

LARGE SCALE POST-SIMULATION DATA PROCESSING AND VISUALIZATION FOR BUILDING ENERGY ANALYSIS

Junru Shen, Jeanie Chan, and Andrea Frisque
Stantec Consulting, Vancouver, BC

ABSTRACT

In the pursuit of integrated design and with the advent of advanced tools, it has become widespread in the building simulation community to generate big data sets that provide extensive information about potential design options. Design options are analyzed using parametric simulations to evaluate pathways to achieve project targets. Current parametric tools can batch process building energy simulations and compile the results into a single result file. However, they are not featured to incorporate customized post-simulation data-processing or data visualization.

This paper presents a method of using scripting tools to perform post-simulation data processing on large-scale simulations sets that are run through EnergyPlus engine. The powerful combination of parametric simulation and post-simulation data processing pushes the limitations of simulation tools and gives practitioners the flexibility to customize their simulation results and data visualization based on specific project needs. Scripting tools can significantly reduce the time needed for post-simulation calculation on large scale simulation sets while reducing the risk of errors.

INTRODUCTION

With the advent of advanced tools and with the aim of integrated design, it has become prevalent in the building performance simulation community to generate big data sets to provide sufficient data and recommendations to inform decision makers of the best design measures or combination of measures to implement.

Design options are analyzed using parametric simulations to evaluate the pathways to achieve a project's targets. Current parametric tools for energy simulation can process a group of simulations, extract the relevant information from the individual simulation results, and assemble them into a single global results file. However, users do not have the option to add customized post-simulation data-processing before

results are collected in the global result file or plot the data in graphs afterwards.

High level scripting tools have supported building energy analysis in various ways. For example, the Openstudio software development kit provides scripting tools called Measures, which can be used to apply ECMs to EnergyPlus models as well as create reports, visualizations, and custom workflows (Roth et al. 2016). In Chiesa et al. 2019, the programming language Python is used to automate an EnergyPlus based study on the impact of a building's window-to-wall ratio on heating and cooling loads. Python scripting is used to control the whole process; modifying input data, collecting simulation results, and analyzing output data including graph elaborations. Roth et al. 2018 presents four scripting frameworks that are actively being used to automate different modeling tasks and workflows with EnergyPlus: Eppy, Modelkit, OpenStudio Measures, and CBECC. Python and Ruby based scripts are used to perform model development, energy conservation measure (ECM) evaluations, large scale analysis, parametric analysis, and model translations. Finally, Millet et al 2013, use Python to automate various workflows typical to the building performance modeling and simulation process. The paper shows four short examples of how scripting is used for simulation input file templating, data, exchange and interoperability, performance curve regression, and time-series output data postprocessing.

Scripting tools help automate tedious tasks to increase modeler productivity and reduce error; however, it has not been widely used to perform complex data analysis for a large-scale set of simulation results. In this paper, we present an efficient method to perform post-simulation data processing for parametric simulation results from EnergyPlus. Specifically, utilizing the Python scripting language, we have automated complex calculations and data visualization for the results of multiple large-scale building simulations.

METHODOLOGY

The post-simulation data processing involves three steps: 1) simulation result extraction, 2) data processing and 3) data visualization. Details of each individual steps are explained in the following sections.

Step 1: Result Extraction from EnergyPlus

The fundamental step for post simulation data processing is to get the data needed for the calculation.

EnergyPlus (U.S. Department of Energy 2018) is a whole building energy simulation program that provides great flexibility in the output variables. Due to the large number of options for energy systems, there is no universal output variable list that can fit any type of system. To make sure that the output variables are applicable to the specified energy system, EnergyPlus requires at least one successful simulation run to generate the applicable libraries of output variables (i.e. rdd and mdd EnergyPlus output files) for the current EnergyPlus input data file (IDF). From these libraries, a user can copy the desired output variables into the IDF and run the simulation again. The specified output data will be stored in the standard output files from EnergyPlus: eplusout.eso and eplusout.mtr. The ReadVarsEso program, distributed with the EnergyPlus installation package, can be used to pull data from the eso file to a spreadsheet program. This workflow of getting a custom csv file from the simulation result is shown in Figure 1.

