



# การเรียนรู้แบบเสริมกำลังสำหรับ ระบบกักเก็บพลังงานด้วยแบตเตอรี่ Reinforcement Learning for BESS

จาก สเกด้า ออโตเมชั่น จำกัด



Part of the Scada Automation AI Initiative



# วัตถุประสงค์

- โค้ดนี้เป็นการสร้างโมเดล Q-learning สำหรับควบคุมระบบกักเก็บพลังงานด้วยแบตเตอรี่ (Battery Energy Storage System - BESS) โดยมีวัตถุประสงค์เพื่อ:
  - 1  ลดภาระโหลดจากกริด (Peak Shaving)
  - 2  ลดค่าไฟฟ้า โดยเรียนรู้ช่วงเวลาที่ดีที่สุดในการชาร์จและปล่อยพลังงานจากแบตเตอรี่อย่างเหมาะสม แบ่งเป็น ชาร์จ, ดีสชาร์จ และ ไม่ทำอะไรเลย

# ทำไมต้อง BESS และ Reinforcement Learning?

-  **ปัญหา:** การจัดการ BESS ซับซ้อนเกินกว่ามนุษย์จะทำได้ดี เพราะต้องประมวลผลข้อมูลมหาศาลและหลากหลายแบบเรียลไทม์ จากหลายปัจจัย เช่น ราคาไฟผันแปร, โหลด, สถานะประจุ (SOC), การเสื่อมของแบตเตอรี่ เพื่อหาจุดเหมาะสมที่สุดจากหลากหลายเป้าหมาย เช่น การลดค่าไฟ, การลดฟีด, และ การยืดอายุแบตเตอรี่
-  **ทางออก:** การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning: RL) สามารถช่วยเพิ่มประสิทธิภาพการทำงานของ BESS เพื่อประหยัดค่าใช้จ่าย และรักษาเสถียรภาพของระบบไฟฟ้า

# การตั้งค่า/โครงสร้างหลัก



- เวลาจำลอง: 24 ชั่วโมง (ตัดสินใจทุกชั่วโมง)
- สถานะการชาร์จแบตเตอรี่ (**State of Charge: SoC**): แบ่งเป็น 6 ระดับ (0%, 20%, 40%, 60%, 80%, 100%)
- การกระทำ:
  - ชาร์จ (Charge)
  - ดิสชาร์จ (Discharge)
  - ไม่ทำอะไร (Do Nothing)



# Data Inputs

- **ราคาไฟฟ้าตามช่วงเวลา (Time-of-Use: TOU):** ราคาไฟฟ้าเปลี่ยนแปลงตามแต่ละชั่วโมง
- **โปรไฟล์โหลด:** ความต้องการไฟฟ้าของผู้ใช้ในแต่ละชั่วโมง
- **เกณฑ์ภาระโหลดสูง (High Load Threshold):** ขีดจำกัดสูงสุดของโหลดไฟฟ้าที่ดึงจากระบบไฟฟ้า (เช่น 6 kW)

# RL ทำงานอย่างไร: รางวัลและบทลงโทษ

- เป้าหมาย: ฝึกฝนระบบให้ได้รับรางวัลสูงสุด
-  รางวัล:
  - ดิสชาร์จในช่วงราคาไฟฟ้าสูง
  - ดิสชาร์จในช่วงที่มีภาระโหลดสูง
  - ชาร์จในช่วงราคาไฟฟ้าต่ำ
-  บทลงโทษ:
  - ชาร์จในช่วงราคาไฟฟ้าสูง
  - ดิสชาร์จเมื่อแบตเตอรี่หมด
  - ภาระโหลดจากระบบไฟฟ้าเกินเกณฑ์ที่กำหนด

# The Q-Table: The Brain of the System (ตาราง Q: สมองของระบบ)

- **Q-Table คืออะไร:** คือตารางที่เก็บ "ค่า" ของแต่ละการกระทำในแต่ละสถานะ
  - สถานะ: (ชั่วโมง, ระดับ SoC ของแบตเตอรี่)
  - การกระทำ: (ชาร์จ, ดิสชาร์จ, ไม่ทำอะไร)
- **วัตถุประสงค์:** ช่วยให้ระบบเลือกการกระทำที่คาดว่าจะได้รับรางวัลสูงสุด



# วงจรการฝึก: ระบบเรียนรู้อย่างไร

1. **เริ่มต้น:** เริ่มต้นด้วยสถานะแบบเตอริแบบสุ่ม
2. **เลือกการกระทำ:** เลือกการกระทำโดยอ้างอิงจากตาราง Q (และสำรวจความเป็นไปได้ใหม่ๆ)
3. **คำนวณรางวัล:** กำหนดรางวัล (หรือบทลงโทษ) สำหรับการกระทำนั้น
4. **ปรับปรุงตาราง Q:** ปรับปรุงตาราง Q โดยอ้างอิงจากรางวัลที่ได้รับ
5. **ทำซ้ำ:** ทำกระบวนการนี้ซ้ำหลายๆ "รอบ" (วัน)





# การสร้างนโยบาย: การตัดสินใจ

- **นโยบาย:** ชุดของกฎที่กำหนดการกระทำที่ดีที่สุดสำหรับแต่ละสถานะที่เป็นไปได้ (ชั่วโมงและระดับ SoC)
- **สร้างอย่างไร:** โดยการวิเคราะห์ตาราง Q หลังจากฝึกฝน และเลือกการกระทำที่มีค่า Q สูงสุดสำหรับแต่ละสถานะ

