Development of Automated Procedures to Generate Reference Building Models for ASHRAE Standard 90.1 and India's Building Energy Code and Implementation in OpenStudio

Andrew Parker[1], Philip Haves[2], Subhash Jegi[3], Vishal Garg[3], Baptiste Ravache[2]
[1]National Renewable Energy Laboratory, Golden CO, USA
[2]Lawrence Berkeley National Laboratory, Berkeley CA, USA
[3]International Institute of Information Technology, Hyderabad, India

Abstract

This paper describes a software system for automatically generating a reference (baseline) building energy model from the proposed (as-designed) building energy model. This system is built using the OpenStudio Software Development Kit (SDK) and is designed to operate on building energy models in the OpenStudio file format.

Introduction

The assessment of beyond-code performance, such as the ASHRAE Standard 90.1 Performance Rating Method (ASHRAE, 2013), and compliance with building energy codes, such as India's ECBC (Mathur *et al.*, 2010; Kumar *et al.*, 2010), are important roles for building simulation. In each case, the process involves comparing the simulated performance of the proposed design to the simulated performance of a reference baseline building. The baseline building is a minimally compliant building design that is derived from the proposed design using a well-defined procedure. A key step is the generation of the reference building description from the design, using a set of rules defined by the code or standard.

Historically, this has been a tedious manual process plagued by errors in both data entry and interpretation of the code or standard. The prevalence of these errors requires certification or rating agencies to spend significant effort reviewing the baseline models (LEED review is one example), and errors typically result in requests for corrections, which cost design teams time and money.

The paper describes a software system designed to automatically generate a baseline building model for the assessment of 'above-code" energy performance, as defined in Appendix G of ASHRAE Standard 90.1. This procedure has been implemented as a Measure in the OpenStudio (OpenStudio, 2016) suite of tools and libraries for use with the EnergyPlus (EnergyPlus, 2016) whole building simulation program. The software system has been modified to address ECBC, which is based on ASHRAE Standard 90.1, though with national and regional variations to account for Indian requirements and practices

The paper also describes a systematic testing procedure that has been developed and applied to verify correct implementation of both the ASHRAE Standard 90.1 Appendix G rating procedure and the ECBC standard.

Goals

The goal of the system was to automate the process of creating the baseline model so that the users of the system do not need to spend the time to perform the task manually, nor do they need to understand the sometimes cryptic language or complex rules described in the code or standard that define the baseline model.

Initial Challenges & Genesis

The genesis of the initial system stemmed from a problem facing the Xcel Energy of Colorado Energy Design Assistance (EDA) Program. In the EDA program, the utility (Xcel Energy) paid building energy modelers to work with design teams and building owners to convince them to implement more efficient designs. As the energy codes in the program territory became more stringent and complex, the time spent creating the baseline model began to take time away from analyzing efficient designs. To combat this issue, Xcel Energy funded the development of a system to automate the process. This system was not built from scratch; much of the initial foundation was laid by a previous NREL project to create the DOE Prototype Buildings (Thornton, 2011) in OpenStudio format.

Because the energy codes in many countries are modeled on ASHRAE Standard 90.1, the target of the initial software system design, it was possible to extend the system to cover these other codes. The benefits of extending an existing system rather than building a separate system were obvious, and both India and Canada have begun implementing their respective energy codes – ECBC and NECB (NRC, 2015) - in the system.

ECBC Challenges

The adoption of building energy codes and the use of simulation to demonstrate compliance are recent developments in India and so there is a lack of knowledge and experience among designers regarding analysis of building designs based on code requirements. The capability to use building simulation to quantify savings based on energy efficiency parameters as defined by the code is very limited. In addition, the technical capabilities of the agencies tasked with implementing building code are not adequate to verify compliance. A software system that performs automated verification of the compliance of the proposed design then reduces the task of building inspectors to verifying

that the building as built is consistent with the approved design. Some of the major challenges faced in implementation of ECBC are discussed in (Mathur *et al.*, 2010).

Software System

Overview of OpenStudio

OpenStudio is a building energy modeling software comprised of two key components. First, OpenStudio has a graphical user interface (GUI) that users can employ to describe the building design. This is the component that most energy modelers are likely to be familiar with. The second component is the OpenStudio application program interface (API), which is based on the Ruby programming language. The Ruby API allows users to write software that automatically changes the building energy model.

One advantage of using a common programming language, such as Ruby, rather than a custom-made language such as that used by, for example, California's CBECC-Com software (Architectural Energy Corporation, 2016), is that there are many resources for learning the language, and common questions can often be answered by a quick online search.

