Specialisation Project (VT1)
HS2024

Platform for Investment Analysis

Linear Programming Optimization Model for Energy Asset Management in Python

Submitted by
Rui Vieira
Institute of Product Development and Production Technologies (IPP)

Supervisor
Dr. Andrea Giovanni Beccuti
IEFE Model Based Process Optimisation

Study Program
Business Engineering, MSc in Engineering

Zurich University of Applied Sciences
School of Engineering

January 29, 2025

Imprint

Project: Specialisation Project (VT1)
Title: Platform for Investment Analysis
Author: Rui Vieira
Date: January 29, 2025
Keywords: linear programming, quantitative modeling, python, strategic planning, risk assessment, portfolio optimization, asset valuation, power-flow, platform

Study program:
Business Engineering, MSc in Engineering
ZHAW School of Engineering

Supervisor:
Dr. Andrea Giovanni Beccuti
IEFE Model Based Process Optimisation
Email: giovanni.beccuti@zhaw.ch

Abstract

This project presents a platform for investment analysis in energy systems, focusing on the optimization of asset management through linear programming techniques. The framework integrates DC Optimal Power Flow (DC-OPF) for network analysis with economic evaluation methods to support investment decisions in energy infrastructure.

The platform, implemented in Python, combines power system modeling with financial analysis tools to evaluate different investment scenarios. It features automated scenario generation, optimization of operational costs, and AI-assisted analysis of results. The methodology incorporates both technical constraints from power system operations and financial metrics such as Net Present Value (NPV) to provide comprehensive decision support.

Key features include modular architecture for extensibility, integration with industry-standard optimization solvers, and a flexible scenario analysis framework. The platform demonstrates practical applicability through case studies in energy infrastructure investment, showing how it can be used to evaluate complex investment decisions while considering both technical and economic factors.

Keywords: linear programming, quantitative modeling, python, strategic planning, risk assessment, portfolio optimization, asset valuation, power-flow, platform

Contents

1 Introduction

1.1 Project Context

1.2 Objectives

- Develop a Python-based platform for energy investment analysis
- Implement linear programming optimization for asset management
- Investment analysis

2 Theoretical Background

2.1 Linear Programming in Energy Systems

At its core, our problem focuses on meeting electricity demand through optimal generation dispatch: determining how much power each generator should produce to satisfy consumer demand while minimizing costs and respecting system constraints. This fundamental power systems challenge can be effectively modeled using Linear Programming (LP).

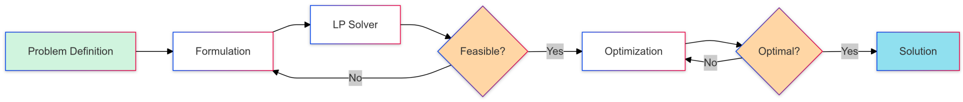

Figure 1: LP optimization flowchart showing key steps from problem formulation to optimal solution.

Linear Programming enables modeling of key power system relationships - such as power balance(matching generation to demand), transmission limits, and generation constraints - as linear equations and inequalities. The basic structure of our optimization problem is:

- **Objective**: Minimize total generation cost
- **Primary Decision**: How much power to generate at each plant
- **Key Constraint**: Total generation must meet demand at all times
- **System Constraints**: Respect network and equipment limitations

This can be expressed mathematically as:

$$\min_{\mathbf{x}} \mathbf{c}^\top \mathbf{x} \quad \text{subject to} \quad A\mathbf{x} \leq \mathbf{b} \tag{1}$$

2.2 DC Optimal Power Flow

The DC Optimal Power Flow (DC-OPF) extends the basic generation dispatch problem by incorporating network constraints. It answers the question: "How should we distribute power generation across the network to meet demand at minimum cost while respecting transmission line limits?" The DC-OPF achieves this by:

- Modeling power flow through transmission lines
- Ensuring power balance at each network node
- Respecting both generation and transmission limits

The "DC" prefix indicates a linearized approximation of the full AC power flow equations, making the problem solvable using LP techniques [7]. This approximation is particularly effective for high-voltage transmission planning [1].

2.2.1 Key Assumptions

The DC approximation makes four key simplifications:

- Voltage magnitudes are fixed at 1.0 per unit

- Line resistances are negligible ($R \ll X$)
- Voltage angle differences are small
- Reactive power (power that oscillates between source and load without doing useful work) is ignored

These assumptions yield a simple relationship between power flow (P_{ij}) and voltage angles (θ):

$$P_{ij} = B_{ij}(\theta_i - \theta_j) \tag{2}$$

2.2.2 Mathematical Formulation

The DC-OPF problem minimizes generation costs subject to network constraints. Each equation represents a physical aspect of power system operation:

1. Power Balance - The fundamental law of power systems

- At each bus i, power in equals power out
- Generation minus demand equals net power flow to neighboring buses
- Determined by line susceptances and voltage angles

$$\sum_{g \in \mathcal{G}_i} P_{g,t} - D_{i,t} = \sum_{j \in \mathcal{N}_i} B_{ij}(\theta_{i,t} - \theta_{j,t}) \quad \forall i \in \mathcal{N}, t \in \mathcal{T} \tag{3}$$