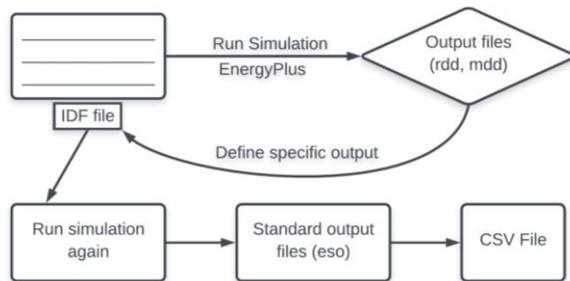


Figure 1. Workflow to Extract Simulation Result in CSV file

Step 2: Large Scale Post-Simulation Data Processing

In the Parametric Design Analysis (Chan, Irwin, et al. 2018) process where results need to be collected on a large number of simulation runs, jEPlus (Zhang et al. 2018) is used to manage simulation runs and collect results after completion. jEPlus is capable of configuring the time interval between each result extraction for each simulation. It is also capable of summarizing building level data for all simulations in a central spreadsheet. Figure 2 shows an example of jEPlus output showing all combined results for parametric simulations. Figure 3 shows an example of

jEPlus output of hourly data in a csv file for one simulation. jEPlus provides a solid platform for compiling the simulation data and job list for a parametric study, however it does not provide the opportunity to do any custom calculations before the results are collected into the summary spreadsheet.

| # | Job_ID | c0: Gas:Facility [J](RunPeriod) | c1: Electricity:Facility [J](RunPeriod) | c2: Humidifier:Electricity [J](RunPeriod) |
|---|--------------------------------------------|---------------------------------|-----------------------------------------|-------------------------------------------|
| 0 | EP_G-T_0-W_0-P1_0-P2_0-P3_0-P4_0-P5_0-P6_1 | 4.08E+13 | 2.91E+13 | 1.33E+13 |
| 1 | EP_G-T_0-W_0-P1_0-P2_0-P3_0-P4_1-P5_1-P6_0 | 3.08E+13 | 2.42E+13 | 6.69E+12 |
| 2 | EP_G-T_0-W_0-P1_0-P2_0-P3_1-P4_0-P5_0-P6_0 | 4.06E+13 | 2.91E+13 | 1.33E+13 |
| 3 | EP_G-T_0-W_0-P1_0-P2_0-P3_1-P4_0-P5_0-P6_1 | 4.06E+13 | 2.91E+13 | 1.33E+13 |
| 4 | EP_G-T_0-W_0-P1_1-P2_0-P3_0-P4_0-P5_0-P6_1 | 4.07E+13 | 2.91E+13 | 1.33E+13 |
| 5 | EP_G-T_0-W_0-P1_1-P2_0-P3_0-P4_1-P5_1-P6_0 | 3.07E+13 | 2.42E+13 | 6.69E+12 |
| 6 | EP_G-T_0-W_0-P1_1-P2_0-P3_1-P4_1-P5_0-P6_1 | 4.10E+13 | 2.79E+13 | 1.32E+13 |
| 7 | EP_G-T_0-W_0-P1_1-P2_0-P3_1-P4_1-P5_1-P6_1 | 3.05E+13 | 2.42E+13 | 6.69E+12 |

