ICT in Building Design

Master Degree in ICT4SS, Politecnico di Torino

Arezou Shadkam	327085
Kiyana Mehdinejad	328655
Ali Abdollahzadeh	327783
M.Mahdi Azizian	324836

Supervisors:
Prof. Giacomo Chiesa
Prof. Lorenzo Bottaccioli

A.Y. 2024 - 2025

Contents

Abstract

Buildings are significant global energy consumers, demanding advanced methods for energy-efficient design and operation. This project details a comprehensive Information and Communication Technologies (ICT) workflow applied to an office building, encompassing design optimization, dynamic simulation, and real-time operational analysis. The methodology integrates parametric EnergyPlus simulations with machine learning-based surrogate models, including Deep Neural Networks (DNNs), to perform multi-objective optimization (NSGA-II) and identify an energy-optimal building configuration. This optimized design was converted into a Functional Mock-up Unit (FMU) for co-simulation in a Python environment, assessing its dynamic performance. A containerized data pipeline utilizing MQTT, InfluxDB, and Grafana facilitated real-time data streaming and visualization of key performance indicators. Furthermore, energy signature analysis characterized the building's energy consumption patterns. The study successfully demonstrates an end-to-end ICT framework capable of achieving significant design improvements and enabling continuous operational monitoring, thereby offering valuable insights for sustainable building practices.

Keywords: Building Energy Optimization, Surrogate Models, Functional Mock-up Unit (FMU), Co-simulation, Real-time Monitoring, EnergyPlus, NSGA-II, ICT in Building Design, Energy Signature Analysis, Deep Neural Networks (DNN).

1 Introduction

This section provides an overview of the context, objectives, and current state of research relevant to the project.

1.1 Background

Buildings are responsible for a large share of global energy use and greenhouse gas emissions. According to the International Energy Agency, the building and construction sector accounted for more than one-third of global final energy consumption in 2022 [1]. Because of this, designing buildings that use less energy is important for creating a more sustainable future.

Information and Communication Technologies (ICT) now play a key role in building design. With simulation tools like EnergyPlus, engineers and designers can model how a building uses energy [3]. This helps them test different design choices before the building is actually built. However, running many simulations takes a lot of time and computing power.

To solve this, machine learning models—called surrogate models—are used. These models learn from a few simulation runs and can then predict results much faster. Popular surrogate models include Gaussian Processes [4] and Neural Networks [5]. These models make it easier to explore and optimize building designs.

Another useful technology is the Functional Mock-up Unit (FMU), which allows simulation models to run in real-time environments and connect with control systems [7]. Together with platforms like InfluxDB and Grafana, this allows real-time monitoring and data analysis of building performance [9].

1.2 Objectives

This project aims to:

- Design and optimize an energy-efficient office building using simulation and data analysis [3].
- Use surrogate models such as Gaussian Processes and Neural Networks to reduce the cost of running many simulations [4, 5].
- Convert the optimized design into a Functional Mock-up Unit (FMU) for real-time simulation [7,10].
- Set up a real-time data pipeline using MQTT, InfluxDB, and Grafana to monitor and visualize building performance [9].
- Analyze energy consumption trends using energy signiture.

1.3 State of the Art

This section reviews existing technologies and methodologies relevant to ICT in building design, focusing on simulation, optimization, and smart control systems.

1.3.1 Building Energy Simulation

Building energy simulation tools play a key role in analyzing energy performance during the design phase. EnergyPlus and DesignBuilder are two widely used simulation platforms capable of modeling complex building systems including heating, cooling, ventilation, lighting, and electrical components [3]. These tools enable parametric studies and energy performance comparisons across multiple design alternatives. In addition to steady-state simulations, EnergyPlus supports dynamic simulations that account for hourly weather data, occupant behavior, and time-varying control strategies.

Recent developments have also focused on integrating these simulations into automated workflows. Tools like BESOS (Building Energy Simulation Optimization Software) offer Python-based APIs to facilitate simulation, data extraction, and performance analysis [11]. Furthermore, simulations are increasingly being used not just for initial design, but also for operational decision support through integration with real-time control systems via standards like Functional Mock-up Units (FMUs) [10].

1.3.2 Surrogate Models and Optimization

While building simulations provide accurate performance metrics, they are often time-consuming when used in large-scale design explorations or optimization tasks. To address this, surrogate modeling has emerged as a promising solution. Surrogate models approximate the behavior of simulation tools using machine learning techniques trained on a limited number of simulation runs. Popular methods include Gaussian Processes (GP) [4], Artificial Neural Networks (ANNs), Multi-layer Perceptrons (MLPs) [5], and Deep Neural Networks (DNNs).

Recent studies have also explored hybrid and ensemble learning approaches to improve the robustness of surrogate predictions [13]. These models are often embedded within multi-objective optimization frameworks such as NSGA-II (Non-dominated Sorting Genetic Algorithm II), which can efficiently explore trade-offs between competing design goals such as energy efficiency, thermal comfort, and cost [6].

Additionally, techniques like Latin Hypercube Sampling (LHS) are employed to generate well-distributed design points in the parameter space, ensuring the training data is representative [12]. Open-source platforms such as Scikit-learn, TensorFlow, and PyGMO have made it easier to implement and customize these algorithms for building energy design tasks.

1.3.3 Smart Building Control and Monitoring

Modern building management systems are evolving into smart, adaptive platforms that can monitor, predict, and optimize performance in real-time. This transition is supported by the use of Functional Mock-up Units (FMUs), which allow for co-simulation of energy models with real-time data feeds and control logic [7, 10]. FMUs enable integration with digital twin frameworks and reinforcement learning environments such as OpenAI Gym, enhancing the ability to test advanced control strategies.

Communication between sensors, controllers, and databases is typically handled using lightweight messaging protocols like MQTT, which support scalable and flexible data transmission. Time-series databases like InfluxDB store operational data efficiently, and visualization platforms like Grafana offer interactive dashboards for real-time monitoring [9].

For predictive control, machine learning models such as Long Short-Term Memory (LSTM), XGBoost, and Prophet are increasingly used to forecast energy consumption and detect anomalies [8]. Model Predictive Control (MPC) strategies are being adopted for HVAC systems to anticipate future states and reduce energy consumption while maintaining thermal comfort [14]. Recent research also investigates reinforcement learning agents for autonomous decision-making in building operations [15].

2 Methodology

This section details the systematic approach adopted in this project, outlining the overall workflow, system setup, simulation inputs, and the specific steps involved in both the early-design and simulation stages.

2.1 Workflowd

This section outlines the overall workflow and pipeline developed for building design optimization and simulation, as illustrated in Figure 1. The process is divided into two main parts: the Early-Design Stage and the Simulation Step. The Early-Design Stage integrates various computational tools and techniques, starting from defining building parameters and objectives, executing EnergyPlus simulations, processing data, training surrogate models, and finally performing multi-objective optimization to identify optimal designs. The Simulation Step focuses on the operational aspect, using real-time data for energy signature analysis. The entire setup is managed within a containerized environment to ensure reproducibility and consistency.