2. Cost Minimization - Economic objective

- Find optimal generation dispatch that minimizes total system cost
- Each generator has an associated marginal cost function (cost per unit of production)

$$\min_{\mathbf{P_g}, \boldsymbol{\theta}} \sum_{g \in \mathcal{G}} \sum_{t \in \mathcal{T}} c_g P_{g,t} \tag{4}$$

3. Line Capacity - Network limitations

- Power flow must stay within thermal limits of transmission lines
- Bi-directional constraint (forward and reverse flow limits)

$$-P_{ij}^{\max} \leq B_{ij}(\theta_{i,t} - \theta_{j,t}) \leq P_{ij}^{\max} \quad \forall (i,j) \in \mathcal{L}, t \in \mathcal{T} \tag{5}$$

4. Generator Limits - Physical constraints

- Each generator is limited by minimum and maximum output
- Generators may have time-varying limits

$$P_g^{\min} \leq P_{g,t} \leq P_g^{\max} \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \tag{6}$$

5. Reference Angle - System reference

- One bus sets the reference for voltage angles
- Typically chosen as the largest generator

$$\theta_{\text{slack},t} = 0 \quad \forall t \in \mathcal{T} \tag{7}$$

2.2.3 Storage System Constraints

The model includes battery storage systems with the following constraints:

1. Energy Balance - Storage state evolution

- Tracks energy level over time
- Accounts for charging and discharging efficiencies

$$E_{s,t+1} = E_{s,t} + \eta_c P_{c,s,t} - \frac{P_{d,s,t}}{\eta_d} \quad \forall s \in \mathcal{S}, t \in \mathcal{T} \tag{8}$$

2. Power Limits - Operational boundaries

- Maximum charging and discharging rates
- Cannot charge and discharge simultaneously

$$0 \leq P_{c,s,t} \leq P_{c,s}^{\max} \quad \forall s \in \mathcal{S}, t \in \mathcal{T} \tag{9}$$

$$0 \leq P_{d,s,t} \leq P_{d,s}^{\max} \quad \forall s \in \mathcal{S}, t \in \mathcal{T} \tag{10}$$

3. Energy Capacity - Storage limits

- Maximum and minimum state of charge
- Often includes end-state condition

$$E_s^{\min} \leq E_{s,t} \leq E_s^{\max} \quad \forall s \in \mathcal{S}, t \in \mathcal{T} \tag{11}$$

$$E_{s,T} = E_{s,0} \quad \forall s \in \mathcal{S} \tag{12}$$

Where:

- $E_{s,t}$: Energy stored in battery s at time t
- $P_{c,s,t}, P_{d,s,t}$: Charging and discharging power
- η_c, η_d: Charging and discharging efficiencies

2.2.4 Implementation Considerations

The model is implemented in Python using PuLP [5], an open-source optimization framework, with the COIN-OR Branch and Cut (CBC) solver [3]. As a first programming project, the focus was on developing a functional and flexible platform rather than computational optimization. This approach prioritized code clarity and feature implementation over algorithmic efficiency.

Howevere, our implementation processes hourly data for generator availability and demand profiles. For a full year analysis, this represents over 8,700 objective functions, each generating multiple constraints per time period. Real power systems often require even finer temporal resolution (15-minute intervals), more elaborate constraints. This is a simplified model but grasps the core principles of the problem.

To manage computational complexity, we use representative weeks selection for seasonal patterns. Involves some uncertainty, but allows meaningful analysis while improving computing time by a factor 20-30x. Future work could focus on algorithmic improvements and computational optimization.

3 Methodology and Implementation

3.1 System Architecture

Description of the linear programming implementation.

3.2 Core Components

How different scenarios are generated and compared.

3.3 Technical Implementation

Implementation of AI-powered analysis features.

4 Results and Validation

4.1 Test Cases

Description of validation scenarios.

4.2 Performance Analysis

Computational efficiency and scalability.

4.3 Case Studies

Real-world applications and insights.

5 Discussion

5.1 Platform Capabilities

Current functionality and limitations.

5.2 Future Improvements

Potential enhancements and extensions.

6 Conclusion

[6] [1] [7] [4] [2]

A Code Model

A.1 Optimization Model

A.2 Solver

A.3 Economic Calculations

A.4 AI Integration

A.5 Report Generation

References

[1] Göran Andersson. Modelling and analysis of electric power systems. Lecture Notes 227-0526-00, ETH Zürich, Zürich, Switzerland, March 2004. Power Systems Laboratory, ETH Zürich.

[2] T. Brown, J. Hörsch, and D. Schlachtberger. PyPSA: Python for Power System Analysis. *Journal of Open Research Software*, 6(4), 2018.

[3] John Forrest and Robin Lougee-Heimer. Cbc user guide. *COIN-OR Foundation*, 2018. Open-source mixed integer programming solver.

[4] MATPOWER Development Team. Matpower documentation, 2024. MATPOWER: A MATLAB Power System Simulation Package.

[5] Stuart Mitchell, Michael O'Sullivan, and Iain Dunning. Pulp: A linear programming toolkit for python. *Optimization Online*, 2011. Software package.

[6] B Stott, J Jardim, and O Alsaç. Dc power flow in analysis of power networks. *IEEE Transactions on Power Systems*, 24(3):1290–1300, 2009.

[7] Allen J Wood, Bruce F Wollenberg, and Gerald B Sheblé. *Power Generation, Operation, and Control.* John Wiley & Sons, 2013.