Figure 2. Example of jEPlus Output – All Combined Results

| Date/Time | Gas:Facility [J](Hourly) | Electricity:Facility [J](Hourly) | Humidifier:Electricity [J](Hourly) |
|----------------|--------------------------|----------------------------------|------------------------------------|
| 01/01 01:00:00 | 7617837243 | 3991057106 | 2698123437 |
| 01/01 02:00:00 | 9127646316 | 4286298514 | 2964694429 |
| 01/01 03:00:00 | 8699046422 | 4195585996 | 2888463309 |
| 01/01 04:00:00 | 8257685521 | 4090388655 | 2791558305 |
| 01/01 05:00:00 | 7800628333 | 3972550723 | 2681698644 |
| 01/01 06:00:00 | 7423847169 | 3861430630 | 2577160839 |
| 01/01 07:00:00 | 7867102075 | 3657941844 | 2343602675 |
| 01/01 08:00:00 | 7136744186 | 3535926484 | 2230157441 |
| 01/01 09:00:00 | 6005816200 | 3536819187 | 2106956517 |
| 01/01 10:00:00 | 5165216279 | 3314855376 | 1908690034 |
| 01/01 11:00:00 | 4884056302 | 3199410459 | 1802750402 |
| 01/01 12:00:00 | 4760819983 | 3154846030 | 1762187519 |
| 01/01 13:00:00 | 4712952533 | 3143148165 | 1751818792 |
| 01/01 14:00:00 | 4512780728 | 3078505673 | 1701734600 |
| 01/01 15:00:00 | 4448928789 | 3011070943 | 1638550040 |
| 01/01 16:00:00 | 4511610309 | 3091561646 | 1714195343 |

Figure 3. Example of jEPlus Output - Job Level CSV file

Post-simulation data processing is needed when the simulation results from the modelling software are not the final performance metrics, or when workarounds are used in the modelling software because of its program limitations.

In some cases, post-simulation data process can be as easy as doing simple mathematical calculations based on annual results, such as calculating the building's greenhouse gas emission intensity (GHGI) based on the annual gas and electricity consumption, their corresponding GHG emission rate, and the total building floor area. Another metric that can be calculated is the thermal energy demand intensity (TEDI) by extracting the plant loop heating demand and dividing it by the total building floor area.

There are also times when detailed hourly calculations need to be done or conditional logic need to be applied, such as calculating how much heat can be extracted from the exhaust air stream and how much heat can be injected into the heating loop. Another example is calculating the amount of heat that can be recovered from a heat recovery chiller. Using a programming script for these cases can save a significant amount of calculation time for large scale simulation work.

To batch process post-simulation data for large-scale simulations, the Python program loops over each simulation job to perform customized calculations and

then collect results for each job. Figure 5 illustrates a typical data flow of processing EnergyPlus output hourly data into the aggregated yearly result through custom calculations. The developed code includes the following functions.

- Get the file name and file paths of the desired data file from each simulation job
- For each file path, load the data file and use Pandas (Augsburger 2019), one of the Python library, to perform vector wise operation and time series manipulation on the hourly data
- Store the calculation result for each simulation job in a data structure that's linked to the simulation job ID

The Python script needs to be applied to both the design cases and the baseline case to get the corresponding yearly results. Each individual design case result is then compared with the baseline result to calculate savings (e.g. energy cost saving, source energy saving, GHG emission saving), again by using Pandas to perform column wise calculation. The design case final performance metrics are eventually merged with the

jEPlus's combined result data based on the simulation job ID. Figure 6 shows this process.

Step 3: Large Scale Simulation Data Visualization

Once the data for the parametric study is ready, customized static or interactive graphic data plots can be configured using the Python data visualisation library, Plotly (Plotly Technologies Inc. 2015). Figure 4 shows an example of parallel coordinates graph generated by Plotly below.

