C. M. Herr, N. Gu, S. Roudavski, M. A. Schnabel, *Circuit Bending, Breaking and Mending: Proceedings of the 16th International Conference on Computer-Aided Architectural Design Research in Asia,* 19–28.

ANALYSIS MODEL FOR INCREMENTAL PRECISION ALONG DESIGN STAGES

MARCELO BERNAL[1, 2]
1. Georgia Institute of Technology, Atlanta, USA
marcelo.bernal@gatech.edu
2. Universidad Técnica Federico Santa Maria, Valparaiso, Chile

Abstract. With current energy analysis tools, architects and engineers cannot rely on the results of energy analyses because they do not report their level of precision. In addition, current tools also do not deliver feedback in real time. Thus, this research addresses the challenge of obtaining feedback in real-time while gradually increasing precision along design stages. For this purpose, this study merges parametric modelling (PM) technologies and the performance-based design (PBD) paradigm into a general design model. The model is based on a parametric and an energy analysis model that share the parameters of a building. The modular architecture of the model involves four main function types: an input processor, optional analysis functions embedding different calculation methods, a decision-maker, and a report generator function. For every step of the design evolution, the decision-maker function generates a specific tree of analysis functions.

Keywords. Performance; decision-making; extensibility; knowledge-based design; design automation.

1. Introduction

The lack of interoperability between design and evaluation tools hinders the iterative process between design and energy analysis. To bridge this gap, the proposed analysis model addresses the question of how to include technical knowledge in different design stages. The main goals of the research are to obtain feedback in real-time for decision-making (Sanguinetti et al, 2010) and gradually increase the precision of such feedback while design is evolving. Obtaining this kind of feedback from technical domains requires the formalization of such expertise into a set of parameters, constraints, and functions.

This translation of knowledge is feasible since most design and engineering parametric software allows writing customized functions through their application programming interfaces (API). These capabilities allow users synchronize engineering with design knowledge instead of to perform analyses aside of the design process. This procedure allows users to visualize the effect of engineering knowledge directly in the parametric model and minimizes design further modifications. Furthermore, design and engineering knowledge embedded within parametric systems will drive the emergence of a new generation of design tools. The proposed model tackles such a challenge by joining both the performance-based design (PBD) paradigm and parametric modelling technologies (PM) using PBD as a guideline for an example of energy analysis and taking advantage from the parametric capabilities of existing CAD tools. While PBD characterizes how a product executes a given function under stress (Becker 2008), what a building must do instead of how it should be constructed (Kalay, 1999), PM allows design variation based on parameters that drive geometrical relationships (Aish et al, 2003).

The proposed model merges the PBD and PM approaches and synchronizes performance calculators with parametric models. Performance calculators simultaneously evaluate the parametric model of the building after every design variation, providing performance indicators (PI) that support decision-making. If more design resolution or details are added, the evaluation increases the precision of the results. Having different levels of precision from the same analysis model requires a variety of calculation methods selected based on provided inputs and desired outputs. The principle of *information hiding* (Parnas, 1972) is proposed to re-arrange the set of necessary functions embedding the calculation methods. For every evaluation, a new analysis program architecture is assembled. The implications and alternatives of this model will be discussed in the following sections.

2. Fundamental model for energy analysis

The activity-diagram shown in Figure 1 represents the process and different components that execute a regular energy analysis. The final outcome is an indicator of the energy required to maintain comfort temperature (Saguinetti et al 2010). The energy analysis model reads parameters from the parametric model, the weather, and materials. The number of aspects determines the number of required analysis functions containing different calculation methods. Although this particular example deals with energy consumption, the model can be adapted for a wider variety of performance aspects. As shown in the diagram, inputs from the building are combined with weather data, which together define the energy flow through the building (Hagentoft, 2003).

Building inputs such as orientation or shading areas are used for estimating heat gained through direct, diffuse, and reflected solar radiation (Szokolay, 2008). Finally, inputs from material properties are used for estimating energy transmitted through opaque and translucent surfaces. These kinds of models operate within any analysis tool with more or less variation. Outcomes are the result of an intricate flux of parameters among different analysis functions in which every function affects the precision of the results. Indeed, the prediction of results depends on how much knowledge we have, how much we want to invest to develop the model, and how much the prediction influences our design decisions (Paredis, 2007).