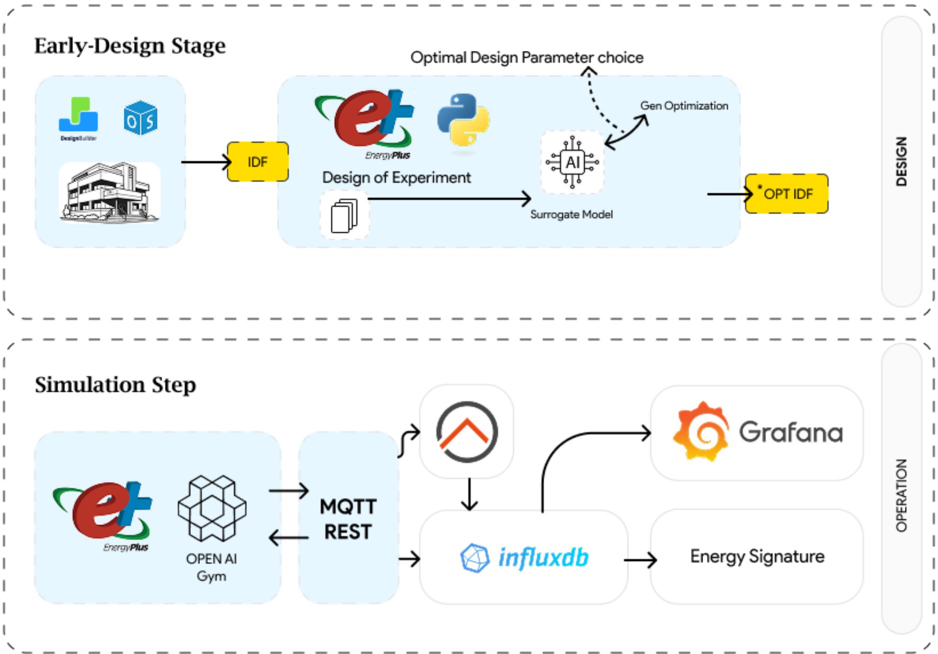


Figure 1: Overview of the Building Design Optimization and Simulation Workflow

2.2 System Setup

2.2.1 Docker Environment and Services

The entire platform for this project is containerized using Docker and managed through 'docker-compose'. This approach ensures that the computational environment is consistent and reproducible across different machines. The following key services, many of which are depicted in Figure 1, were launched as part of the setup:

- **EnergyPlus** – This service facilitates the execution of EnergyPlus simulations within a controlled environment, essential for running the building energy models. It is a core component in both the design and simulation stages.
- **Mosquitto** – An MQTT message broker used for lightweight and efficient communication between different components of the system, such as passing simulation requests and receiving results, particularly evident in the Simulation Step.

- **InfluxDB** – A purpose-built time-series database used to store real-time simulation data and performance metrics, enabling efficient data retrieval and analysis. This database is key for storing operational data.

- **Grafana** – A powerful open-source platform integrated for visual monitoring and creating interactive dashboards, allowing for real-time visualization of simulation progress and results, primarily used for data presentation in the Simulation Step.

2.2.2 Images and Container Management

To establish the operational environment, two specific Docker images were pulled and utilized:

- `lorenzobottaccioli/building_design`: This image likely contains the necessary tools and libraries for setting up and running the design optimization workflows.

- `lorenzobottaccioli/virtual_building_slim`: This image probably provides a streamlined EnergyPlus environment or related virtual building components necessary for running the simulations.

These images ensure that all dependencies and software versions are consistently managed within their respective containers.

2.3 Simulation Inputs

This subsection details the primary data files and initial model configurations used to drive the energy simulations.

2.3.1 Reading IDF and EPW Files

The foundational building model for this study was the ‘Office.idf‘ file which was provided by Professor. Prior to use, this model was upgraded from EnergyPlus version 9.4 to 9.6 using the EnergyPlus IDFVersionUpdater tool to ensure compatibility with the simulation environment. The weather data crucial for the energy simulations was provided by the ‘Larnaca.epw‘ file, loaded using the ‘epw‘ Python library. Both the IDF and EPW files were parsed and prepared for simulation using the ‘eppy‘ [10] and ‘besos‘ [11] Python libraries.

2.3.2 Base Building Model (‘Office.idf‘)

The starting point for this study’s simulations was a pre-defined EnergyPlus Input Data File (IDF) named ‘Office.idf‘. This file, provided for the project, describes the initial geometry and fundamental characteristics of the office building model. Figure 2 visually represents this foundational model, which served as the basis before any parametric modifications or optimizations were applied.

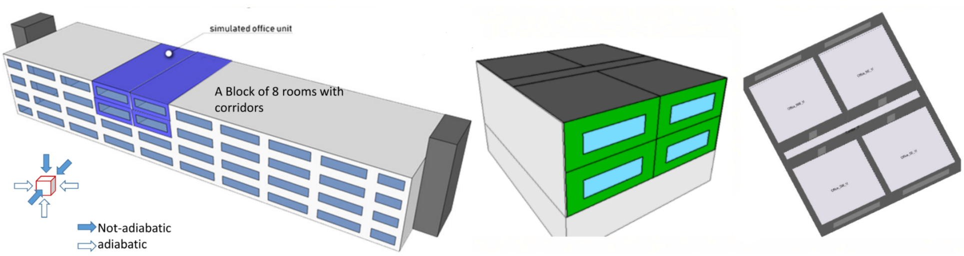


Figure 2: A block of 8 rooms with corridors

2.3.3 Parameter and Objective Definition

Based on the course instructions, the original ‘Office.idf‘ building model was modified to allow for parametric variations. The key design parameters adjusted included:

- **Windows**: The window construction type was updated to triple-glazed (Tpl) within the ‘FenestrationSurface:Detailed‘ object, offering different insulation and solar heat gain properties.

- **Shading**: The control mechanism for shading devices was changed from a continuous '24/7' operation to 'Summer-Ventilation'-based control, allowing for adaptive shading strategies.
- **Zone Infiltration**: The infiltration rate was defined in 'AirChanges/Hour' mode, with a value set to 0.2 ACH.
- **Lights**: The lighting power density was reduced from an initial 7.5 W/m² to 4 W/m², aiming for lower internal heat gains and electricity consumption.
- **Ventilation**: The natural ventilation strategy was updated to an ACH mode, with a rate of 4 ACH.
- **HVAC**: The sensible heat recovery efficiency in the HVAC system was set to 0.9, an increase from the previous value of 0.7, to improve energy recovery.

The objectives selected for optimization, which represent the energy performance metrics to be minimized, included:

- `Electricity:Facility` (Total electricity consumption of the building)
- `DistrictHeating:Facility` (Total district heating consumption)
- `DistrictCooling:Facility` (Total district cooling consumption)

2.4 Early-Design stage Methodology

The core of this project's design phase involves an optimization process to identify the most energy-efficient configuration for our office building. This process integrates building energy simulations with data analysis and preparation for machine learning.

