

PNNL-30138

A Parallel Computing Infrastructure for Building Energy Simulation

July 2020

Jian Zhang
Eli T Brock
Yunyang Ye

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.** Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062;
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
email: orders@ntis.gov <<https://www.ntis.gov/about>>
Online ordering: <http://www.ntis.gov>

A Parallel Computing Infrastructure for Building Energy Simulation

July 2020

Jian Zhang
Eli T Brock
Yunyang Ye

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Abstract

In order to study grid-interactive efficient buildings, Pacific Northwest National Laboratories (PNNL) needs an infrastructure for urban-scale building energy modeling. Such an infrastructure should be fast, scalable, and easy-to-use. Given a set of data from the Energy Information Administration's Commercial Building Energy Consumption Survey (CBECS) and tool to translate survey data into simulation inputs, this project aimed to conduct the simulation of the entire dataset in parallel. Before running the simulations, the necessary software was bundled into a container for use on the PNNL supercomputing network. Then, the parallel simulation workflow was designed using GNU Make, a file creation software, and submitted to a supercomputing partition which could run hundreds of simulations simultaneously. The EnergyPlus simulations output hourly electric meter data for each CBECS sample, which represents the electricity consumption of similar commercial buildings across the United States. Analyzing and visualizing the meter data is important to the future of the work, and this project wrote code to make common analysis methods simple, fast, and accessible. Moving forwards, the model will need to be expanded to include data from other sources and its accuracy will need to be improved and eventually validated.

Introduction

The commercial building sector accounted for approximately 18% of site electricity consumption in the United States in 2019 [1]. As such, they present an exciting opportunity for improvements in energy efficiency. Efficiency-enhancing components, regulations, and technologies have the potential to precipitate large-scale decarbonization and energy cost savings. Building energy simulations are an effective and cost-effective way to evaluate the effectiveness of these innovations, which is critical for informing policymakers, engineers, and other decision-makers.

The building energy simulation infrastructure at Pacific Northwest National Laboratory (PNNL) is built primarily to inform building energy codes by analyzing potential efficiency upgrades on the scale of an individual building. Since components included in these codes are generally widely adopted, the simulation results can be used to indirectly forecast large-scale energy savings. However, certain grid-interactive efficient building technologies can drive energy savings on an urban scale by optimizing the shape of the load profile, or energy demand over time, seen by electricity suppliers. A common example is heating water – current water heating systems retain heat very well, so water can be heated at any time throughout the day and then consumed as needed. Shifting water heating away from peak hours optimizes the load profile by reducing the generation capacity suppliers need to maintain. Since the potency of these technologies is predicated on buildings' relationships with each other and with suppliers, their effects are best captured by simulations of entire building stocks rather than individual buildings.

Running simulations on an urban scale is quite resource-intensive, given the number of building samples necessary to accurately reflect the makeup of the commercial building stock. The purview of this project is the construction of a new parallel computing infrastructure to facilitate a fast, replicable simulation workflow along with instructive analysis and visualization processes.

Building Samples

A set of weighted samples, used to represent the building stock, constitutes the basis for grid-scale simulations. These samples can come from a variety of sources, so long as they include the necessary energy-related information to inform the model. As a starting point, the project uses the Energy Information Administration's Commercial Building Energy Consumption Survey (CBECS) dataset from 2012 (the survey was re-administered in 2018, but the results have yet to be released) [2].

The CBECS encompassed over 6,000 buildings and collected design and operation characteristics, which are used to inform the simulation model. Importantly, the CBECS data for each surveyed building also includes energy use data – which will eventually serve as a benchmark to validate the models – and a weight parameter establishing how many similar buildings are represented by the sample.

The processes for extracting the relevant characteristics and translating them into simulation input format were completed before the author began work on the project. However, executing these conversion processes in a parallel computing environment constitutes its own challenge.

Parallel Computing

Containerized Computing Environment

The conversion process is complex and requires a set of software tools to complete. Among these tools are Python and Ruby, the programming languages which direct the process, and some additional bundles of code called “packages” which expand those languages. The workflow also requires OpenStudio, a suite of building energy modeling software tools developed by the National Renewable Energy Laboratory (NREL).