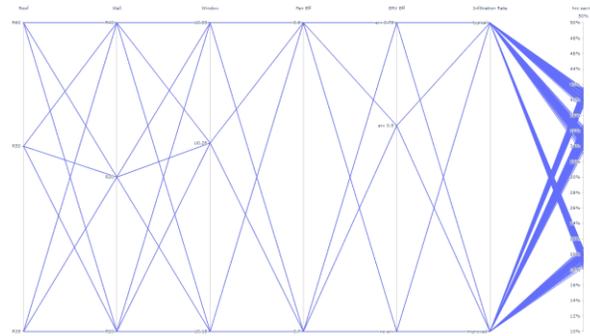


Figure 4. Example of Parallel Coordinates Graph Generated by Python

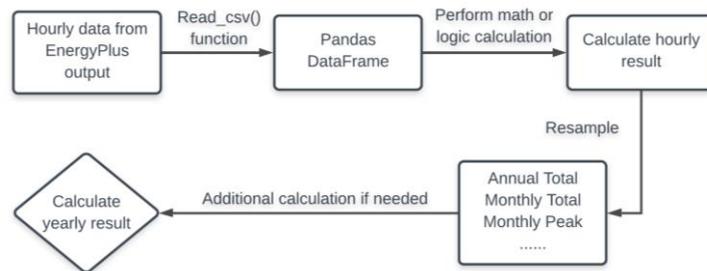


Figure 5. Typical Post-Simulation Data Processing Flow

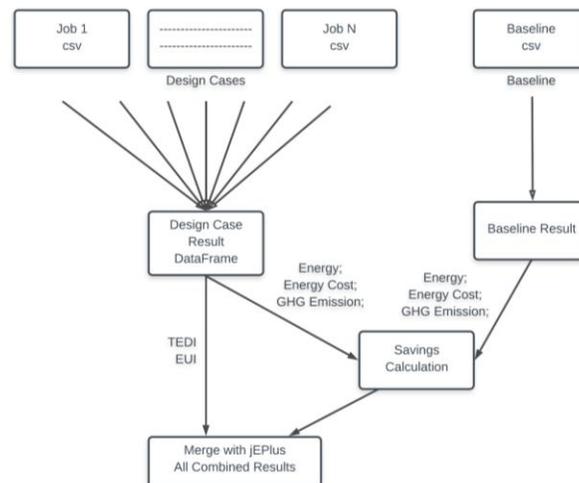


Figure 6. Post-Simulation Data Process applied to Parametric Study and Baseline Comparison

SIMULATION

This post-simulation calculation method has been applied on a pre-design phase energy study of a hospital project in British Columbia, Canada. The building will have a floor area of approximately 20,000 square meters to accommodate various medical programs. A simulation exercise was engaged to assist the design team in finding a solution that will achieve the targeted number of Optimize Energy points under the LEED® v4 rating system. Parametric Design Analysis is used to evaluate various combinations of energy saving measures. The post-simulation calculation is integrated with parametric simulation results to streamline the process of obtaining the desired performance metrics.

The proposed mechanical design uses a heat recovery chiller for cooling and heating, and gas boilers as a backup system for heating. At the zone level, spaces are conditioned via a variable air volume system with reheat. Energy recovery ventilators (ERV) are used to provide ventilation to the spaces and recover energy from the exhaust air. On top of that, exhaust air heat reclaim coils are used to extract heat from the exhaust air as well. Gas-fired humidifiers are used in each air handling loop to add moisture to the supply air to the desired humidity level.

The desired performance metrics are thermal energy demand intensity (TEDI), total energy use intensity (TEUI), energy cost savings, source energy savings and GHG emission savings compared to the baseline case following the ASHRAE 90.1-2010 Appendix G modelling methodology. The Alternative Energy Performance Metric was used in this project to maximize the LEED v4 Optimize Energy points. This metric is beneficial for this project because of the fuel switching impacts of reducing natural gas heating using an electric source (i.e. heat recovery chiller). This metric takes the average of the two highest savings from energy cost, source energy, and GHG emission.

Post-simulation calculations are heavily used on this project because of the software limitations on simulating a heat recovery chiller, gas humidifier, and exhaust air heat reclaim coils. Moreover, they are also used to produce the additional performance metrics required to show compliance to the project's sustainability goal.

The original model is built in Open Studio interface, configured with electric humidifiers, a gas boiler, and a chiller. The post-simulation calculations cover the following scope:

Humidification

- Adjusting the electricity and gas consumption to reflect gas humidifier energy consumption when

only electric humidifier can be used in the model due to software limitation

Heat Recovery Chiller and Heat Reclaim Coils

- Calculating the amount of heat that can be recovered from the heat recovery chiller based on chiller's condenser heat transfer rate
- Calculating the amount of heat that can be reclaimed from the exhaust heat reclaim coil based on the exhaust air temperature and humidity ratio and mass flow rate
- Calculating the amount of electricity required to run the heat reclaim loop
- Deduct the boiler's heating rate based on the amount of free heat available to mimic the water loop heat recovery effect
- Adjusting the electricity demand cost, electricity and gas consumption cost affected by the HVAC work arounds

Performance Metrics

- Calculating the Total Energy Use Intensity (TEUI) based on heating plant loop demand and total floor area
- Calculating source energy and GHG emission based on electricity/gas consumption and predefined factors
- Calculating the savings of energy cost, source energy and GHG emission between the proposed design cases and the baseline case.