2.4.1 Defining the Problem and Sampling the Design Space

The first step was to clearly define what aspects of the building we could change (parameters) and what we wanted to improve (objectives). Our parameters included building features like wall and roof insulation thickness, window types, ventilation rates (Air Changes per Hour), lighting power density (Watts per Zone Floor Area), and shading setpoints (both temperature and radiation-based). The objectives were to minimize total electricity, district heating, and district cooling energy consumption.

To explore the many possible combinations of these parameters, we used **Latin Hypercube Sampling (LHS)** [12]. This method ensures that our chosen samples are spread out well across the entire range of possible parameter values. For this project, we generated 30 distinct design configurations.

2.4.2 Executing EnergyPlus Simulations

Each of the 30 unique input samples was then simulated using EnergyPlus [3]. The `EvaluatorEP` class from the `besos` library [11] managed this process, running each simulation and extracting the desired energy performance metrics. The weather conditions for Larnaca, Cyprus, were applied using the `Larnaca.epw` file. Running these simulations is computationally intensive; for our samples, the total simulation time was approximately 32 minutes and 8 seconds.

2.4.3 Data Storage and Initial Analysis

After all simulations were complete, the results were stored in a Python `.pkl` (pickle) file. This allows us to quickly load the data later without needing to rerun the time-consuming EnergyPlus simulations.

To understand the range and distribution of our simulation outputs, we performed an initial statistical analysis in Pandas. This provided key statistics in table 1 for all input parameters and energy outputs .

Table 1: Descriptive Statistics of Simulation Outputs

	Wall Ins.	Roof Ins.	Lights W/A	Ventilation ACH	Temp Setpoint shading	Rad Setpoint shading	Electricity:Facility	DistrictHeating:Facility	DistrictCooling:Facility
count	30.000	30.000	30.000	30.000	30.000	30.000	3.000×10^{1}	3.000×10^{1}	3.000×10^{1}
mean	0.301	0.451	14.009	3.009	23.999	189.910	1.313×10^{11}	3.126×10^{5}	1.558×10^{11}
std	0.117	0.206	3.521	1.730	3.517	65.153	1.098×10^{10}	1.712×10^{6}	1.367×10^{10}
min	0.113	0.111	8.354	0.191	18.149	82.610	1.141×10^{11}	0.000	1.347×10^{11}
25%	0.200	0.272	11.224	1.601	21.165	135.241	1.223×10^{11}	0.000	1.439×10^{11}
50%	0.299	0.454	13.877	3.002	23.972	191.861	1.324×10^{11}	0.000	1.548×10^{11}
75%	0.396	0.622	17.016	4.390	26.788	244.731	1.399×10^{11}	0.000	1.658×10^{11}
max	0.489	0.797	19.676	5.806	29.700	299.124	1.500×10^{11}	9.377×10^{6}	1.837×10^{11}

2.4.4 Correlation Analysis

To quantitatively assess the relationships observed in the initial visualizations, the Pearson correlation coefficient (r) was calculated between all input parameters and output objectives. The formula for Pearson correlation is given by:

$$r = \frac{N \sum XY - (\sum X \sum Y)}{\sqrt{[N \sum x^2 - (\sum x)^2][N \sum y^2 - (\sum y)^2]}} \tag{1}$$

where N is the number of samples, X is the vector of observations for variable 1, and Y is the vector of observations for variable 2. The resulting correlation coefficients were used to create a heatmap, which is presented in the Results section.

2.4.5 Data Preprocessing for Surrogate Modeling

Before training our machine learning surrogate models, the data required further preparation to be suitable:

- **Categorical Feature Encoding:** The 'Windows types' parameter, which was originally descriptive, was converted into numerical representations (e.g., 1, 2, 3).
- **Train-Test Split:** The entire dataset was divided into a training set (80% of the data) and a testing set (20% of the data).
- **Normalization:** All input and output variables were normalized to have a mean of 0 and a standard deviation of 1.

2.4.6 Surrogate Modeling

To speed up the optimization process, we replaced computationally expensive EnergyPlus simulations with faster, approximate "surrogate models" [5]. These models learn how building design affects energy performance. We explored three types of surrogate models:

- **Gaussian Process Regressor (GP):** Implemented using scikit-learn [4]. GPs provide predictions and estimate uncertainty. The model used a Constant and RBF Kernel, with hyperparameters tuned by 5-fold cross-validation.
- **Neural Network (MLPRegressor):** A feedforward neural network from scikit-learn [5]. It had three hidden layers (100 neurons each) and used the ‘relu‘ activation and ‘adam‘ optimizer. Hyperparameters were tuned using 5-fold cross-validation.
- **TensorFlow Deep Neural Network (DNN):** A more complex neural network built with TensorFlow and Keras. It included multiple dense layers with ReLU activation and Batch Normalization for stability. The model was trained for 100 epochs using the ‘adam‘ optimizer and mean squared error (MSE) loss.

Once the best-performing surrogate model was trained (the TensorFlow DNN), it was used for design optimization to rapidly explore many design possibilities, which is unfeasible with direct EnergyPlus simulations.

2.4.7 Design Space Exploration and Multi-objective Optimization

The selected surrogate model was integrated into a ‘EvaluatorGeneric‘ object from the ‘besos‘ library [11] to mimic the EnergyPlus simulator for rapid evaluations. We generated 500 new design candidates using Latin Hypercube Sampling [12] and evaluated each with the surrogate model to predict energy performance across all objectives. Multi-objective optimization was then performed using the NSGA-II algorithm [6]. This evolutionary algorithm is well-suited for conflicting objectives like minimizing electricity, heating, and cooling simultaneously. The optimization ran for 5000 evaluations with a population size of 10000 to thoroughly search for optimal design trade-offs.

2.4.8 Optimal Design Selection and Model Update

From the NSGA-II results, Pareto-optimal solutions were extracted, representing designs where no objective can be improved without worsening another. These solutions form the Pareto front, illustrating the most efficient design trade-offs. A single "optimal design" was then chosen from this set, primarily based on minimizing total energy demand (sum of electricity, heating, and cooling), or by finding the design closest to an "ideal" point for balanced performance. Finally, the parameters of this chosen optimal design were implemented back into the original EnergyPlus IDF, and the modified model was saved as **optimal** version for future analysis.

2.5 Simulation Stage Methodology

This section briefly describes the dynamic simulation phase, covering the preparation of the optimal building model, its export to FMU, simulation for data generation, and the pipeline for data visualization and energy signature analysis.

2.5.1 Optimal IDF Preparation

To prepare the optimal building model for dynamic simulation, we modified the EnergyPlus IDF by adding these key objects:

- `EnergyManagementSystem:Sensor`: To get data like temperatures and energy use from simulation.
- `EnergyManagementSystem:Actuator`: To control parts of the building model, like ventilation.
- `EnergyManagementSystem:Program`: To write simple control rules, such as changing ventilation based on temperature.
- `ExternalInterface:FunctionalMockupUnitExport:From:Variable`: To choose which sensor data and control points are available when the model is exported as an FMU.