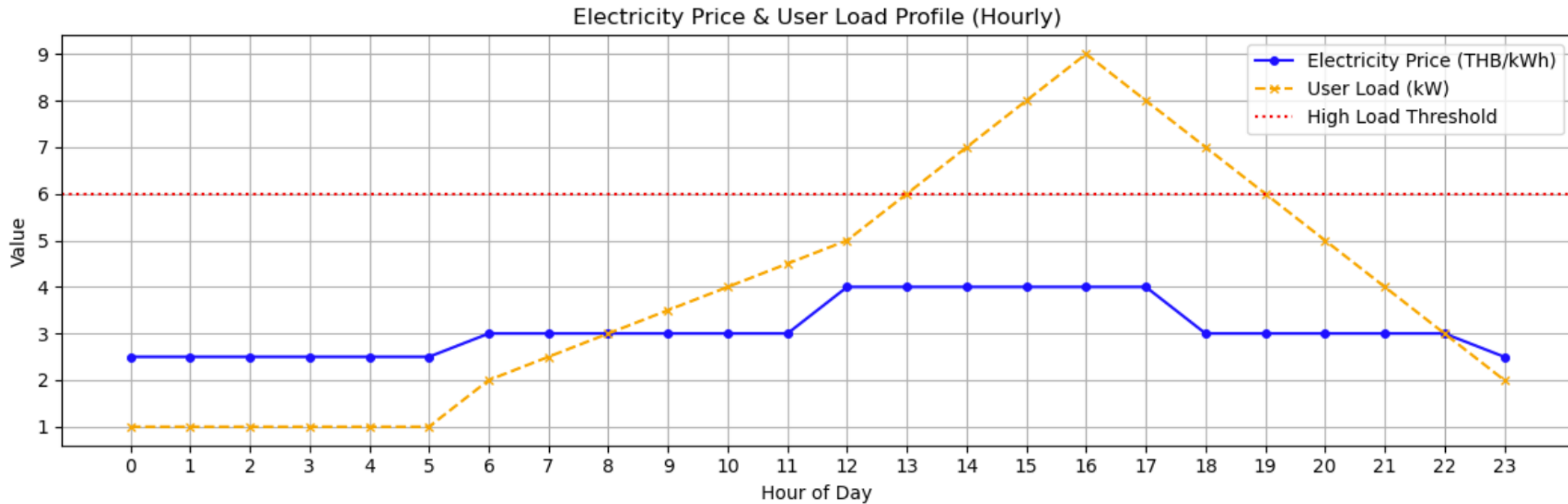
# การจำลองและแสดงผล

- **จำลอง:** รันระบบโดยใช้นโยบายที่ได้เรียนรู้
- **แสดงผล:**
  - ราคาไฟฟ้าและโปรไฟล์โหลด
  - สถานะการชาร์จของแบตเตอรี่ตามเวลา
  - ภาระโหลดบนระบบไฟฟ้า และกำลังไฟฟ้าที่มาจากแบตเตอรี่
  - นโยบายที่เหมาะสม (heatmap)

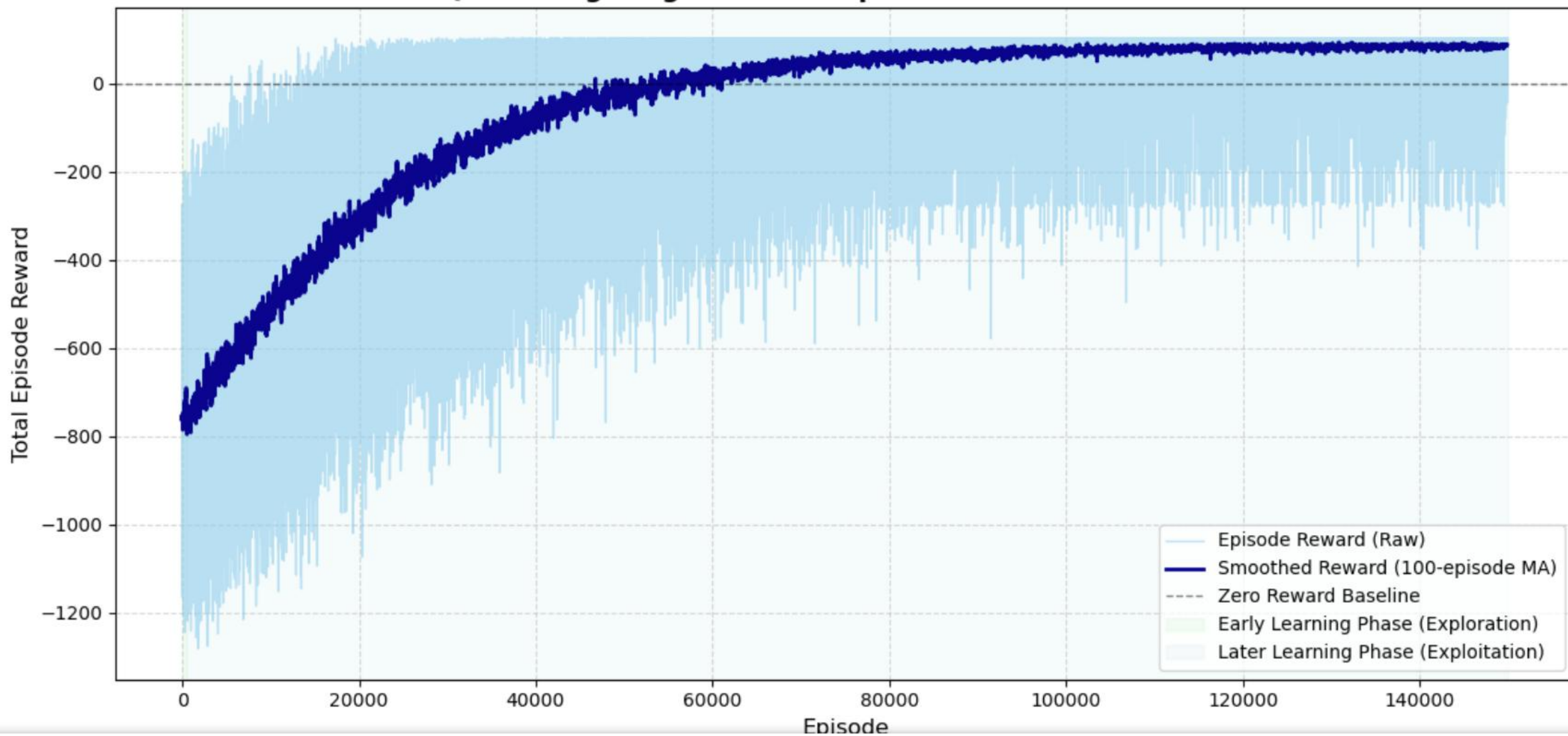
# Key Parameters(พารามิเตอร์หลัก)