DISCUSSION AND RESULT ANALYSIS

Figure 7 and Figure 8 show energy use breakdown for one simulation case before and after post-simulation calculations. It can be seen in Figure 7 that due to the limitations of the modelling software, humidification is reported as an electric end-use, whereas in the design, a gas humidification system is used. Furthermore, the heating end-use does not account for savings from the exhaust air reclaim coils or heat recovery chiller. The EUI before post-simulation calculations is 760 kWh/m².

Conversely, in Figure 8, humidification is recorded properly as a gas end-use. After the post-simulation calculations, the savings from the heat recovery devices are captured in the heating end-use, which is significantly smaller. The EUI after post-simulation calculations is 562 kWh/m², which is a 26% reduction. The savings from the energy efficient mechanical systems are reflected through the application of post simulation data processing. Scripting isn't the only way to achieve these corrections. Modelers should first seek appropriate built in modules to be used in the energy model to reflect these savings however when there is no other options, scripting provides the opportunity to

model parametric and auto correct a wide variety of deficiencies in EnergyPlus.

Scripting provides a wide-open window for corrections, ranging from simplified rough estimation to detailed calculations. It is important to leverage the level of accuracy and level of details needed based on the project needs. Since the presented project is at pre-design state, the implemented post simulation calculation does not take into account factors including partial load curves, etc.

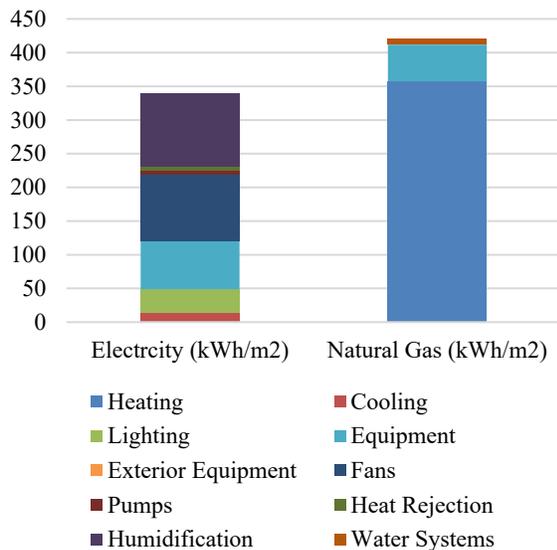


Figure 7. Energy Use Breakdown - Before Post-Processing

The calculation was performed on all 960 simulation cases and the results are shown in Figure 9. A parallel coordinates graph is used to present multi-dimensional data to see interactions between the design choices and the final performance metrics. The first six axes represent different parameters for design choices. Each line on the graph represents one simulation case. The result of each simulation case is shown in the last three axes.

All simulation cases shown in blue have the heat recovery chiller as part of its design; whereas, the orange cases do not have a heat recovery chiller. It is interesting to note that the heat recovery chiller does not impact the TEDI value; this is to be expected as it does not theoretically affect the building’s thermal demand. From this graph, we can see that the ERV has a significant impact on the TEDI and EUI. This impact can be seen in the two separate orange or blue bands.