Conventionally, users simply download whatever software dependencies they need onto their personal machines. However, this approach was impractical for this project for two main reasons.

First, downloading software and installing packages requires administrative privileges. While users typically have these privileges on their local machines, supercomputing clusters like PNNL's are a shared resource; laboratory researchers can use them, but they are independently administered. Consequentially, any software installation would involve a request to the administrators which, aside from being slow and potentially bureaucratic, is not guaranteed to be approved.

Secondly, the dependencies for this project are complex enough to make their installation on any machine especially tedious and prone to error. As outlined in the introduction, the workflow for this project should be easily replicable. As such, it is prudent to avoid inconsistent computing environments associated with different researchers independently installing software.

For this project, software containers were an attractive alternative to conventional installations. Putting software in containers eliminates complicated installation and ensures that every user is working in the same environment. Containers are the difference between individually collecting everything you need for work before leaving home and keeping all those things in a backpack, ready to pick up when you need it. The most popular container creation and management software is called Docker. However, this project used Singularity, which is very similar to Docker but built for high-performance computing applications.

NREL maintains a Singularity container for OpenStudio, which is often used by other PNNL building simulation projects. However, it lacked some of the Python and Ruby functionality required for this project, so a new container had to be built. Using NREL's container as a baseline, the author installed Python packages and Ruby extensions (called “gems”) until the

container supported the entire conversion workflow. The container-building process was quite involved and was critical for establishing a working parallel simulation infrastructure.

Parallel Computing Workflow

Processing a single sample starts with using the data stored in its native data format – JSON in this case – to inform the creation of an Input Data File (IDF) for the EnergyPlus simulation software. Once the IDF is prepared, it is passed into EnergyPlus, which simulates a year of that building's energy consumption and generates a set of output files characterizing its energy use.

Executing this process for all 6,720 simulations synchronously (one-by-one) would take hours and will only become more time-intensive as more cases are added in the future. Using the supercomputer, however, execution can happen asynchronously (in parallel), dramatically speeding up the time frame. Specifically, using all 12 supercomputing nodes available on the partition which supports Singularity, 288 jobs can run at once and the entire CBECS dataset simulates in about half an hour.

The parallel workflow is organized using GNU Make, a software tool which simplifies file generation by establishing a system of dependencies. In this case, one command tells make to run all the simulations. It then attempts to simulate each sample using its IDF. If the IDF does not exist or is outdated, it then knows to run the conversion first. There are two primary benefits to using Make. First, it abstracts away the simulation process into one line of code, rendering it accessible for unfamiliar researchers. Secondly, it eliminates redundant processes; if only some of the JSON files or IDFs have been updated since the last run, Make only runs those simulations, as the others would not change.

Data Aggregation

While EnergyPlus outputs several different types of data, this project is concerned with the electric meter data, generated hourly. Postprocessing code, written for the project, facilitates quick analysis and visualization of the meter data. After loading this code, meter data for a handful of components purposes can be extracted and visualized for any time frame in just a few lines of code. Data aggregation, then, is as simple as combining the weighted sum of each sample's total hourly electricity consumption. Figure 1 shows the individual weighted load profiles for each sample on July 30, 2012 (the year the CBECS data was collected):