These changes allowed the model to be simulated dynamically and connected to other tools.

2.5.2 FMU Export

The IDF, containing the optimal design and interface elements, was converted into a Functional Mock-up Unit (FMU version 2.0) using the `EnergyPlusToFMU-v3.1.0` tool. This enabled its use in co-simulation environments.

2.5.3 Simulation Data Generation (CSV)

The exported FMU was simulated for a one-year period (1994, hourly steps) using the `fmi_gym` Python library. During these simulations, a Python script continuously extracted observations from the running FMU. These observations included:

- Internal zone temperatures (e.g., `T_Block1_OfficeXSWX1f`, `T_Block2_CorridorX2f`).
- Whole-building energy consumptions (`Electricity`, `DistrictHeating`, `DistrictCooling`).
- Environmental data (e.g., `Wind Speed`, `DHI`, `DNI`, `GHI` for solar radiation).

This key operational data was recorded and saved into a CSV file for further processing and analysis.

2.5.4 Data Streaming to Grafana (via MQTT)

Data from the CSV file was streamed to Grafana for visualization. A Python script published each data point as a JSON payload to an MQTT topic (`virtual_building_send`). This data was then ingested by InfluxDB, which served as the data source for Grafana dashboards, allowing for the monitoring of the building's simulated operational performance. Results are available in section 4.

2.5.5 Energy Signature Analysis

Energy signature modeling was performed to assess the building's electricity consumption ($P(t)$) in relation to the indoor-outdoor temperature difference ($\Delta T(t)$). Using one year of simulated hourly data, two regression models were developed:

1. A log-linear model: $\ln(P(t)) = \alpha + \beta \Delta T(t) + \varepsilon'(t)$.
2. A direct linear model: $P(t) = \gamma + \delta \Delta T(t) + \varepsilon''(t)$.

These models helped quantify the base load and temperature sensitivity of the building's electricity use.

3 Results: Building Design Optimization

This section presents the key findings from the building design optimization phase, covering the initial simulation outputs, the accuracy of the trained surrogate models, and the final outcomes of the multi objective optimization process.

3.1 Simulation Outputs

The initial EnergyPlus simulations of the sampled building designs generated raw data for our analysis, representing the energy performance across diverse design configurations. These outputs provided an overview of energy consumption, with descriptive statistics (such as range, mean, and distribution of electricity, heating, and cooling demands) previously detailed in Table 1 in the Methodology section.

3.1.1 Output Visualization

To visually analyze the initial simulation outputs and understand the impact of varying design parameters, several plots were generated and are presented here:

- **Bar Charts:**This bar charts visually represents the individual parameter and energy objective values for 30 designs. It highlights the diverse distribution of inputs, and the significant range in outputs such as electricity and cooling demands. Notably, District only heating appears very low or zero across most designs, except when Wall and Roof insulation are both very low.

Figure 3: Bar Charts of Initial Simulation Inputs and Outputs

- **Pair Plots:**This figure presents scatter plots of inputs versus outputs, visually revealing relationships between each design parameter and energy objective. These plots generally show a scattered distribution, indicating non-linear or weak linear correlations between many input-output pairs, suggesting complex interactions. Notably, some plots (likely involving district heating, as observed previously) show points clustered at very low values, confirming its minimal contribution across most designs. This visualization was essential for identifying initial trends and the overall complexity of the design space.

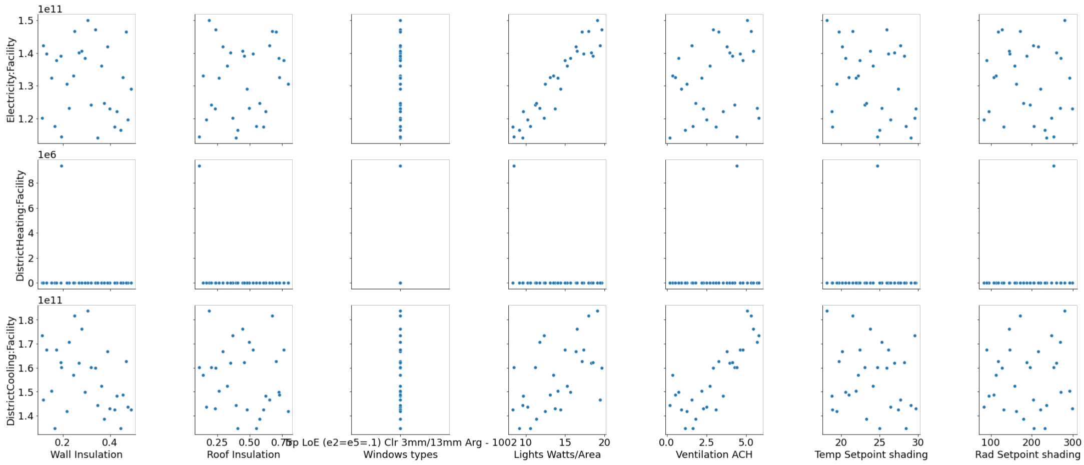


Figure 4: Pair Plots of Initial Simulation Inputs vs. Outputs

- **Correlation Heatmap:** This figure displays linear relationships between inputs and outputs. It highlights a very strong positive correlation (0.94) between "Lights Watts/Area" and electricity consumption, and a strong positive correlation (0.82) between "Ventilation ACH" and cooling demand. Heating demand generally shows weaker correlations.

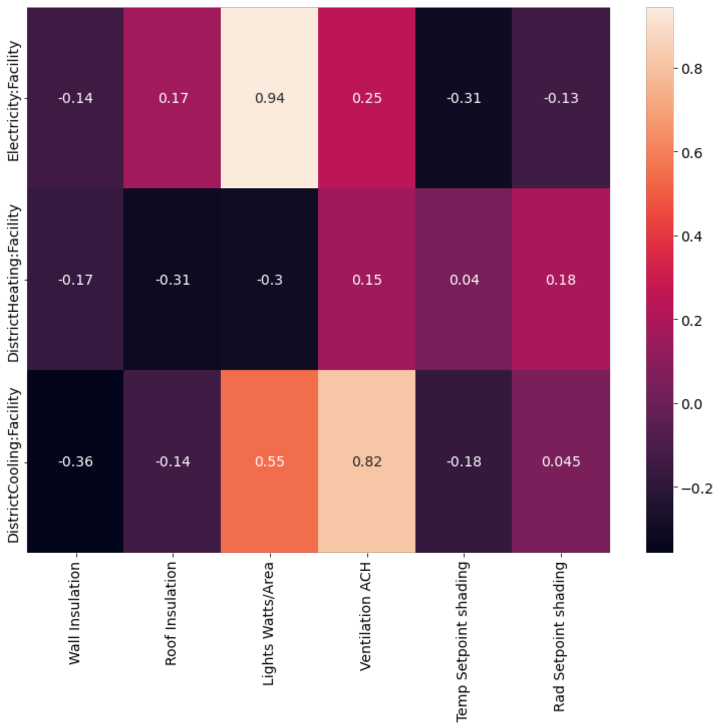


Figure 5: Correlation Heatmap of Initial Simulation Inputs and Outputs

3.2 Surrogate Model Accuracy

The accuracy of trained surrogate models is crucial for replacing time-consuming EnergyPlus simulations. We evaluated each model's performance on a 20% test dataset.

