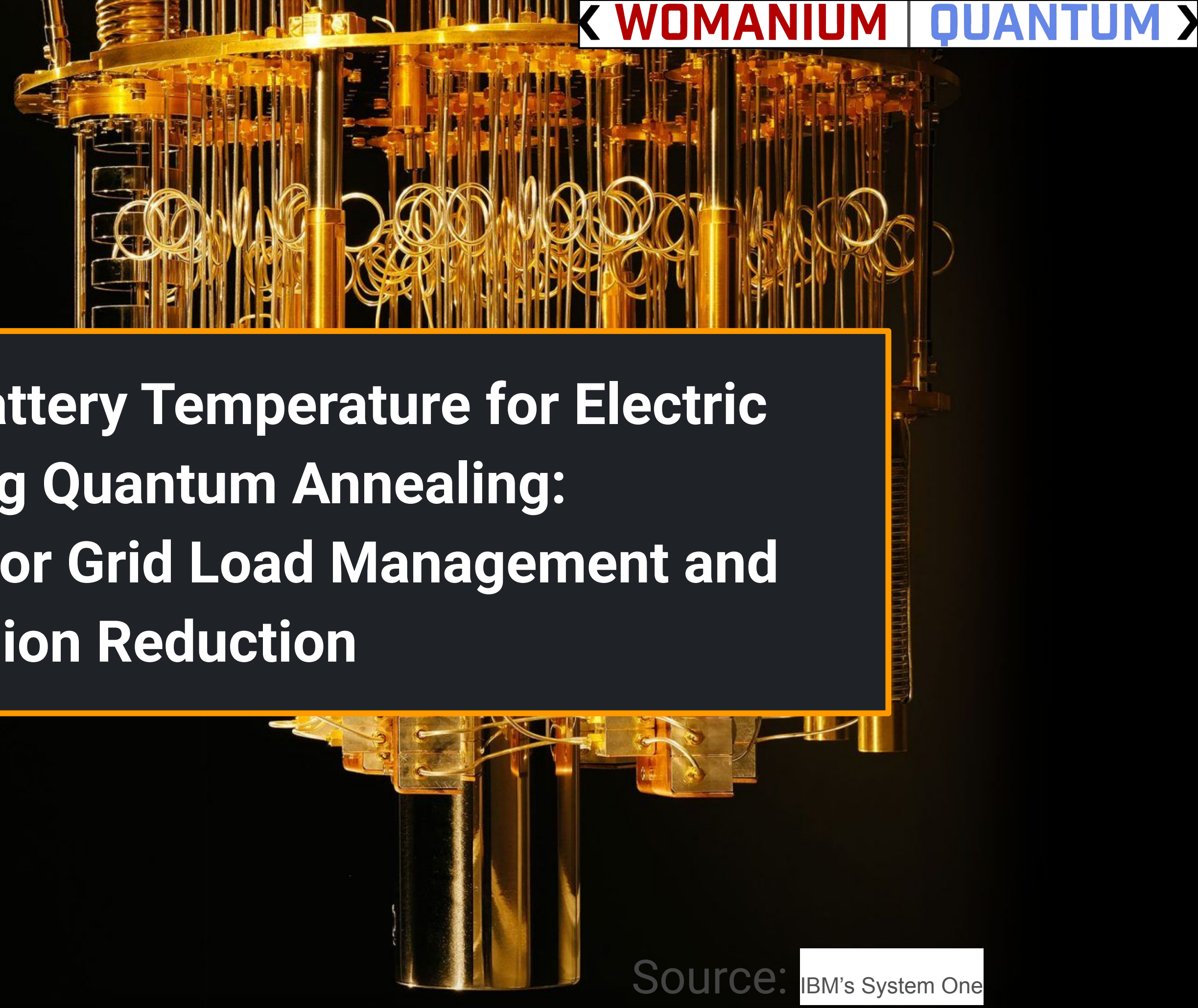
WOMANIUM | QUANTUM

Optimizing Battery Temperature for Electric Vehicles Using Quantum Annealing: Implications for Grid Load Management and Carbon Emission Reduction