System Architecture

The system is divided into two parts. The first is a series of lookup tables that store information about various aspects of the building governed by a code or standard. For example, minimum efficiency for a chiller can be looked up based on standard, chiller type, condenser type, and capacity. Because ASHRAE 90.1, ECBC and NECB are so similar, data from these standards can be stored in the same tables.

The second part is a series of computer methods to apply the logic present in the standard. For example, the method "*AirLoopHVAC.energy_recovery_required?*" checks if the standard requires energy recovery for this particular system. In some cases, these functions need to extract information from the lookup tables. These methods have access to all of the information in the proposed building model, including geometry, construction materials, internal loads and schedules, HVAC systems, etc. This enables the complex logic and exceptions found in the standards to be accounted for. Again, because ASHRAE 90.1, ECBC, and NECB are very similar, much of the logic in the methods can be shared.

One key aspect of the system is that it is completely open-source (OpenStudio-standards, 2016). This means that the logic can be inspected and reviewed in detail by any interested parties. These parties can, and have, suggested areas of improvement or pointed out problems so that they can be addressed.

Supported Standards

The system currently supports Appendix G of ASHRAE 90.1-2007, 2010, and 2013. The Prescriptive and Whole Building standards from India's ECBC-2007 energy code are being added, as is the Canadian NECB.

Integration with OpenStudio Measures

As with many applications built on OpenStudio, the process of automating the generation of the baseline models is expressed as an OpenStudio Measure. OpenStudio Measures are scripts that automate portions of the energy modeling workflow. These scripts conform to a specific interface (NREL, 2014) that takes as energy model as well as user arguments as input. The scripts make use of the OpenStudio Ruby API to edit the energy model and the output is a modified energy model. OpenStudio Measures can be chained together to implement a complete building energy modeling workflow.

For example, the measure might find the default construction used by roof surfaces in the model, copy this construction and add insulation material to the outside, then set the new construction with added insulation as the default construction to be used by the roof surfaces.

Figure 1: OpenStudio Measure operation.

OpenStudio Measures can help energy modelers by:
- Reducing modeling time and cost
- Finding deeper savings
- Lowering administrative and training costs
- Maintaining quality and consistency

System Process Overview

A general overview of the automated procedure to generate a baseline building model for ASHRAE Standard 90.1 Appendix G and India's ECBC is as follows:

Manual (user)

1. Before the process begins, the user must assign space types (e.g. Open Office, Corridor, etc.) from a predefined list to each space in their model. The user must also define the type of frames used for each window from a predefined list.

Automated

2. Any external shading devices are removed from the model.
3. Skylights and windows are shrunk to the maximum allowed size, in the case of windows, by raising the sill height (so as not to affect daylight penetration into the space) and, in the case of skylights, by shrinking toward their centroids.

4. Internal loads (occupants, lights, and internal equipment) are set to code-maximum allowed values. Not all standards govern all types of loads. For some standards (such as ASHRAE 90.1), user-defined schedules are used. For other standards, the user-defined schedules are replaced with prototypical schedules.
5. Potentially daylit areas are calculated based on the geometry of the model. This can be a complex problem due to the wide variety of fenestration layouts in real buildings and due to the fact that not all internal aspects of the building (tall shelving, cubicles, etc.) are explicitly included in the model.
6. Each surface in the model has a particular type of construction assigned, and the insulation in the construction is set such that the overall u-factor meets the code minimum value.
7. Most existing HVAC components in the model are removed, with the exception of exhaust fans.
8. In general, the types of the baseline HVAC system(s) are determined based on a combination of space category, building area, and number of stories in the building. Typically, those spaces that differ significantly from the rest of the spaces are assigned separate single-zone HVAC systems.
9. The baseline HVAC system(s) are put into the model, typically one per story. During this step, only properties of the systems that are not governed by the standard are set, based on the proposed design.
10. Properties used for air system sizing (design supply air temperature, design supply air temperature difference) are set.
11. HVAC system controls that are not dependent on system capacity or flow rate are set.
12. Properties used for water system sizing are set.
13. A sizing run is performed. In this process, the sizing routines in EnergyPlus determine the capacities and flow rates necessary to meet the heating and cooling loads in the baseline building on a design day.
14. HVAC system controls that are dependent on system capacity or flow rate are set. These include demand control ventilation, energy recovery ventilators, and motorized outdoor air dampers.
15. If chillers, boiler, or cooling towers are included in the baseline HVAC system, the single items of equipment used during the sizing run are split into multiple discrete items based on the rules of the standard.
16. An additional sizing run is performed to determine capacities of the newly created chillers, boilers, cooling towers, and their pumps.
17. The code maximum allowable fan and pump power is calculated and these values are set by adjusting the fan or pump pressure rise.
18. The minimum efficiency and performance curves for each item of HVAC equipment in the air and water systems are assigned based on capacity and equipment type.