3.2.1 Evaluation Metrics (Normalized RMSE)

The primary metric was Normalized Root Mean Squared Error (RMSE), expressed as a percentage of the mean of actual output values. Lower values indicate higher accuracy.

- **Gaussian Process (GP):**
  - Electricity:Facility: 4.83%
  - DistrictHeating:Facility: inf (Likely due to near-zero mean values in test set)
  - DistrictCooling:Facility: 4.36%
- **Neural Network (MLPRegressor):**
  - Electricity:Facility: 5.88×10^{10}% (Indicates severe misprediction)
  - DistrictHeating:Facility: inf
  - DistrictCooling:Facility: 8.95×10^{10}%
- **TensorFlow Deep Neural Network (DNN):**
  - Electricity:Facility: 3.53%
  - DistrictHeating:Facility: inf
  - DistrictCooling:Facility: 3.94%

The "inf" values for DistrictHeating across all models consistently suggest near-zero actual heating demand in the test set, leading to undefined or extremely large percentage errors upon normalization.

3.2.2 Model Comparison

The **TensorFlow Deep Neural Network (DNN)** demonstrated the highest accuracy for Electricity (3.53%) and DistrictCooling (3.94%), exhibiting the lowest percentage errors among the models. While the Gaussian Process model also performed well, the scikit-learn MLPRegressor showed significantly unacceptable errors. Consequently, the TensorFlow DNN was selected as the primary surrogate for optimization.

3.3 Optimization Outcomes

The multi-objective optimization performed using the selected TensorFlow DNN surrogate model yielded a set of Pareto-optimal designs that represent the best trade-offs in energy performance.

3.3.1 Surrogate Evaluation Results and Exploration

After the selected surrogate model was evaluated with 500 new samples, the results provided a dense mapping of the design space. A parallel coordinates plot was generated to visualize these results from the surrogate model's predictions. This plot helps to understand how different combinations of input parameters lead to various predicted energy consumption levels.

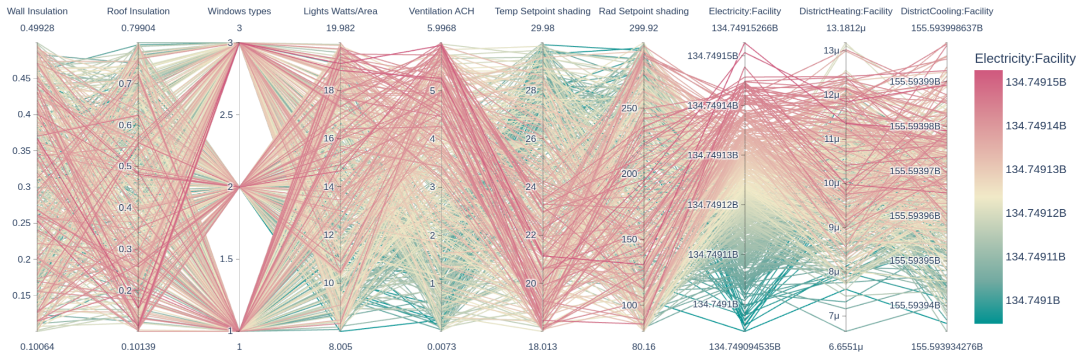


Figure 6: Parallel Coordinates Plot of Surrogate Evaluation Results

This plot visually represents the predicted energy performance across 500 unique designs as calculated by the surrogate model. Each line connects the parameter values and resulting energy demands for one design, highlighting trends and correlations in the vast design space.

3.3.2 Pareto Front Visualization

The outcome of the NSGA-II optimization algorithm is a set of Pareto-optimal solutions, which are designs that offer the best possible trade-offs between the conflicting objectives (minimizing electricity, heating, and cooling). These solutions collectively form the Pareto front. A 3D plot is used to visualize these Pareto-optimal designs within the objective space.

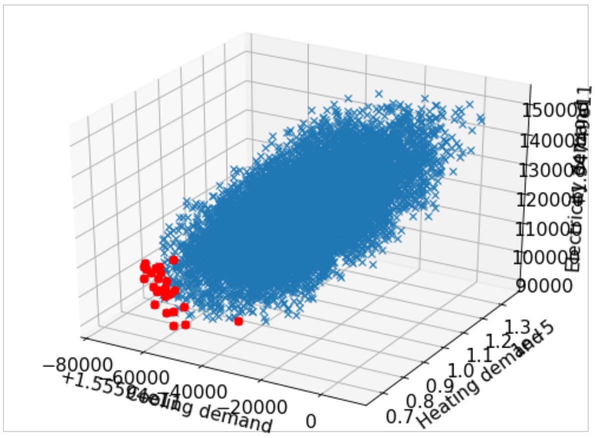


Figure 7: 3D Plot of Optimization Results and Pareto Front

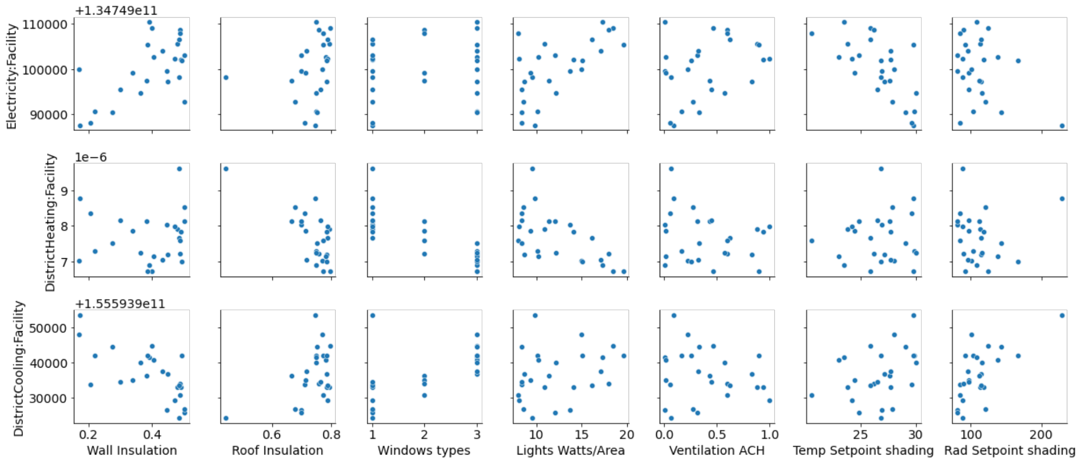
This 3D plot displays all design configurations evaluated during optimization (blue crosses) and highlights the Pareto-optimal designs (red circles). It visually demonstrates the trade-offs achievable between cooling, heating, and electricity demands.