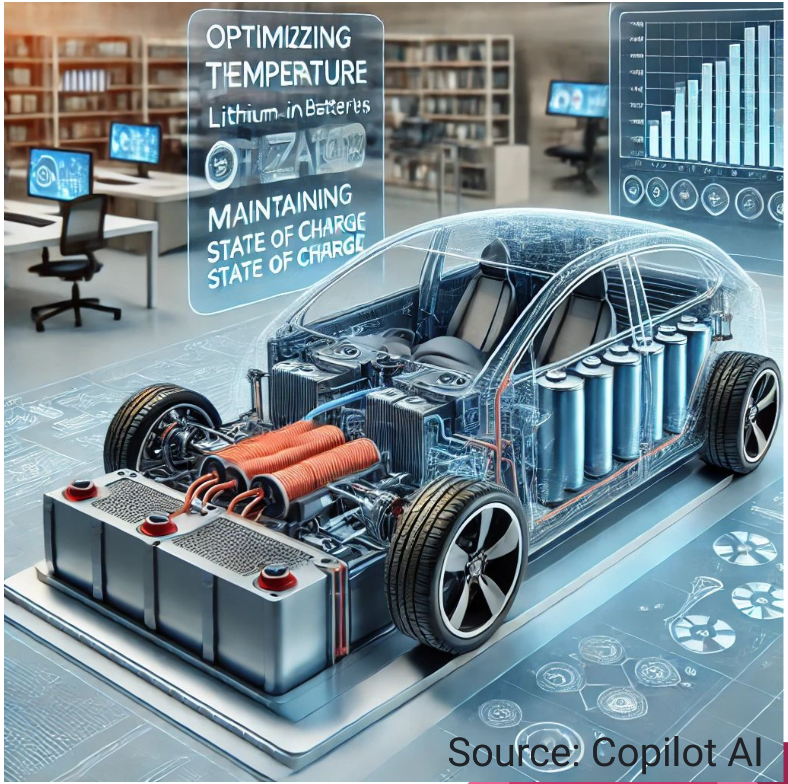


Source: IBM’s System One

Womanium Quantum+AI

Problem Statement

optimising battery temperature for lithium-ion batteries in electric vehicles (EVs) while maintaining an optimal State of Charge (SoC)



Source: Copilot AI

- **Battery Temperature: Affects the SoH and efficiency of EV batteries, directly impacting their lifespan**
- **State of Charge (SoC): Keeping SoC within optimal ranges is vital for efficient battery operation and user experience.**
- **Grid Load Management: Optimized battery performance can enable more effective smart charging strategies, reducing peak-time grid stress.**
- **Carbon Emissions: Efficient battery management can lead to better utilization of low-carbon energy sources, aiding in the reduction of carbon emissions.**

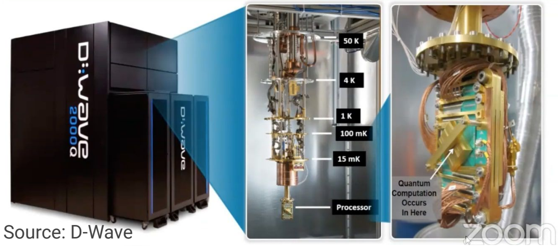
What is your project about?

The project aims to optimize battery management systems for Li-Ion batteries using Quantum Annealing, focusing on enhancing efficiency and performance by optimizing battery temperature and improving State of Health (SoH) estimation.

What problem are you trying to solve?

The problem we are trying to solve is finding the optimal battery temperature while maintaining State of Charge (SoC) levels using Quantum Annealing.

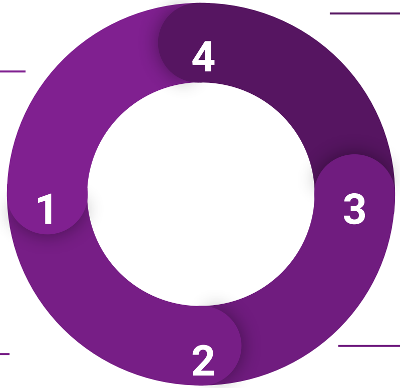
Project Solution

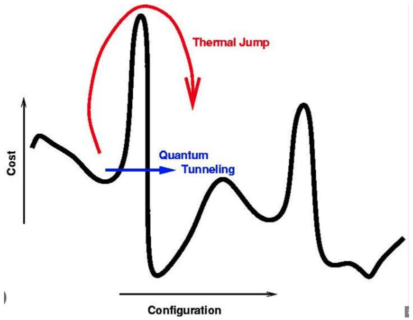


Source: D-Wave

Quantum Annealing (D-Wave).