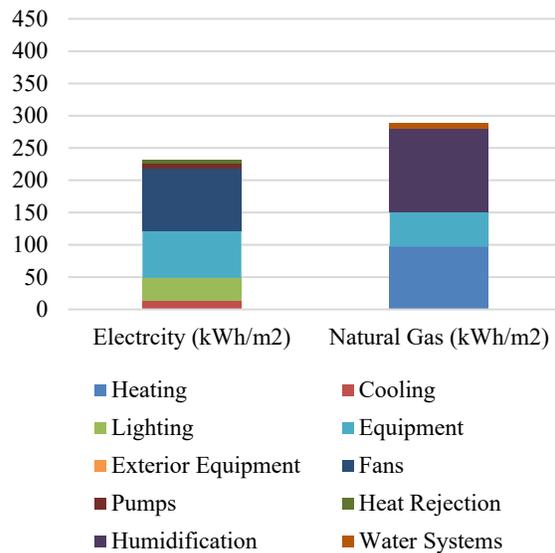


Figure 8. Energy Use Breakdown - After Post-Processing

The performance target for the hospital project was 12 LEED v4 Optimize Energy points (i.e. 24% savings using the Alternative Energy Performance Metric). On the graph, this percent savings is shown as “LEED Savings %”. Figure 10 shows the Design Options that meet the project target. All the selection Design Options are in blue; therefore, the project cannot meet its targets without a heat recovery chiller. The post simulation scripting process has helped the design team to estimate the energy savings of using a heat recovery chiller while it is impossible to directly model it in the selected simulation interface.

CONCLUSION

The powerful combination of parametric simulation and post-simulation data processing pushes the limitations of simulation tools and provides practitioners the flexibility to customize their simulation results and data visualizations based on specific project needs. The use of a scripting tool significantly reduces the time needed for post-simulation calculations on a large scale while reducing the risk of errors at the same time.

This is an essential tool for projects that have mechanical systems that cannot be modelled in the building energy modelling software. As a conservative approach, one can omit these savings; however, this may jeopardize the project in meeting its performance targets.

