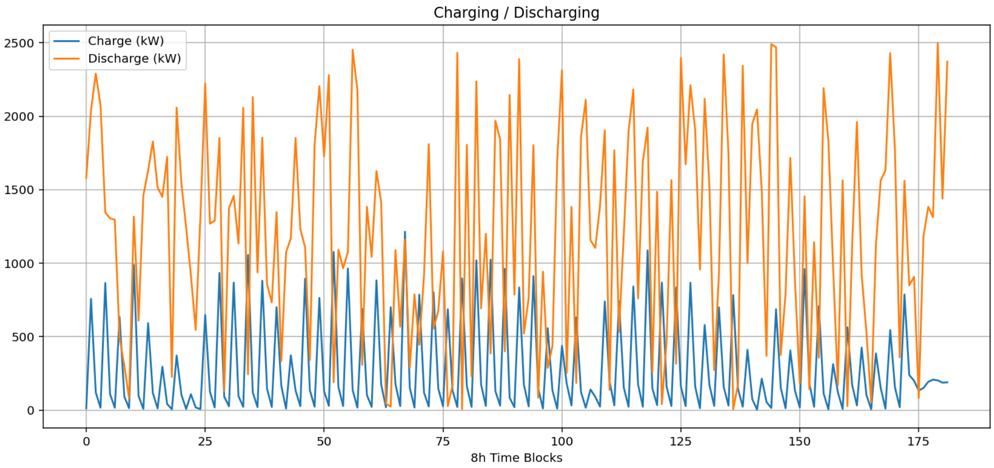

Figure 12: Total Charge and Discharge of Battery (PSO 14p51c)

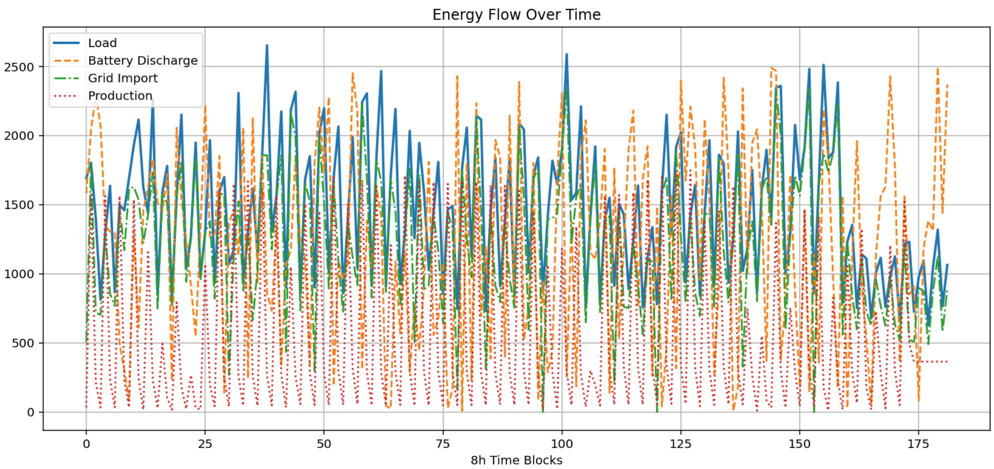

Figure 13: Energy Flow (PSO 14p51c)

Reinforcement Learning (RL)