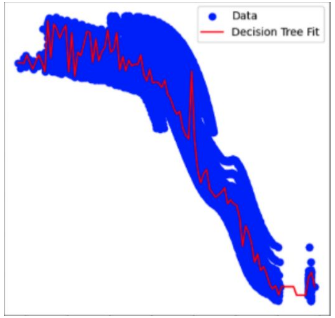
Data Initialization





Source: Quantum Stack Exchange

Objective Function using Regression Models



QUBO Formulation

1. **Data Preparation**: By converting NASA battery datasets from MATLAB to CSV format, and processed the data for SoC calculation.

```
def load_data(battery):
  file_path = '/content/battery_data/B0005.mat'
  if not os.path.exists(file_path):
    raise FileNotFoundError(f"File not found: {file_path}")  # Raise an er

  mat = scipy.io.loadmat(file_path)
  print('Total data in dataset: ', len(mat[battery][0, 0]['cycle'][0]))
  counter = 0
  dataset = []
  capacity_data = []

  for i in range(len(mat[battery][0, 0]['cycle'][0])):
    row = mat[battery][0, 0]['cycle'][0, i]
    if row['type'][0] == 'discharge':
      ambient_temperature = row['ambient_temperature'][0][0]
```

2. **Objective Function Formulation**: By using statistical method to derive the objective function for the QUBO problem, targeting temperature minimization.

$$\mathrm{Temp}(V, I, t) = -16.3967 + 39.4782 \cdot V + 7.3247 \cdot I + 0.0056 \cdot t - 7.0293 \cdot V^2 + 0.5705 \cdot V \cdot I - 0.0011 \cdot V \cdot t + 3.3260 \cdot I^2 - 0.0004 \cdot I \cdot t - 0.0000 \cdot t^2$$

3. **QUBO Formulation**: Developed the Q matrix for the quantum annealing process.

$$Q = \begin{pmatrix} 32.4489 & 0.28525 & -0.00055 \\ 0.28525 & 10.6507 & -0.0002 \\ -0.00055 & -0.0002 & 0.0056 \end{pmatrix}$$

4. **Quantum Annealing**: Currently working on embedding the objective function for D-Wave's quantum annealer.

Solving Objective Function using D-Wave Solver - BQM Technique

In [ ]:

```
from dwave.system import DWaveSampler, EmbeddingComposite
from dimod import BinaryQuadraticModel
```

In [ ]:

```
from dwave.system import DWaveSampler, EmbeddingComposite
from dimod import BinaryQuadraticModel

# Assume Q_dict and c are already defined

# Create BQM
bqm = BinaryQuadraticModel.from_qubo(Q_dict, offset=c)

# Set up sampler
sampler = EmbeddingComposite(DWaveSampler())

# Run quantum annealing
num_reads = 10000
sampleset = sampler.sample(bqm, num_reads=num_reads)

# Analyze results
best_solution = sampleset.first.sample
best_energy = sampleset.first.energy

print("Best solution:", best_solution)
print("Energy of best solution:", best_energy)
```

Success

While our project is still in progress, the project has the potential to:

1. **Extend Battery Life and Improve EV Performance**: Better temperature management could lead to longer battery life and enhanced EV performance.
2. **Optimize Smart Charging Strategies**: More accurate SoC,SoH estimation can lead to optimized charging, reducing grid load during peak hours.
3. **Reduce Carbon Emissions**: By optimizing the use of low-carbon energy sources during charging, carbon emissions could be significantly lowered.

Success Metrics: We achieved about 75% progress in applying Quantum Annealing to optimize battery temperature while maintaining State of Charge (SoC).

Next Steps in the project…

- **Compare Classical, AI, and Quantum Methods**: Assess the performance of different methods in optimizing SoH and SoC.
- **Optimize Constraints and Penalties:** Fine-tune the QUBO formulation for better results.
- **Expand Literature Review**: Continue to explore emerging research in battery management and quantum optimization.

Future Scope

Key Areas for Future Development:

1. **Refinement of QUBO Formulation**: We will enhance the QUBO formulation by incorporating additional constraints and refining penalty terms to better capture the complexities of battery temperature and SoC management.

2. **Optimization of Quantum Annealing Setup**: Further optimization of the quantum annealing process will be conducted to improve the accuracy and efficiency of the solutions obtained. This will involve fine-tuning the embedding process and exploring advanced quantum algorithms.