- **การตั้งค่า BESS:**
  - ชั่วโมง, ระดับ SoC, การกระทำ
- **ราคาไฟฟ้าตามช่วงเวลา (TOU):**
  - ราคาในช่วง Off-Peak, Mid-Peak, On-Peak
- **โปรไฟล์โหลด:**
  - ความต้องการไฟฟ้าทั่วไปของผู้ใช้
- **Hyperparameters ของ Q-Learning:**
  - Episodes, Alpha, Gamma, พารามิเตอร์ Epsilon
- **ระบบรางวัล:**
  - อิงตามราคาไฟฟ้า, โหลด, และสถานะแบตเตอรี่

# กราฟค่าไฟฟ้า (TOU) โหลดโปรไฟล์ และ ค่าพีคเทรชโฮลด์



# Q-Learning Progress: Total Episode Rewards Over Time

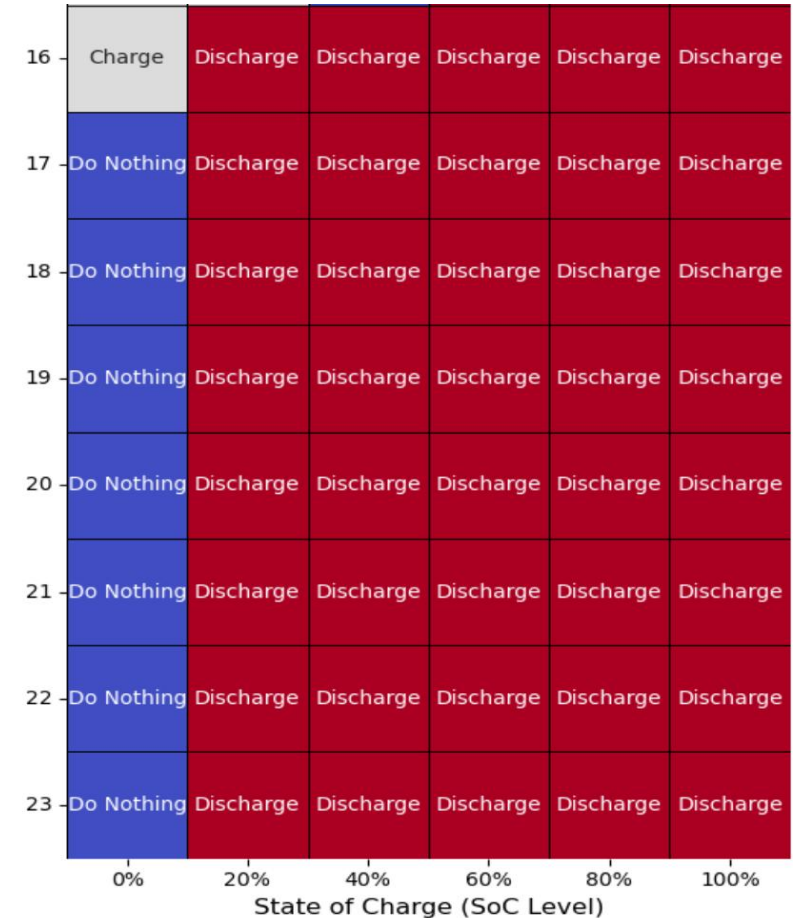
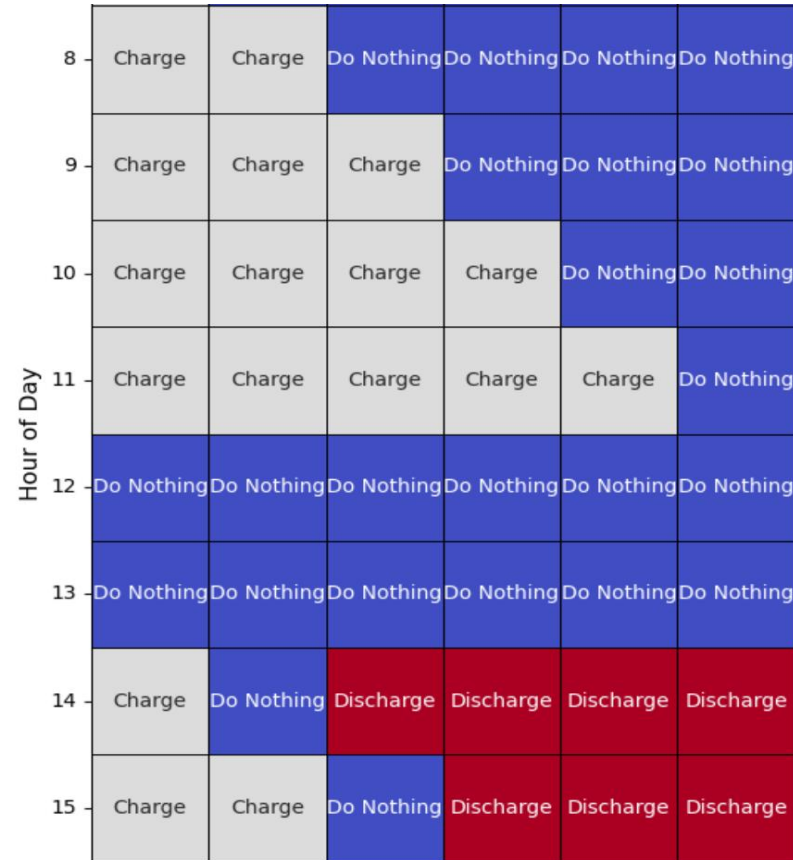
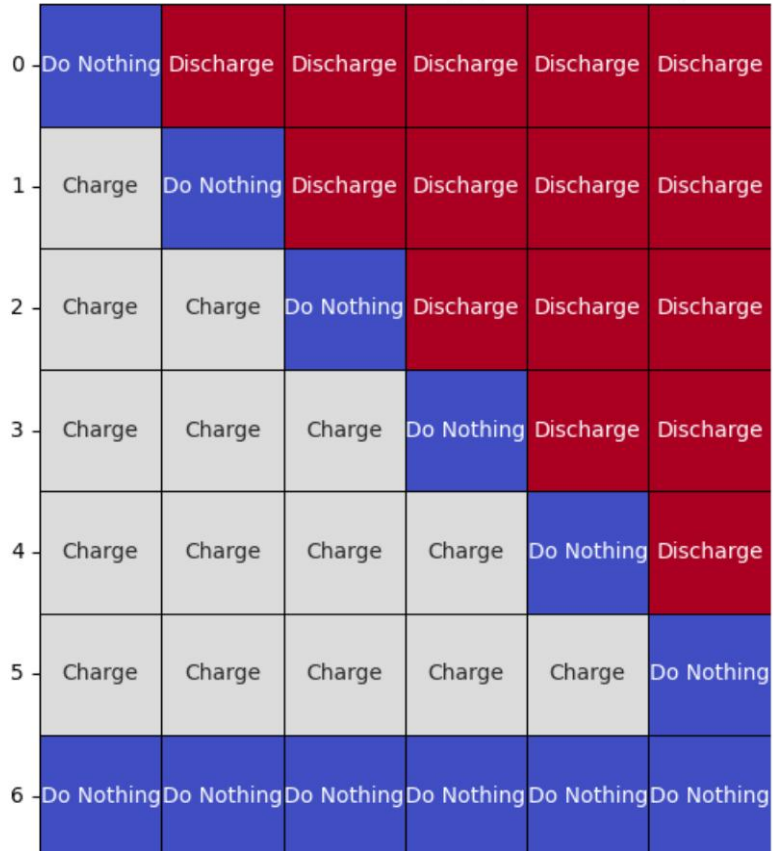