3.3.3 Analysis of Optimal Designs

Further analysis of the Pareto-optimal solutions was performed to understand their specific characteristics and the relationships between their parameters and performance within the optimized set.

- **Pair Plots of Pareto-Optimal Designs:** Pair plots focusing exclusively on the Pareto-optimal designs were generated to illustrate the relationships between input parameters and the corresponding energy objectives for these highly efficient solutions. These plots help identify common features or ranges of parameters that lead to optimal performance.

Figure 8: Pair Plots of Pareto-Optimal Designs



These plots detail the relationships between parameters and energy objectives specifically for the highly efficient Pareto-optimal designs, indicating common features among the best-performing solutions.

- **Correlation Heatmap of Optimal Designs:** A correlation heatmap was created using only the Pareto-optimal solutions to identify which design parameters are most influential in shaping the optimal trade-offs. This heatmap highlights how parameters correlate with each other and with the energy objectives specifically within the Pareto front.

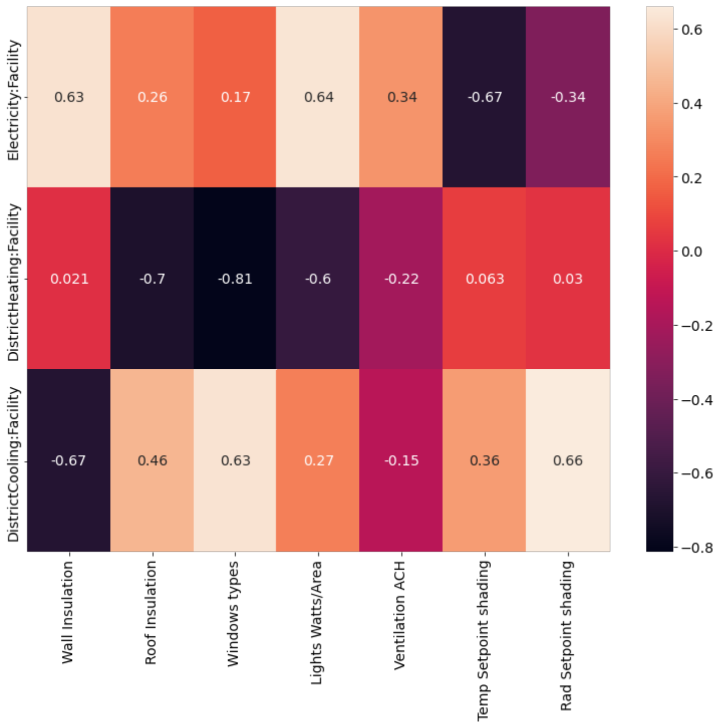


Figure 9: Correlation Heatmap for Pareto-Optimal Solutions

This heatmap illustrates the correlation coefficients among parameters and energy objectives specifically within the Pareto-optimal set, revealing key parameter sensitivities for optimized designs.

3.3.4 Selected Optimal Design

From the set of Pareto-optimal solutions, a single optimal design was chosen as the final recommended building configuration. This selection was based on minimizing the total energy demand (sum of Electricity, District Heating, and District Cooling), or finding the design closest to an ideal minimum across all objectives in the objective space. The specific parameters for this chosen optimal design are summarized below:

Table 2: Selected Optimal Design Parameters

Parameter	**Optimal Value**
Wall Insulation	0.499768
Roof Insulation	0.679734
Windows types	1 (Triple Glazing)
Lights Watts/Area	8.601171
Ventilation ACH	0.272749
Temp Setpoint shading	27.840928
Rad Setpoint shading	120.402001

This table presents the specific values for each design parameter of the selected optimal building configuration. This design represents the best energy performance trade-off identified by the optimization process.

The total energy demand for this optimal design was:

- Electricity:Facility: 1.347491×10^{11}
- DistrictHeating:Facility: 0.000009
- DistrictCooling:Facility: 1.555939×10^{11}
- Total Energy: 2.903430×10^{11}

This design demonstrates significantly reduced heating demand, indicating that the optimization successfully identified parameters leading to minimal heating energy use, likely due to a combination of high insulation and appropriate window/shading selections.

4 Results: Building Operational

This section presents the findings from the dynamic operational assessment of the optimized building model. The performance was monitored under simulated operational conditions using the FMU and data pipeline described in the Methodology (Section 2.5).

4.1 Visualization of Operational Performance

The operational performance of the building, including internal temperatures and energy consumption, was visualized using Grafana dashboards. Key observations from one year simulation (1994) are presented below.

Examples of the real-time operational data visualized in Grafana dashboards are shown in the figures below.

Figure 10: Grafana Dashboard: Internal Zone Temperatures

This figure shows internal air temperatures for multiple building zones throughout 1994. During the warmer period (mid-April to mid-October), temperatures were actively managed, with the control system frequently capping them at an upper limit of approximately 26°C. This strategy effectively prevented sustained high temperatures in these zones..

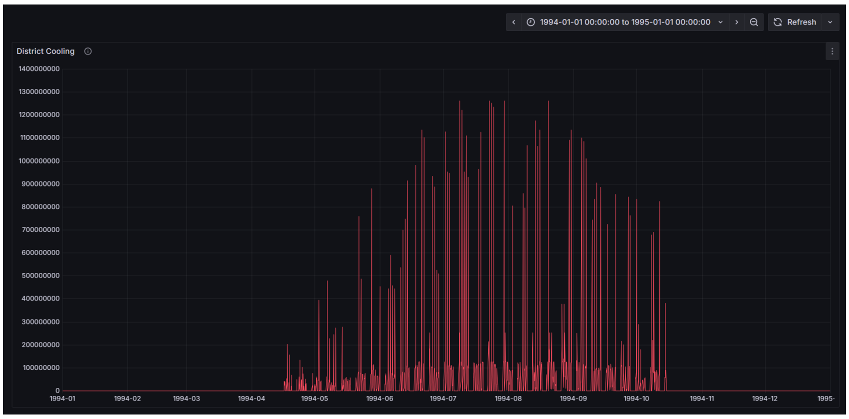


Figure 11: Grafana Dashboard: Building Energy Consumption (District Cooling)

This 1994 data reveals minimal district cooling usage from January through mid-April. Thereafter, consumption significantly increases, with frequent, high-magnitude intermittent bursts from mid-April (intensifying May through mid-October). This operational pattern suggests an on/off control strategy for the cooling system, correlating with the need to manage indoor temperatures (around the 26°C upper limit) during warmer months.