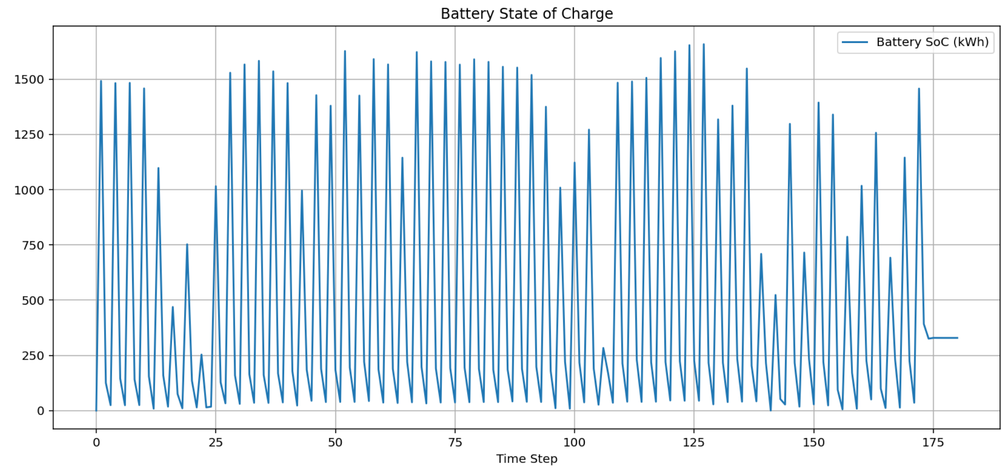

Figure 14: Battery State of Charge Over Time (RL)

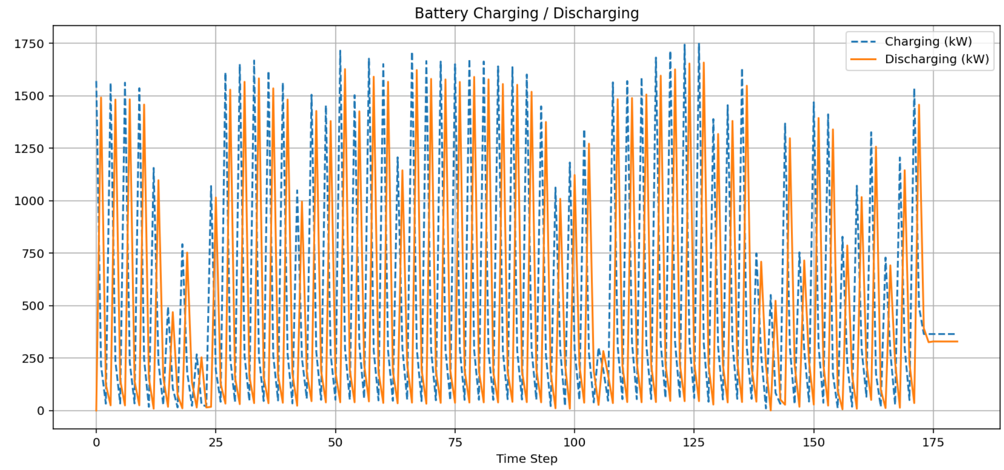

Figure 15: Total Charge and Discharge of Battery (RL)

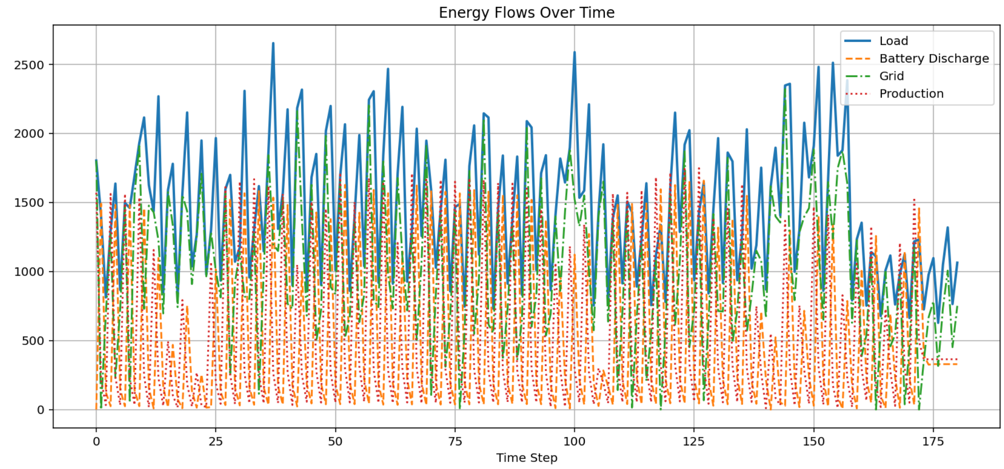

Figure 16: Energy Flow (RL)