=== Optimal Policy (Hour x SoC) - With Enhanced Peak Shaving Focus ===

Hour	0% (Empty)	20%	40%	60%	80%	100% (Full)
0	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
1	Charge	Do Nothing	Discharge	Discharge	Discharge	Discharge
2	Charge	Charge	Do Nothing	Discharge	Discharge	Discharge
3	Charge	Charge	Charge	Do Nothing	Discharge	Discharge
4	Charge	Charge	Charge	Charge	Do Nothing	Discharge
5	Charge	Charge	Charge	Charge	Charge	Do Nothing
6	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing
7	Charge	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing
8	Charge	Charge	Do Nothing	Do Nothing	Do Nothing	Do Nothing
9	Charge	Charge	Charge	Do Nothing	Do Nothing	Do Nothing
10	Charge	Charge	Charge	Charge	Do Nothing	Do Nothing
11	Charge	Charge	Charge	Charge	Charge	Do Nothing
12	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing
13	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing	Do Nothing
14	Charge	Do Nothing	Discharge	Discharge	Discharge	Discharge
15	Charge	Charge	Do Nothing	Discharge	Discharge	Discharge
16	Charge	Discharge	Discharge	Discharge	Discharge	Discharge
17	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
18	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
19	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
20	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
21	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
22	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge
23	Do Nothing	Discharge	Discharge	Discharge	Discharge	Discharge

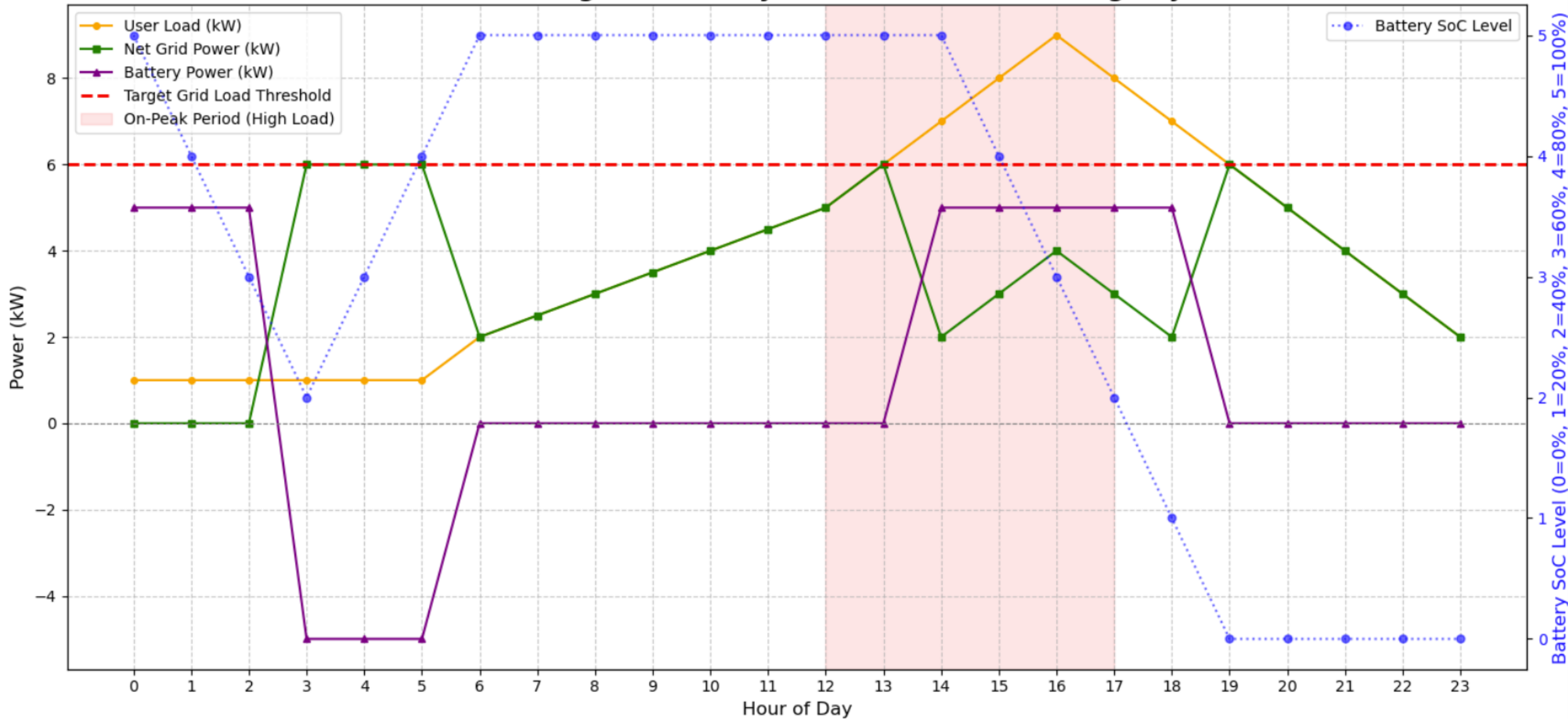
# ฮีทแมป (Heat Map)