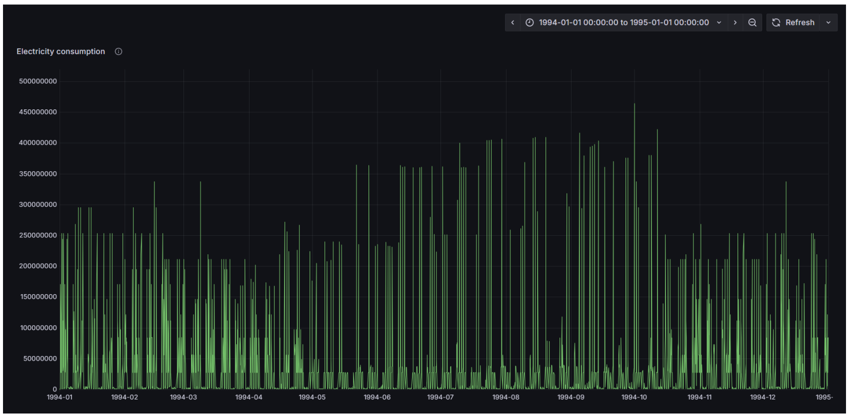


Figure 12: Grafana Dashboard: Building Energy Consumption (Electricity)

This plot illustrating the electricity consumption of the building over the simulation period, highlighting demand fluctuations and peak usage times.

4.1.1 Environmental Conditions Monitoring

Monitoring external environmental conditions is crucial as they directly influence the building's energy performance and thermal loads. Various environmental factors were streamed and visualized in Grafana:

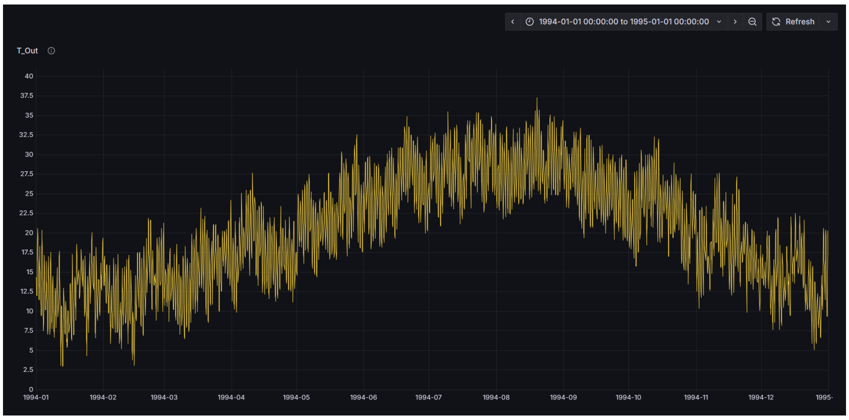


Figure 13: Grafana Dashboard: Outdoor Temperature

This 1994 outdoor temperature data shows a clear seasonal cycle, with values ranging from winter lows near 5°C to summer peaks approaching 35-38°C. This external temperature profile is a primary driver influencing the building's heating and cooling energy demands..

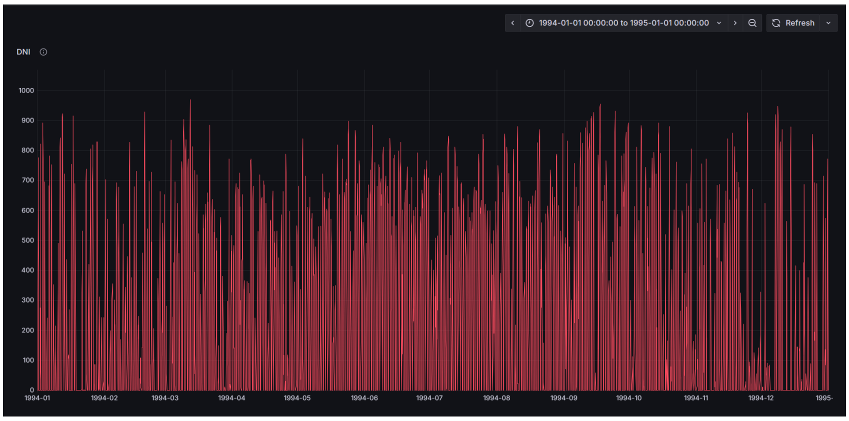


Figure 14: Grafana Dashboard: Direct Normal Irradiance (DNI)

This figure shows the Direct Normal Irradiance, indicating the amount of direct solar radiation incident on a surface, which is crucial for understanding solar heat gain and shading effectiveness.

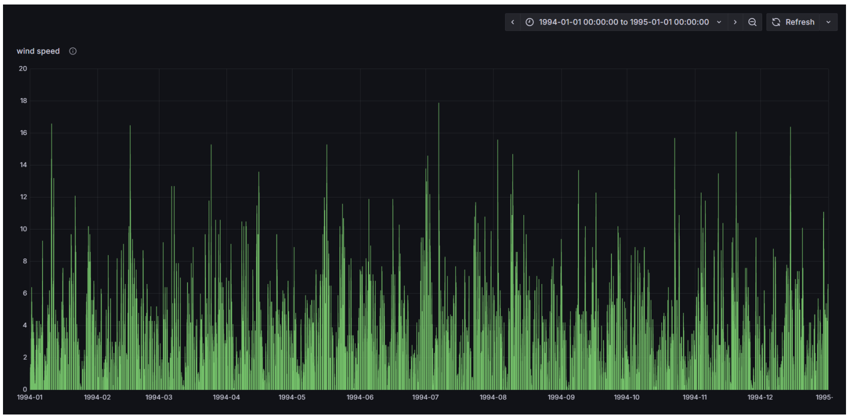


Figure 15: Grafana Dashboard: Wind Speed

A Grafana plot displaying the wind speed, an environmental factor influencing infiltration and heat transfer through the building envelope.

4.2 Energy Signature Modeling

Energy signature modeling assesses how building energy use responds to the outdoor-indoor temperature difference ($\Delta T = T_{\text{out}} - T_{\text{in}}$), defining base loads and temperature sensitivities using 1994 simulated data.

Weekly Energy Signature Weekly data aggregation typically smooths variability, revealing clearer trends useful for prediction. For instance, a log-linear model for weekly electricity consumption ($P_{\text{elec, weekly}}$) against the weekly average ΔT_{weekly} can be expressed as:

$$\ln(P_{\text{elec, weekly}}(t)) = \alpha + \beta \Delta T_{\text{weekly}}(t) + \varepsilon'(t)$$

Figure 16 displays such a relationship where the trend is more distinct.