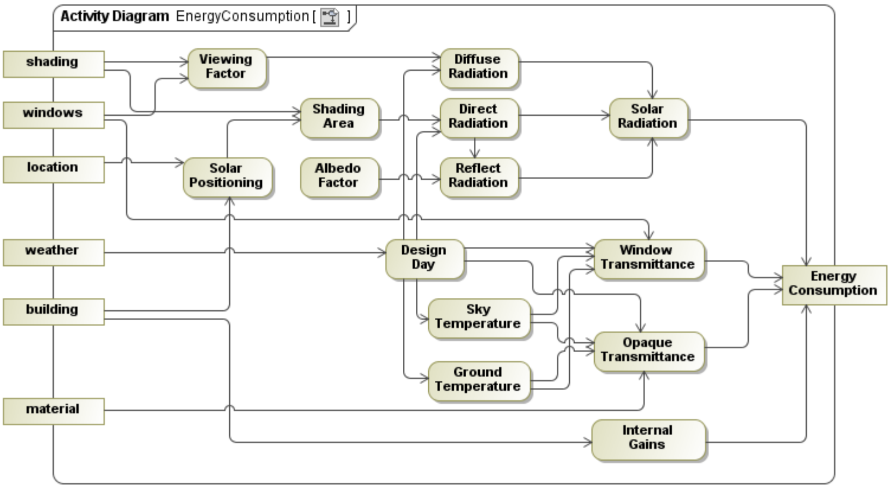

Figure 1. Diagram of the calculator for energy consumption, adapted from Sanguinetti et al

3. From the black-box to glass-box perspective

3.1. THE BLACK-BOX PERSPECTIVE

What happens within current energy analysis tools while the evaluation is executed? Few users can answer this question. Furthermore, how can we be confident about the results of an analysis of something we only partially understand? Most of the analytic tools operate under the black-box perspective. The tools process inputs and provide output hindering the chain of relationships, considerations, and calculation methods involved in the process. This lack of transparency is a key issue since the accuracy of results varies depending on calculation methods. For example, we can examine the process that calculates the temperature of the sky (*Tsky*), a very important parameter involved in the calculation of the heat transferred by radiation through opaque and glazing surfaces. To obtain such a value in Celsius units, three different authors propose different methods.

- Tsky = Tair [0.8 +((Tair -5)-273)/250]^¼ (Bliss, 1961)
- Tsky = Tair - ({1-0.261 exp[-0.000777(273-Tair)^2]}T^4/5.67x10 ^-8) (Idso and Jackson, 1969)
- Tsky = 1.2 Tair -14 (Hagentoft, 2001)

Those three methods use Tair, or current air temperature, as the main input. Nevertheless, later, to obtain Tsky they apply different methods. If we run the proposed calculation methods, we will obtain variations in the final result. Sorting these methods based on their precision is not an easy task for non-experts. Hagentoft proposes a simple way of estimating such a temperature by applying just a factor over Tair. However, determining the degree of precision of the other two methods seems to be more complex. The same dilemma occurs while we are calculating the other parameters involved in energy analyses. For diffuse solar radiation, we can apply a factor over direct solar radiation or perform the actual calculation. We can determine the viewing factor (how much sky we can see from a window to calculate loss because of radiation) by using a single reference point in the centre of the window or integrating several points to improve the accuracy. The definition of a reference design day varies depending on different records from weather data. We could select monthly, daily, or hourly records to determine maximum/minimum values or average values, generating a wide poll of possible design days for different purposes. Users cannot easily determine the precision of current evaluation tools since they hinder all these considerations.

3.2. THE GLASS-BOX PERSPECTIVE

Unlike the dominant black-box perspective, the glass-box perspective represents a different approach based on transparency. This perspective (Paredis, 2007) exposes the inner components of the model as a composition of interactions of black boxes, which makes the set of relationships between calculation methods involved in the process transparent and the assessment of the results by the users easier. Nevertheless, such approaches require some fundamental knowledge about physical phenomena from users if they want to assess what is going on within the glass box. In addition, calculation methods can be added or edited for specific purposes. Discrepancies from the calculation methods described above show that transparency is crucial because it can provide reliable feedback during the design process. From the glass box perspective users should know if they are obtaining results on a monthly or hourly basis, the precision level of the method used to calculate any parameter, and the accumulated degree of uncertainty.