Learned Optimal Policy (Hour vs. SoC)



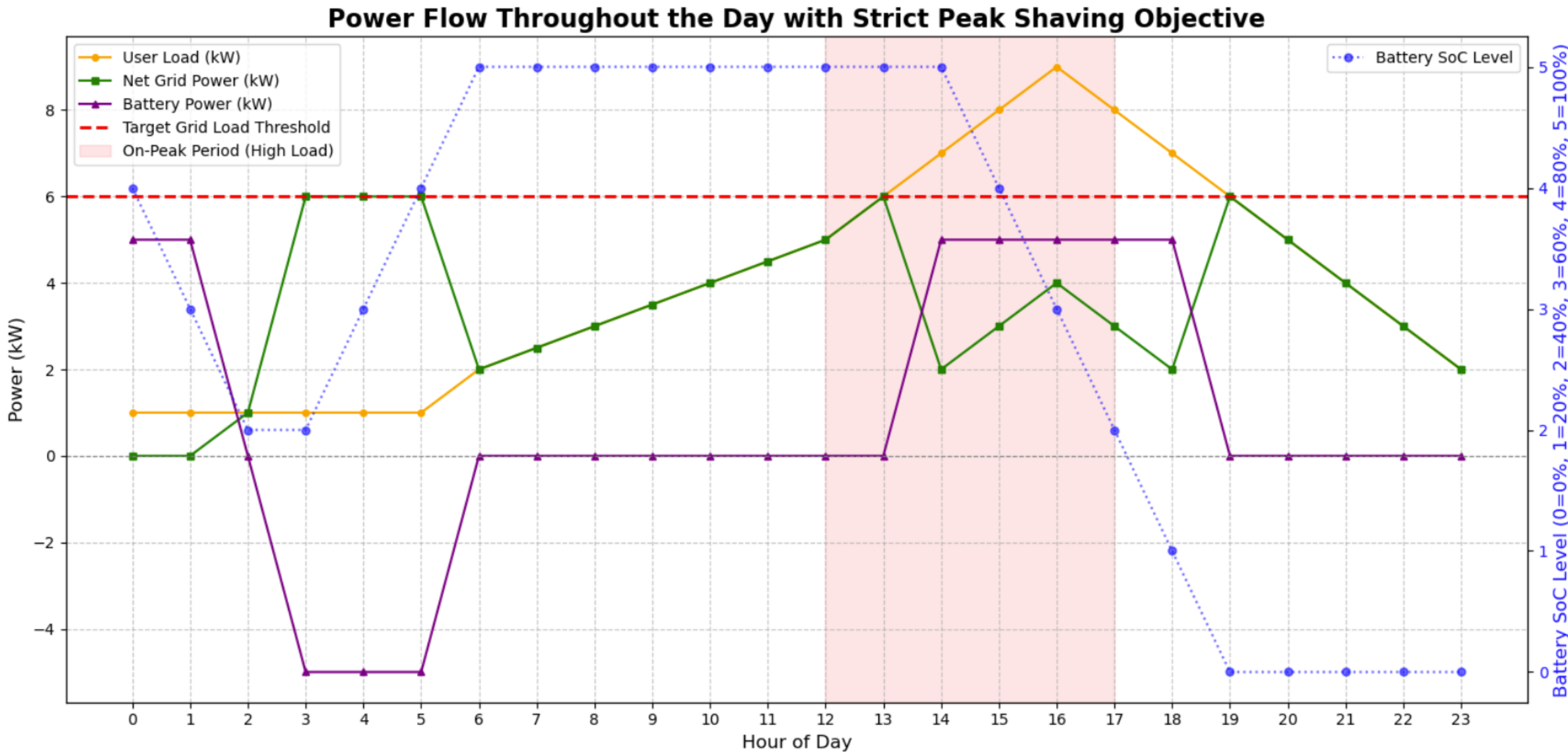
# SoC แบตเตอรี่เริ่มต้นที่ 100%

Power Flow Throughout the Day with Strict Peak Shaving Objective



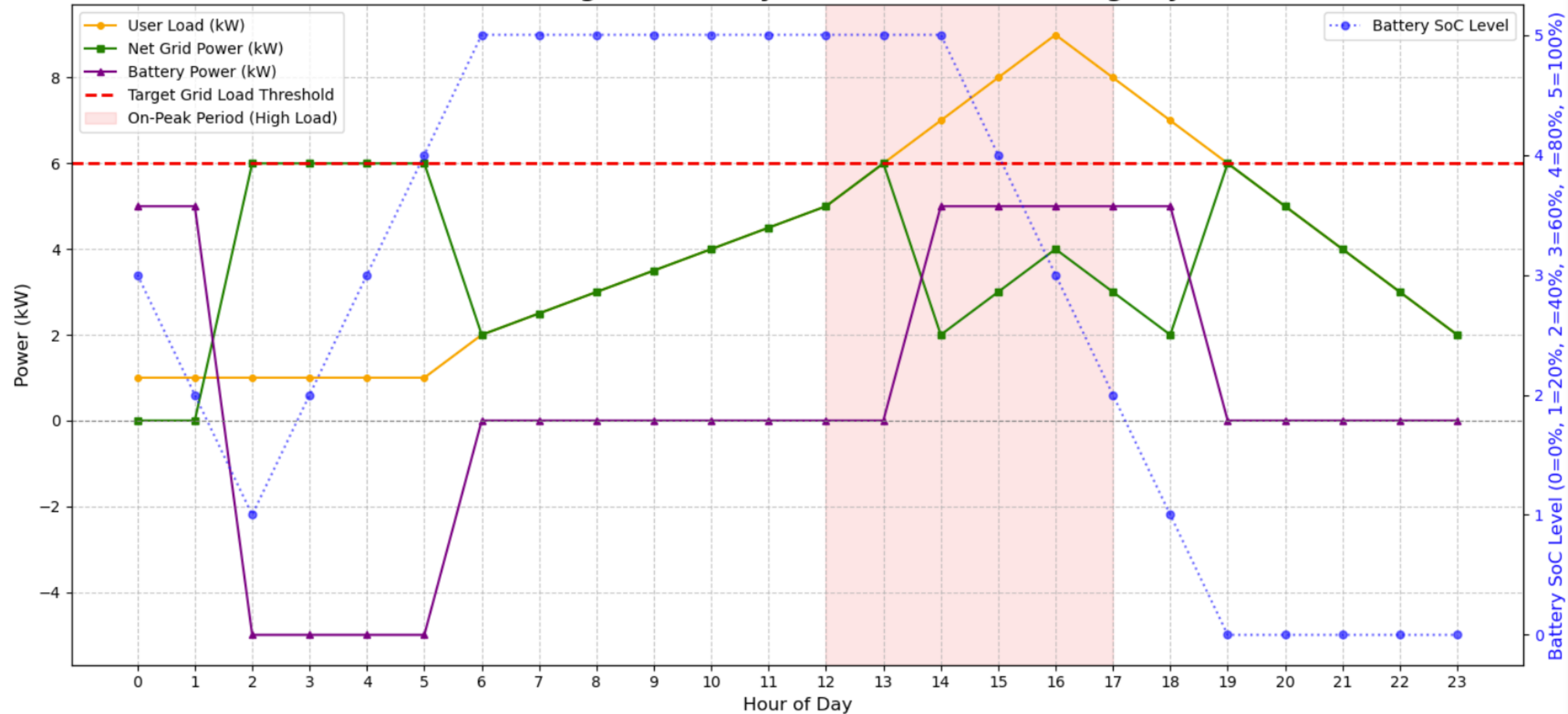


# SoC แบตเตอรี่เริ่มต้นที่ 80%



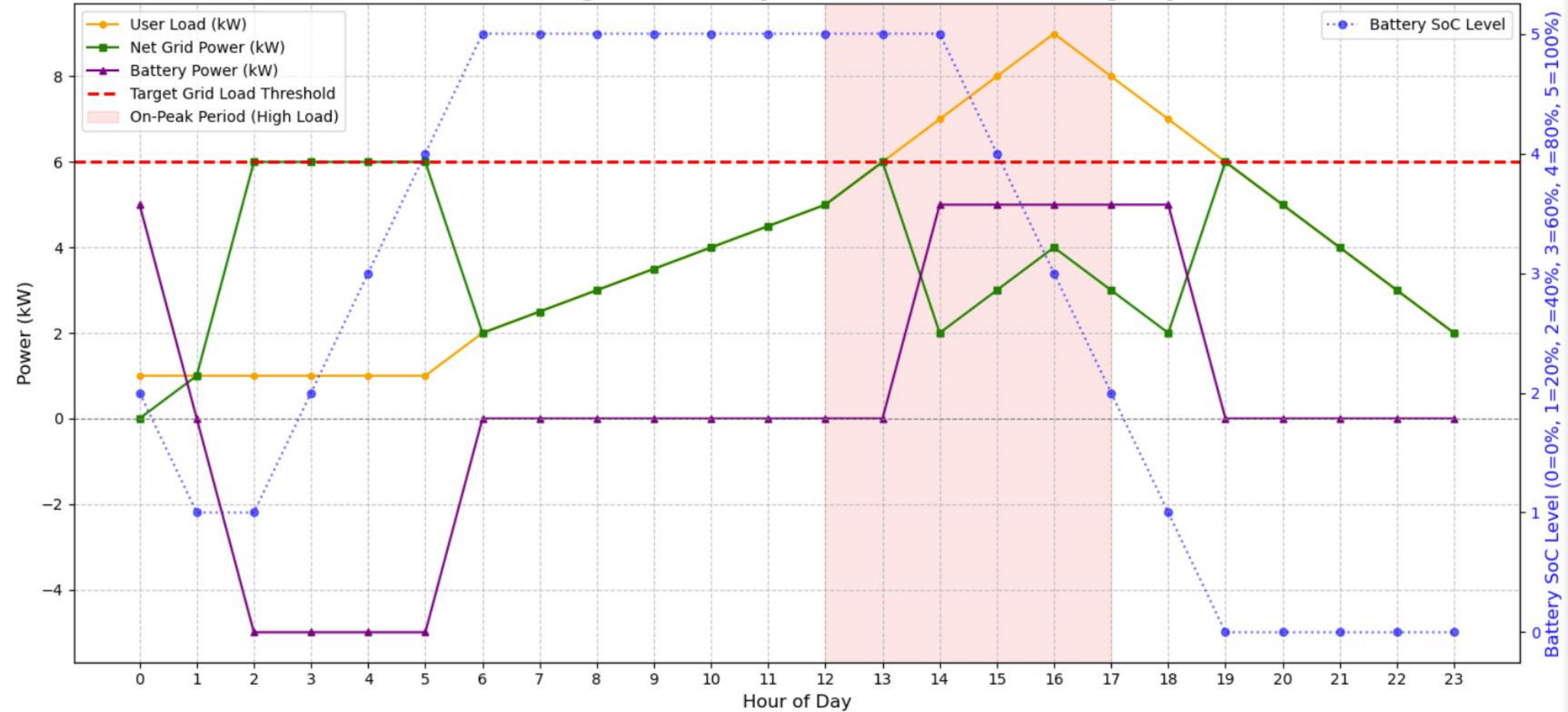
# SoC แบตเตอรี่เริ่มต้นที่ 60%

## Power Flow Throughout the Day with Strict Peak Shaving Objective



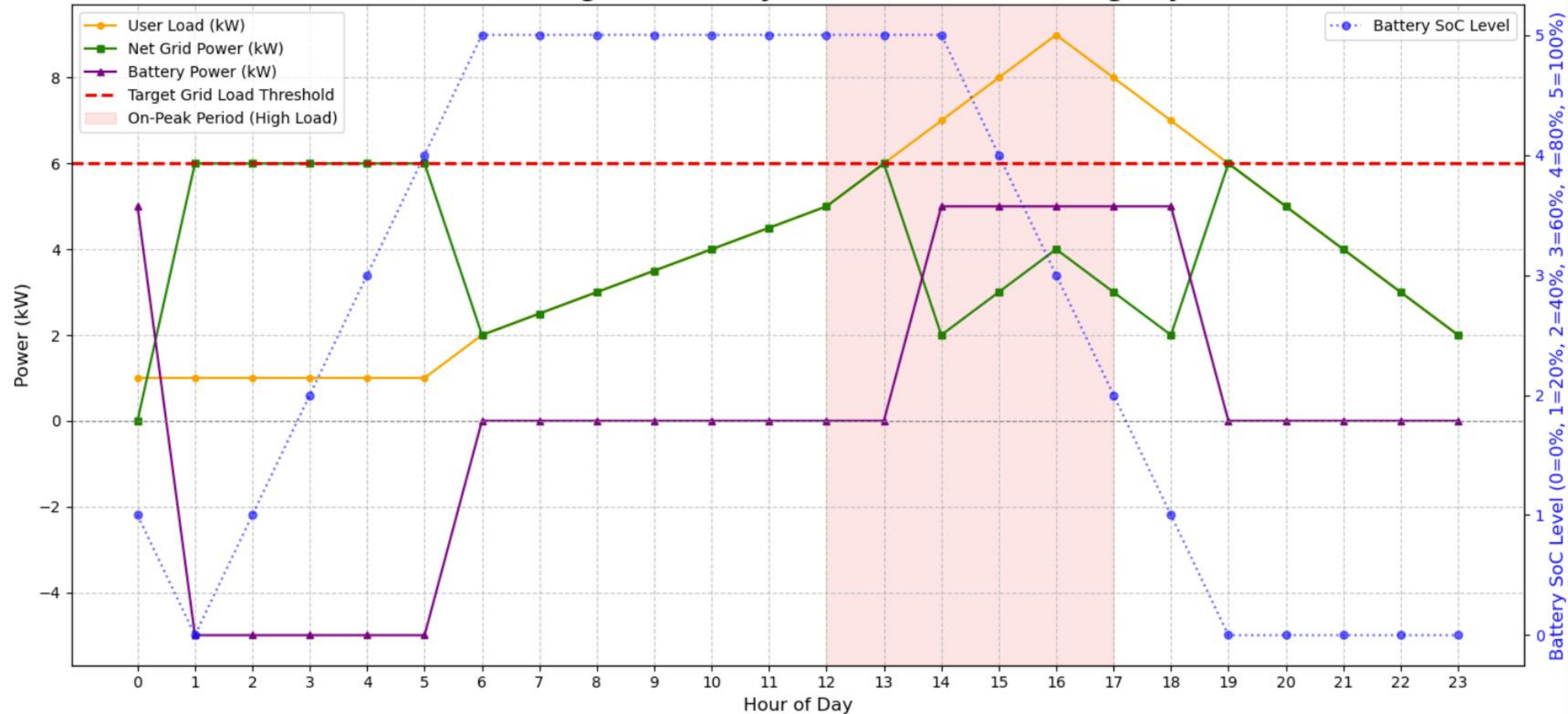
# SoC แบตเตอรี่เริ่มต้นที่ 40%

## Power Flow Throughout the Day with Strict Peak Shaving Objective



# SoC แบตเตอรี่เริ่มต้นที่ 20%

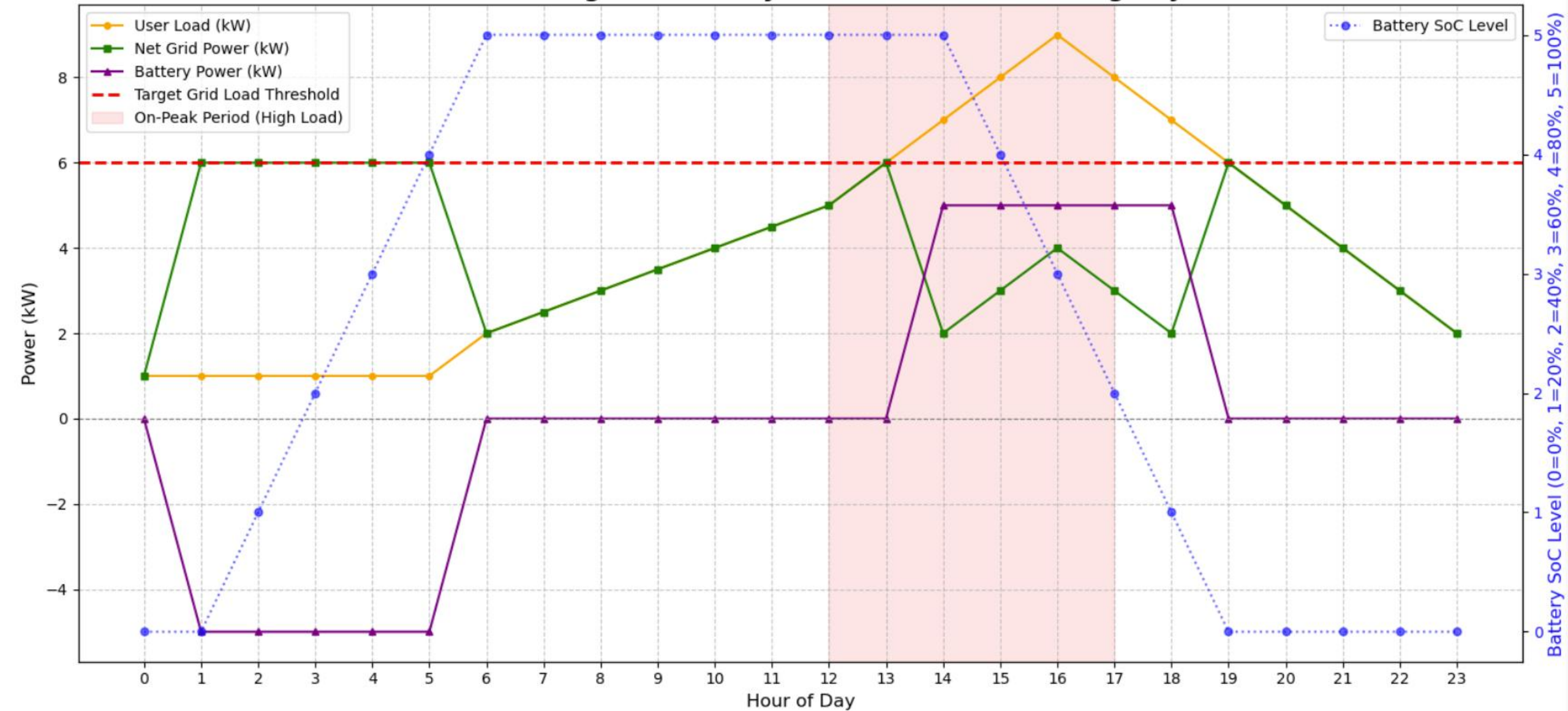