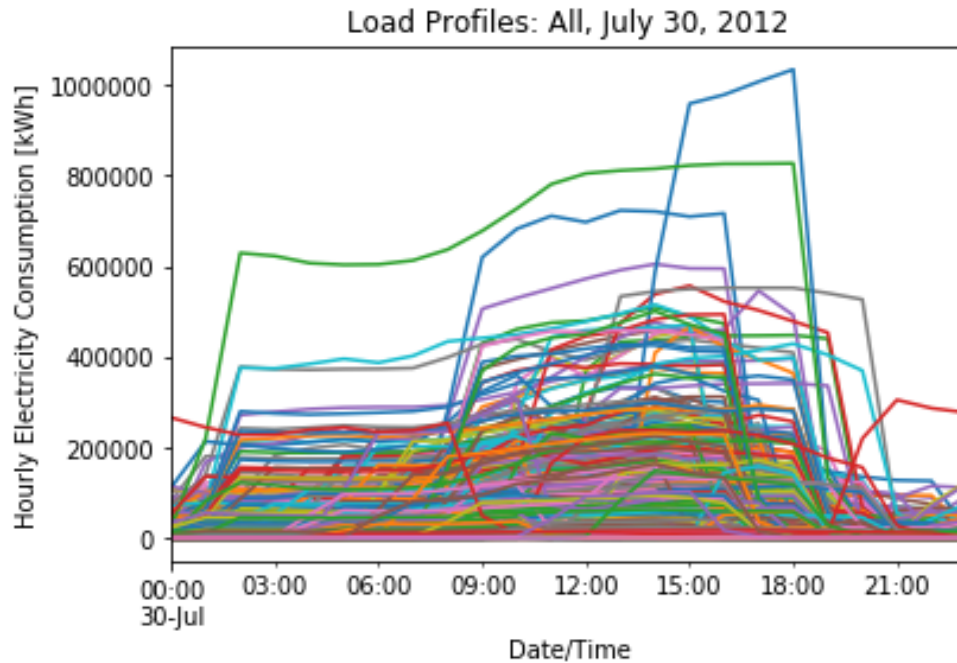


Figure 1: Individual load profiles, July 30

Figure 2 gives the sum of all that data for every hour in the day:

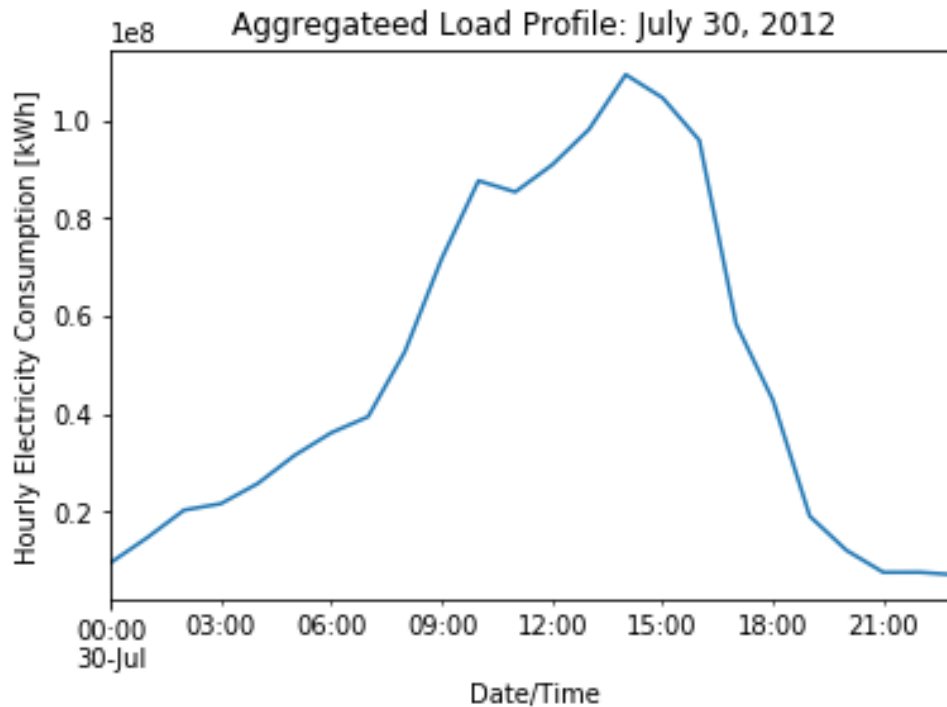


Figure 2: Aggregated load profile, July 30

That curve tracks with what can be expected: energy use rises during the day, peaking at mid-afternoon, and then tapers off at night. Figures 3 and 4 give the same aggregated load profile over the course of the entire month of July:

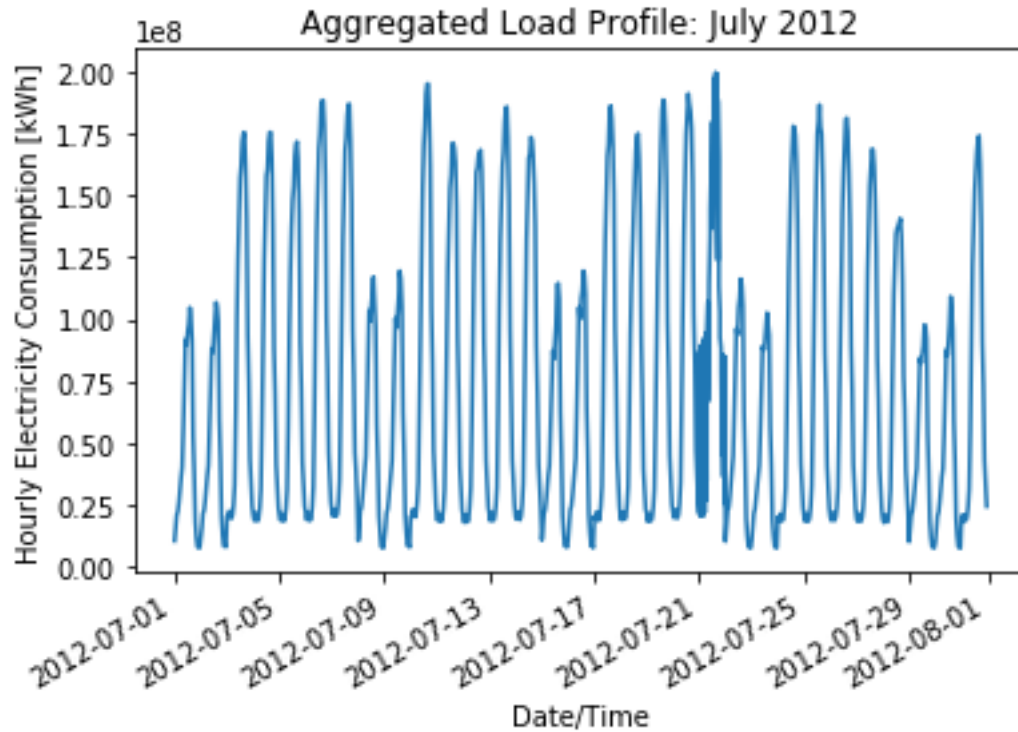


Figure 3: Aggregated load profile, July

Again, the data makes sense; notice the lull on weekends. Figure 4 expands the timeframe to the entire year:

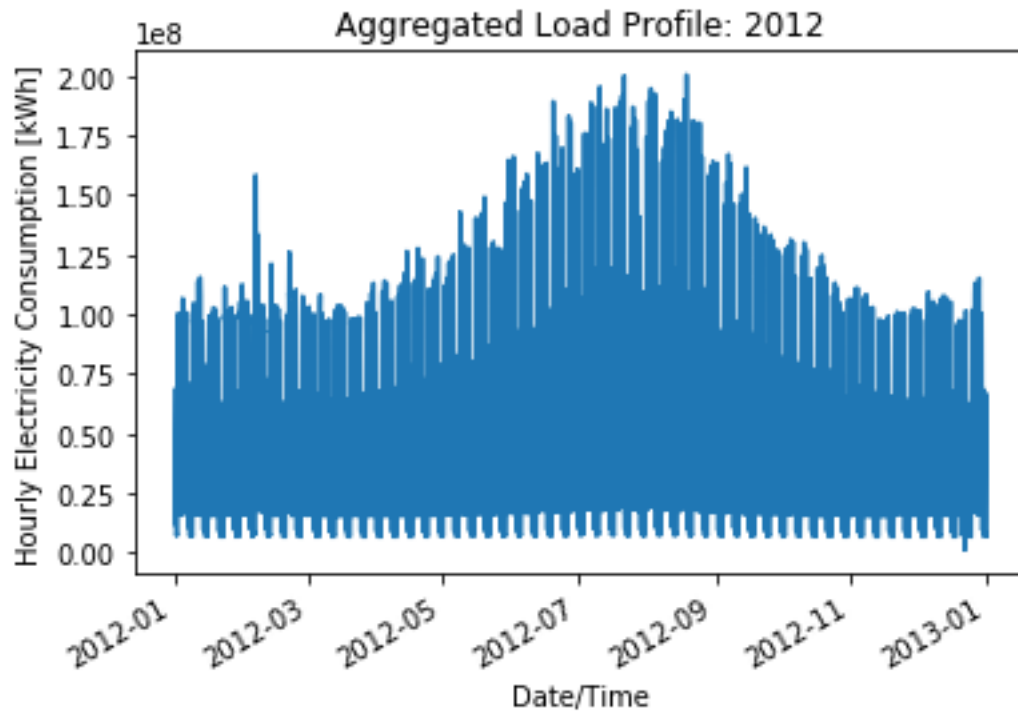


Figure 4: Aggregated Load Profile, year

The peak during the summer months is a consequence of electricity-intensive cooling.

Data Comparison Infrastructure

At this stage in the larger product, the building models are not well-developed enough that it is reasonable to evaluate the accuracy of the simulation results. Furthermore, the simulation workflow does not yet account for location diversity. In other words, if a CBECS sample of a school is taken in Seattle, all the buildings that sample represents are also assumed to be in Seattle – a costly assumption, given the different energy needs of buildings in different climates.

However, it was worthwhile to establish some basic comparison tools so they would be ready to use in future steps. In its current form, the data has the correct format to support a validation workflow, but it should be noted that the data in the following figures is not especially meaningful at this stage.

The postprocessing code includes the percent error in total simulated energy consumption as compared with the actual yearly energy consumption from CBECS. Figure 5 shows the current distribution of percent errors, both unweighted and weighted by total energy consumption.