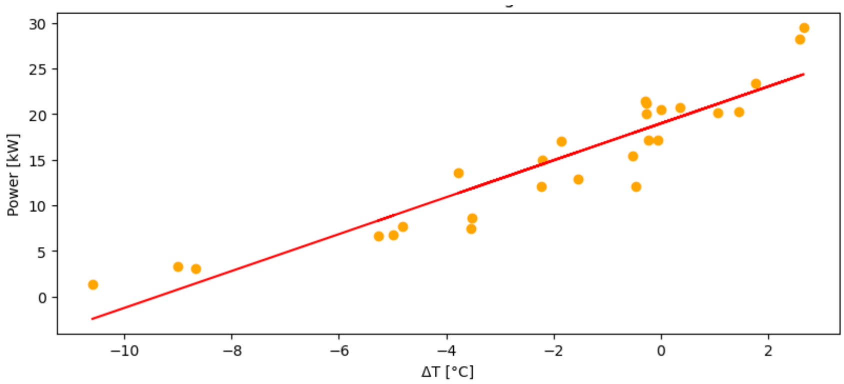


Figure 16: Weekly aggregated data: ΔT_{weekly} vs. $\ln(P_{\text{elec, weekly}})$ indicating a clearer linear trend.

Daily Energy Signature Daily data provides more detailed insight into specific end-uses like District Cooling ($P_{\text{cool, daily}}$ in kW), though often with more scatter. A linear model was fitted:

$$P_{\text{cool, daily}}(t) = \gamma + \delta \Delta T_{\text{daily}}(t) + \varepsilon''(t)$$

This yielded a base load (γ) of ≈ 19.0 kW and a sensitivity (δ) of ≈ 2.02 kW/°C, with an R-squared of 0.863. Figure 17 shows this, illustrating more variance.

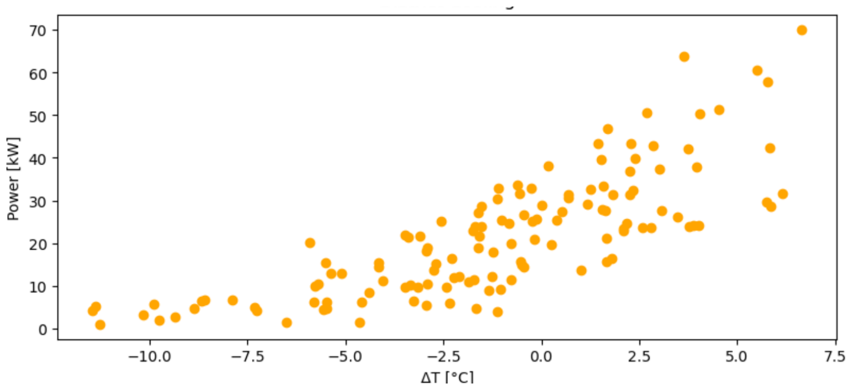


Figure 17: Daily aggregated data: ΔT_{daily} vs. District Cooling Power ($P_{\text{cool, daily}}$) with its linear fit.

In summary, while daily data offers granularity, weekly aggregation often yields a more stable and predictable energy signature.

5 Conclusion

This project successfully demonstrated an Information and Communication Technologies (ICT)-driven workflow for the design, optimization, and operational analysis of an energy-efficient office building. Through the systematic integration of advanced simulation tools, machine learning techniques, and real-time data handling, an optimized building design was achieved alongside a dynamic framework for its performance assessment, fulfilling all primary objectives.

The initial phase focused on design optimization. By employing EnergyPlus simulations and a highly accurate TensorFlow Deep Neural Network (DNN) surrogate model within a Non-dominated Sorting Genetic Algorithm II (NSGA-II) multi-objective optimization process, an optimal building configuration was identified. This configuration featured high levels of insulation, superior glazing, reduced lighting power density, and optimized ventilation and shading strategies. These enhancements resulted in a significant reduction in heating demand and substantially lowered overall energy consumption.

Subsequently, this optimized building model was converted into a Functional Mock-up Unit (FMU) for dynamic operational analysis. One year of simulated operation was executed, with key performance indicators streamed for real-time visualization. This enabled the monitoring of effective environmental control, such as maintaining stable indoor temperatures, and provided insights into HVAC operational patterns. Energy signature analysis performed on the simulated output further quantified the building's energy behavior by identifying key relationships between energy use and environmental conditions, including base loads and temperature sensitivities.

The successful integration of these methodologies—from parametric modeling and simulation-based optimization to surrogate modeling, FMU co-simulation, and containerized data analytics—highlights the significant capabilities of ICT in advancing sustainable building design. This project not only delivered an optimized building solution but also established a versatile and reproducible platform for future explorations in building performance, such as testing advanced control strategies or developing digital twin applications.

References

[1] International Energy Agency, "Global Status Report for Buildings and Construction," 2022. https://www.iea.org/reports/global-status-report-for-buildings-and-construction-2022

[2] G. Chiesa and L. Bottaccioli, "ICT in Building Design," course materials, Politecnico di Torino, 2020.

[3] D. B. Crawley et al., "EnergyPlus: Creating a New-Generation Building Energy Simulation Program," *Energy and Buildings*, vol. 33, no. 4, 2001.

[4] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[5] K. Bamdad et al., "Surrogate modeling for building energy simulation using Artificial Neural Networks," *Energy Procedia*, 2017.

[6] K. Deb et al., "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, 2002.

[7] LBNL, "EnergyPlusToFMU Tool," https://simulationresearch.lbl.gov/fmu/EnergyPlus/export

[8] F. Alshehri et al., "Forecasting Building Energy Consumption Using Machine Learning," *Energies*, 2022.

[9] L. Bottaccioli et al., "Modular virtual testbed for ICT-based building control solutions," *Automation in Construction*, vol. 110, 2020. https://doi.org/10.1016/j.autcon.2019.103002

[10] T. Nouidui, M. Wetter, and M. Zuo, "Functional Mock-up Units for co-simulation using EnergyPlus," *IBPSA Building Simulation Conference*, 2014.

[11] A. Rysanek et al., "BESOS: A Python platform for automated building performance optimisation," 2016. https://besos.readthedocs.io

[12] M. D. McKay, R. J. Beckman, and W. J. Conover, "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, vol. 21, no. 2, 1979.

[13] S. Wang and Z. Ma, "Supervised learning-based surrogate models for building performance simulation: A review," *Applied Energy*, vol. 221, pp. 383–398, 2018. https://doi.org/10.1016/j.apenergy.2018.03.107

[14] M. Killian and M. Kozek, "Ten questions concerning model predictive control for energy efficient buildings," *Building and Environment*, vol. 105, pp. 403–412, 2016. https://doi.org/10.1016/j.buildenv.2016.05.034

[15] T. Wei, Y. Wang, and Q. Zhu, "Deep reinforcement learning for building HVAC control," in *Proceedings of the 54th Annual Design Automation Conference (DAC)*, ACM, 2017. https://doi.org/10.1145/3061639.3062224

[16] A. Hopkins and M. Bell, "Fault Detection in HVAC Systems via Energy Signature Models," *Energy and Buildings*, vol. 92, pp. 141–152, 2015. https://doi.org/10.1016/j.enbuild.2014.11.082

[17] K. Kawamoto, S. Murakami, and F. Favoino, "Energy Signature Analysis for Building Monitoring," *ASHRAE Journal*, vol. 53, no. 10, pp. 12–22, 2011.