## Power Flow Throughout the Day with Strict Peak Shaving Objective





# SoC แบตเตอรี่เริ่มต้นที่ 0%

## Power Flow Throughout the Day with Strict Peak Shaving Objective



# บทสรุป

- Reinforcement Learning สามารถเพิ่มประสิทธิภาพการทำงานของ BESS เพื่อประหยัดค่าใช้จ่าย และรักษาสถียรภาพของระบบไฟฟ้า
- องค์ประกอบสำคัญ: ตาราง Q, ระบบรางวัล และวงจรการฝึกฝน
- หลังจากการฝึกฝน BESS สามารถทำงานได้โดยอัตโนมัติ โดยอิงตามนโยบายที่ได้เรียนรู้

# BESS\_Q\_learning\_PeakShv\_4June25.ipynb

```
jupyter BESS_Q_learning_PeakShv_4June25 Last Checkpoint: 2 minutes ago
File Edit View Run Kernel Settings Help
+ ✂ 📄 📋 ▶ ■ ↺ ▶▶ Code ▼

[4]: #Gemini
import numpy as np
import random
import matplotlib.pyplot as plt
import seaborn as sns

# === BESS Settings === #
HOURS = 24
# MODIFIED: 6 SoC Levels (0%, 20%, 40%, 60%, 80%, 100%)
SOC_LEVELS = 6
ACTIONS = ['Do Nothing', 'Charge', 'Discharge']
NUM_ACTIONS = len(ACTIONS)

# === Electricity price per hour (THB/kWh) - TOU Tariff in Thailand === #
electricity_prices = [
    2.50, 2.50, 2.50, 2.50, 2.50, 2.50, # Off-Peak
    3.00, 3.00, 3.00, 3.00, 3.00, 3.00, # Mid-Peak
    4.00, 4.00, 4.00, 4.00, 4.00, 4.00, # On-Peak - Highest prices
    3.00, 3.00, 3.00, 3.00, 3.00, 2.50 # Evening tapering
]
```

[https://github.com/DrHammerhead/SoC-estimation/blob/486bd3566e998cb2a33e49f06e9f11396f9c0919/BESS\\_Q\\_learning\\_PeakShv\\_4June25.ipynb](https://github.com/DrHammerhead/SoC-estimation/blob/486bd3566e998cb2a33e49f06e9f11396f9c0919/BESS_Q_learning_PeakShv_4June25.ipynb)

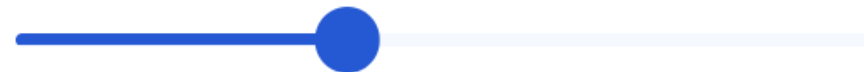
# สรุปเรื่องราวด้วย AI Podcast

Audio Overview



Create an Audio Overview in more languages! [Learn more](#)

เรียนรู้ BESS ด้วย Reinf...



03:04 / 07:59 • ไทย

[https://drive.google.com/file/d/1NoCDJck4oxsVhg4JkQUB\\_-dSteizWuR1/view?usp=sharing](https://drive.google.com/file/d/1NoCDJck4oxsVhg4JkQUB_-dSteizWuR1/view?usp=sharing)