Output

One goal during the system design was to be both robust and transparent. For this reason, at the end of the baseline automation process, a set of informative information, warning, and error messages is presented to the user. These help the user review the changes that were made in the generation of the baseline building model, both to look for mistakes and to understand exactly what happened. The authors have received feedback from experienced energy modelers that these messages have given them confidence in the system because they could see what it was doing.

Limitations

As experienced energy modelers know, it is possible to encounter scenarios that are either not represented well in the proposed building energy model, or that are uncommon and therefore not handled by the system by design. During the development process, the developers of the system strove to cover the most common scenarios. Those scenarios that were intentionally not covered are documented in the output messages that are displayed to the user. As the system is still under incremental development, presenting a comprehensive list of limitations in this paper does not make sense, however, a few example limitations as of the time of writing include:

- Optimal start of HVAC systems
- Fan power allowances for MERV filters in proposed
- Correct handling of laboratory ventilation requirements

As modelers use the system and encounter more scenarios that need to be supported, they can communicate these needs to the developers, or even modify the software to support these things themselves.

Testing

In order to ensure that the baseline automation system worked as intended, test cases were created. Experienced energy modelers working on actual projects were hired to make statements (known as assertions in computer programming) about the expected properties of various aspects of their models after being run through the baseline automation process. These assertions were based on their professional experience and understanding of the energy code. The assertions cover a wide variety of rules ranging from material properties such as wall U-value to more complex rules such as distribution of fenestration on walls and configuration of baseline HVAC systems.

Sets of programmatic tests were created to test these assertions. If an asserted condition was not met, the test failed. During the development of the system, these tests helped to identify areas where changes were needed.

In addition to those test cases inspired by real buildings, several building models were created from reference building models (Deru, 2011) by altering various objects, in order to target specific areas of the energy code that were not tested by the real buildings. These additional models were based on the rule set implementation tests proposed for the California Energy Commission's Title 24 energy code (CEC, 2015) and enable a wider coverage of the rule set, including climate zones, building types, HVAC equipment or envelope properties that are encountered relatively infrequently.

Twelve custom proposed design models were created, covering five climate zones, six building types and four different HVAC systems. Those models allow for the evaluation of 134 assertions; in other words, 134 individual rules or specific code values (such as U-factor, window-to-wall ratio, lighting power density) were able to be tested.

One example of the tests created is to take the small office reference building, change the envelope materials from a concrete structure to a wood-framed structure with poor insulation, increase the fenestration area over the energy code threshold and add overhangs. The expected baseline model created is tested for walls U-value, fenestration area and lack of exterior shading elements (Figure 2).
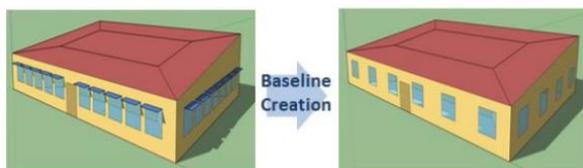

Figure 2: Compliance test example. On the right, the prototype design and on the left the expected baseline model.

The ongoing purpose that these tests serve is to ensure that future changes to the system do not break existing functionality ('regression testing'). For this reason, an automated system that runs the full test suite whenever a change to the system is made has been implemented. If any of the tests that previously passed suddenly fail, the changes are not incorporated until the issues that caused the failures are corrected. This test suite is invaluable when trying to coordinate a complex software effort between different groups of people, often in different countries, working on the same software at once.

Implementation Challenges

ASHRAE 90.1

One of the key challenges in automating the baseline automation process for ASHRAE 90.1 was handling the calculation of potentially daylit areas. This was difficult both because the geometry of fenestration in real buildings is often complex, and because not all aspects of the building design that impact daylighting (shelving, cubicles) are included in the energy model.

Another key challenge was the correct treatment of ventilation rates in the baseline model, particularly when the baseline model had different HVAC system types than the proposed model. The approach that was taken is an approximation of what a 'typical' design engineer would do; this is an area of the software system that bears further scrutiny.

ECBC

ECBC has a lot of similarities with ASHRAE 90.1 2004. The code has 5 sections - Envelope, HVAC, Service Hot Water & Pumping, Lighting and Electrical Power. In each section there are mandatory provisions.

ECBC has two compliance approaches: the Prescriptive method and the Whole Building performance method. The Whole Building performance method has similarities with Appendix G of ASHRAE Standard 90.1 - 2004.

In the Envelope section of ECBC, the U Factors for Roof and Wall and Fenestration are defined. These values are available in lookup table in similar way as in Appendix G. There are two major differences between ECBC and ASHRAE Standard 90.1 Appendix G in the Envelope section:

1. Opaque assemblies such as roof, floors, doors, and walls shall be modeled as having the same heat capacity as the proposed design. The software system has to extract the properties of walls and roofs from the proposed design and create the corresponding components for the baseline.
2. The U-Factor for opaque assemblies is different for 24 hour use buildings and daytime use buildings. A new variable was added to hold the use period information.