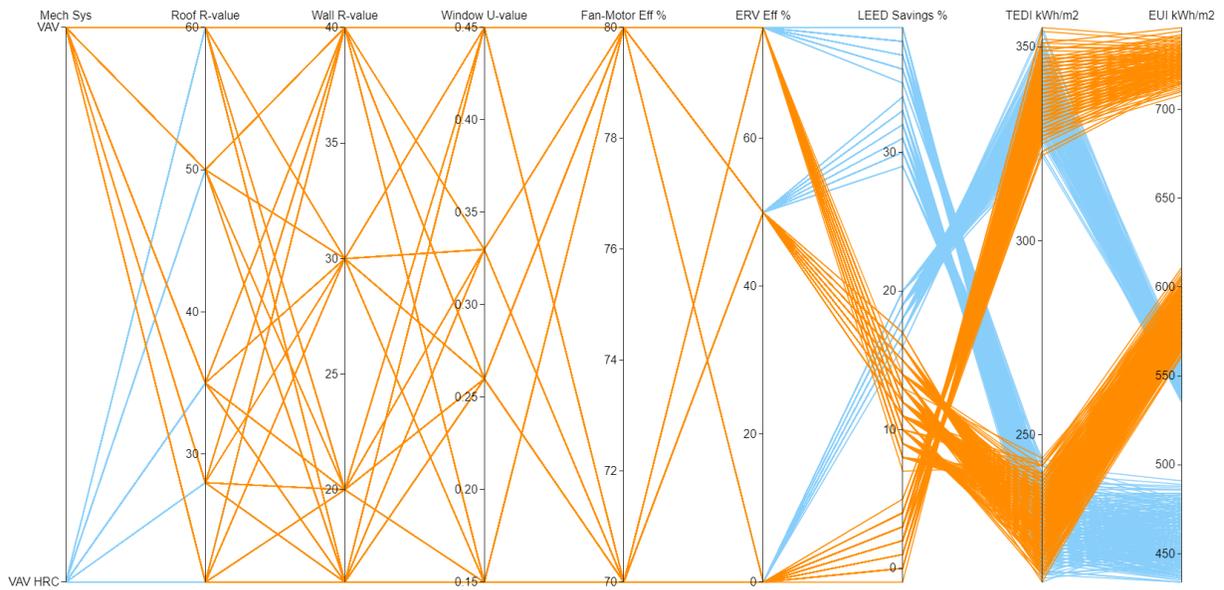


Figure 9. Parametric Design Analysis Results – All Design Options

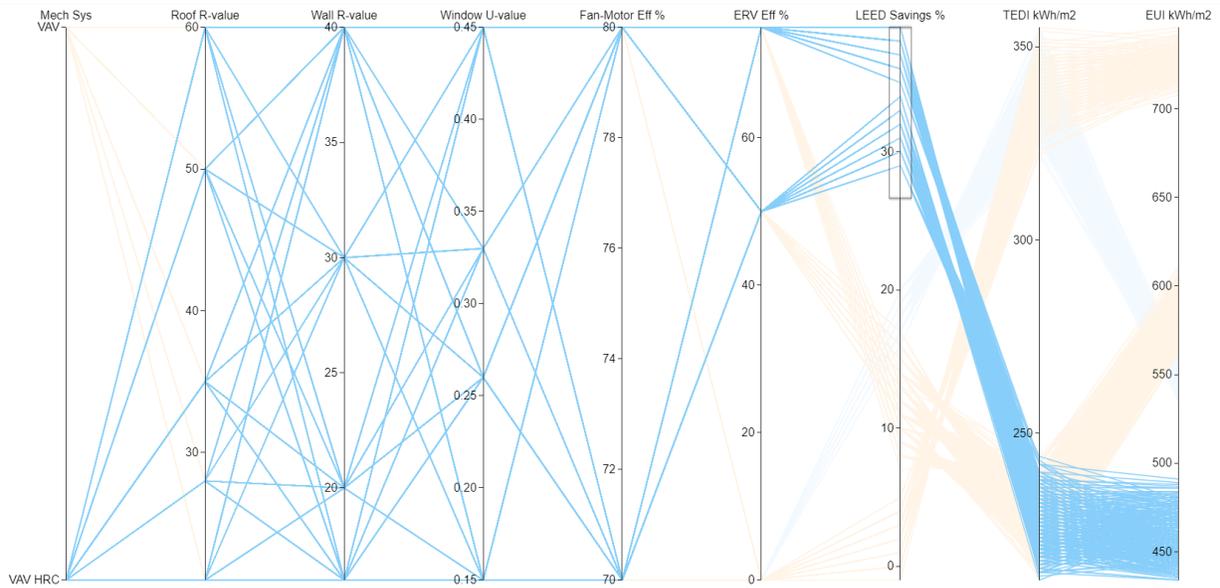


Figure 10. Parametric Design Analysis Results – Design Options that Meet Project Target

REFERENCES

- Augspurger, Tom et al. 2019. "pandas." *Python Data Analysis Library*. <https://pandas.pydata.org/>.
- Chan, Jeanie, Andrea Frisque, and Anika Jang. 2019. "Designing to TEDI, TEUI, and GHGI Performance Metrics." *Building Simulation 2019*. Rome, Italy: IBPSA.
- Chan, Jeanie, Randy Irwin, Andrea Frisque, and Eric Dunford. 2018. "Cost-effective net-zero energy/GHG solutions for retrofit projects using parametric whole building energy simulations." *eSim 2018*. Montreal, Canada: IPBSA. 558-567.
- Chiesa, Giacomo et al. 2019. "Parametric Optimization of Window-to-Wall Ratio for Passive Buildings Adopting Scripting Methodology To Dynamic-Energy Simulation." *Sustainability* 11: 3078.
- Miller, Clayton, Christian Hersberger, and Marcus Jones. 2013. "Automation of Common Building Energy Simulation Workflows Using Python." *13th Conference of International Building Performance Simulation Association*. Chambéry, France: IBPSA. 210-217.
- Plotly Technologies Inc. 2015. *Collaborative data science*. Montréal, QC: Plotly Technologies Inc.
- Roth, Amir et. al. 2018. "Scripting Frameworks for Enhancing EnergyPlus Modelling Productivity." *2018 Building Performance Analysis Conference and SimBuild co-organized by ASHRAE and IBPSA-USA*. Chicago, USA.
- Roth, Amir, David Goldwasser, and Andrew Parker. 2016. "There's a measure for that!" *Energy and Buildings* 321-331.
- Energy, U.S. Department of. n.d. *EnergyPlus*. Accessed February 9, 2018. <https://energyplus.net/>.
- Zhang, Yi, and Ivan Korolija. n.d. *JEPlus – An EnergyPlus simulation manager for parametrics*. Accessed February 9, 2018. <http://www.jeplus.org>.