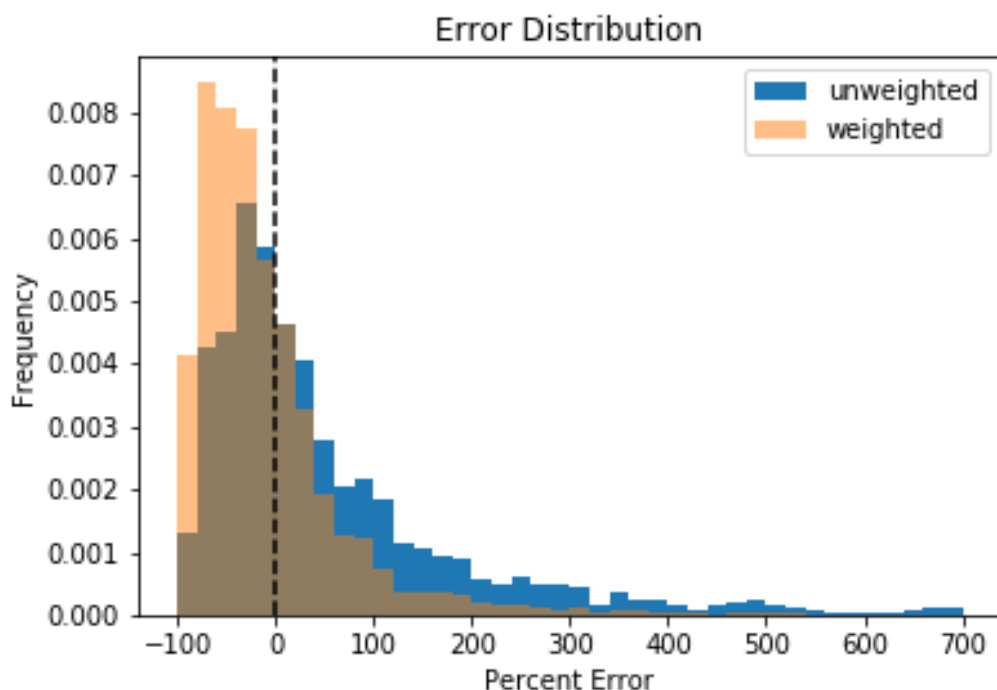


Figure 5: Preliminary percent error distribution

Again, while the data is not ready to be analyzed, that this figure was produced with a few lines of code demonstrates a valuable post-processing capability. It is similarly straightforward to get data on actual and simulated energy intensity, or energy consumption per unit of area, as shown in figure 6:

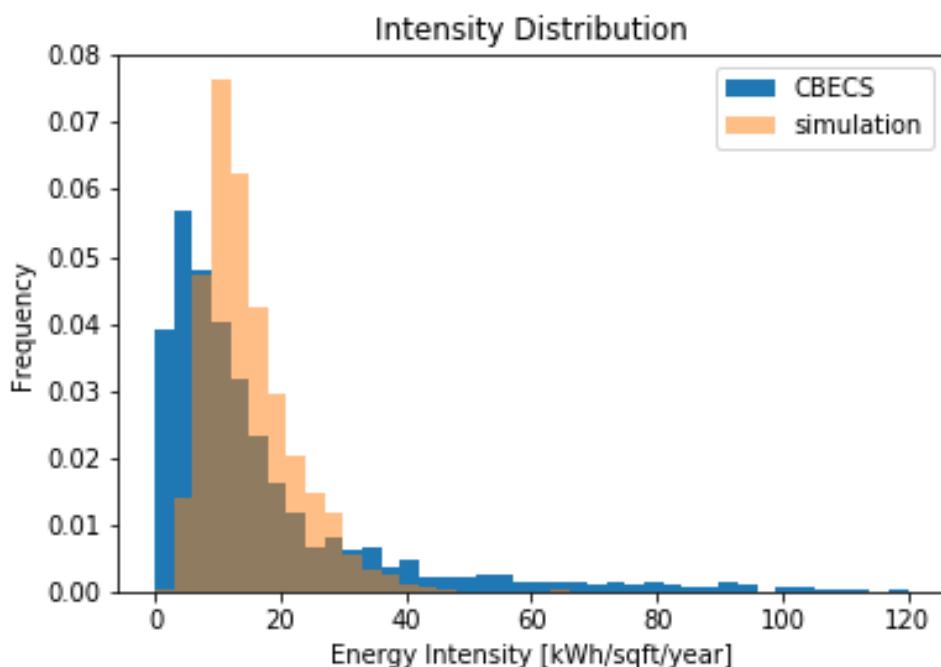


Figure 6: Preliminary energy intensity distribution

In future work, this postprocessing infrastructure can inform improvements to the building models.

Moving Forward

In the immediate future, the simulation workflow will need to be expanded to include more HVAC systems, as it currently supports only three, comprising about half of the CBECS samples. Beyond that, including building samples from other sources will make the simulation more robust, accurate, and representative. Neither change constitutes a fundamental change in the workflow, which is built to be scaled up.

As mentioned, the building models will eventually need to be validated against actual energy consumption data. In the long-term, the simulation infrastructure developed during this project will make future development straightforward and accessible.

References

[1] EIA, Annual Energy Outlook 2020.

<https://www.eia.gov/outlooks/aeo/data/browser/#/?id=2-AEO2020®ion=1-0&cases=ref2020&start=2018&end=2050&f=A&linechart=~~~~~ref2020-d112119a.12-2-AEO2020.1-0~ref2020-d112119a.27-2-AEO2020.1-0~ref2020-d112119a.53-2-AEO2020.1-0~ref2020-d112119a.71-2-AEO2020.1-0&map=ref2020-d112119a.5-2-AEO2020.1-0&ctype=linechart&sourcekey=0>

[2] CBECS, <https://www.eia.gov/consumption/commercial/>

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354
1-888-375-PNNL (7665)

www.pnnl.gov