In the HVAC section, a similar methodology has been followed in ECBC as in ASHRAE Standard 90.1 Appendix G. Besides mandatory requirements related to minimum equipment efficiencies, there are sections specifying the controls, HVAC system mapping, economizers etc., However, there is one major difference. If the baseline system type uses chilled water and the proposed building has an air cooled chiller, then the baseline model shall have an air cooled chiller. Otherwise the baseline model shall have water cooled centrifugal chillers. If the proposed building has a combination of air and water cooled chillers, then the baseline model shall have a combination of air and water cooled chillers in the same proportion.

In the Service Hot Water and Pumping section, there is a requirement that the standard design shall have a solar system capable of meeting 20% of the design load.

The Lighting section has mandatory and Prescriptive requirements. The lighting power in standard design can be determined by using Space Function Method or Building Area Method. In ECBC there is a mandatory requirement for Controls in Daylighted areas. These controls need to be included in the baseline model.

Adding Support for Additional Energy Codes

The authors firmly believe in the value of a collaborative effort and are very interested in helping others add support for their energy codes to this system. The first step in adding support for another energy code is simply contacting the authors. The process to add a new energy code generally follows these steps:

1. Add basic energy code data (construction properties, equipment efficiency values, etc.) to the lookup tables. This step requires no software programming experience.
2. Chain together the previously created software methods that most closely match the new energy code. Often, these will be slightly different than the new energy code but will provide a reasonable starting point. This requires basic software programming experience.
3. Modify the previously created software methods to reflect the specifics of the new energy code. This requires more advanced software programming experience.
4. Create sample building energy models and unit tests that can be used to ensure that the software is working correctly, and to ensure that future changes do not break functionality.

Conclusion

In the US, baseline model generation is a significant part of the cost of assessing expected energy performance, both for code compliance and for 'above-code' rating systems, such as LEED. This cost is borne by the building owner and/or by utilities running new-construction incentive programs, which originally motivated Xcel Energy to fund the creation of this capability. In India, one of the biggest challenges in the implementation of ECBC is the non-availability of trained professionals who can perform energy simulation as per the complex requirements of energy codes. This procedure for automatically generating reference building models is expected to be of great benefit to architects, designers, HVAC consultants etc. Automated baseline generation, built on a common code base, has great potential to address these problems.

Acknowledgments

References

Architectural Energy Corporation. (2016). *CBECC-Com California Building Energy Code Compliance Software for Nonresidential Buildings.* http://bees.archenergy.com/

ASHRAE. (2013). *ANSI/ASHRAE/IES Standard 90.1-2013. Energy Standard for Buildings Except Low Rise Residential Building.* American society of Heating, Refrigerating, and Air-Conditioning Engineers, Atlanta, Georgia.

California Energy Commission. (2015). *Nonresidential Alternative Calculation Method Reference Manual. CEC-400-2015-025-CMF.* California Energy Commission, Sacramento, California.

Deru, Michael, Kristin Field, Daniel Studer, Kyle Benne, Brent Griffith, and Paul Torcellini. (2011). *U.S. Department of Energy Commercial Reference Building Models of the National Building Stock.* NREL/TP-5500-46861, National Renewable Energy Laboratory, Golden, Colorado.

EnergyPlus. (2016). https://energyplus.net/ Mathur Jyotirmay, Vishal Garg, Vijeta Jangra. (2010). *Energy Conservation Building Code in India: Status, Issues and Opportunities.* ASME 4 th International Conference on Energy Sustainability.

Kumar Satish, Ravi Kapoor, Rajan Rawal, Sanjey Seth, Archana Walia. (2010). *Developing an Energy Conservation Building Code Implementation Strategy in India.* ACEEE Summer Study on Energy Efficiency in Buildings.

NRC (2015). National Energy Code of Canada for Buildings. http://www.nrc-cnrc.gc.ca/eng/publications/codes_centre/2015_national_energy_code_buildings.html

OpenStudio. (2016). https://www.openstudio.net/

OpenStudio-standards. (2016) GitHub repository. https://github.com/NREL/openstudio-standards

Thornton B.A., M. Rosenberg, E.E. Richman, W. Wang, Y. Xie, J. Zhang, H. Cho, V.V. Mendon, R.A. Athalye and B. Liu. 2011. *Achieving the 30% Goal: Energy and Cost Savings Analysis of ASHRAE Standard 90.1-2010.* PNNL-20405, Pacific Northwest National Laboratory, Richland, Washington.