4. Features of the analysis model for incremental precision

4.1. FEEDBACK

The first target is to provide feedback to designers for decision-making. Design exploration is performed under the statement *what happens if, which* is not a naïve statement. On the contrary, it is an exploration based on implicit design knowledge, rules of thumb, previous experiences, good practices, or design guidelines. Along such an exploration, the analysis model should provide feedback from which users can visualize the outcomes or consequences of changes in the values of the parameters. This technique, which is based on exported parameters from the model through spreadsheets or simple text files, minimizes the complexity derived from the interoperability between formats since only parameters are shared between the parametric and the analysis model. Thus, having completed any modification in the parametric model, analysis model automatically updates and re-evaluates the parameters. Because most current parametric modelling software can share parameters, the approach of the proposed model can be applied for other kinds of analyses.

4.2. INCREMENTAL PRECISION

4.2.1. Shifting parameters

Along the design process, new parameters are constantly created and deleted. Parametric models typically have two kinds of parameters that are continuously evolving through the process: driving parameters in the top of the hierarchy (e.g., floor-to-floor height), and driven parameters (e.g., façade square feet). The analysis system should be robust enough to recognize the addition, edition, or deletion of those parameters.

4.2.2. Material parameters

The evolving material parameters involve material specification of the building components (e.g., U-values or solar heat gain coefficient SHGC for different window types). Although decisions regarding material specifications take place in late design stages, users can make some rough assumptions to perform the initial set of analyses. An analysis model for incremental precision must identify the replacement of default values for actual parameters and report the degree of precision of the analysis while these new parameters are added.

4.2.3. Weather data

Weather data information provide annual, monthly, daily, or even hourly average values. This variable source of information drastically influences the degree of precision of the performed analysis. Input variables such as ambient, ground, and sky temperatures, and direct, diffuse, and reflected solar radiation vary depending on the kind of information obtained from weather data. The analysis model should handle such a diversity of inputs.

4.2.4. Calculation methods

Analysis functions are another key component of the analysis model since results vary from rough estimations to more accurate values depending on the level of design definition (i.e., from monthly to hour-by-hour solar energy gains). Thus, analysis functions with similar signatures, embedding different calculation methods, could be assembled into a custom analysis model according to the complexity of the task, or the same function could provide different outcomes. If an input is null, the function could assign a default value or perform different calculations depending on the input types. Moreover, in the energy analysis field, we can identify two approaches that define calculation methods for physical phenomena: rough factors, derived from expert knowledge, and equations. For example, to estimate diffuse solar radiation, we can apply a factor on direct solar radiation or perform a more complex calculation obtaining different results from a lower to higher level of precision. In an adaptable analysis model, such considerations should be included in estimates of the degree of precision of any analysis.

4.3. EXTENSIBILITY

An adaptable analysis model grows by extending existing technology (Schaefer, 2010), integrating new knowledge, and re-using previous resources (Lee et al, 2005) to address new design challenges. Continuous improvement and infinite growth are fundamental features of an extensible analysis model. Continuous improvement maintains openness of the model by refining algorithms according to shifting conditions without affecting its functionality. Infinity growth implies that changes are implemented on a working platform based on the potential of the model to be modified by adding new functionalities or extending existing ones.

5. System modularity for incremental precision

The proposed model is based on a modular architecture by applying information-hiding principles (Parnas, 1972), that facilitate maintenance, allow parallel development, and increase flexibility since components can be added or replaced without affecting adjacent modules.

5.1. INFORMATION-HIDING PRINCIPLES

First, we must define module and function for the purpose of this paper. While a module is related to a responsibility task, a function is the actual implementation. A module has little knowledge about another module, usually it only exchanges inputs and outputs in the required format, and hides all the internal calculations. One or multiple modules, based on their affinity, can be embedded within a function, which can be constantly re-assembled, forming temporary structures responding to a given task without affecting the entire system (Parnas, 1972). Continuing with the example of the temperature of the sky and its three alternative calculation methods, we have two possible ways of implementing them: three alternative modules implemented in separated functions selected by external conditionals or one major function containing the three alternative modules. In the second case, internal conditionals decide what module should be used based on the input types.

5.2. ANALYSIS MODEL ARCHITECTURE

The proposed model (Figure 2) has four function types organized in a shifting hierarchical structure: input processor (IP), optional analysis functions (OAF), decision-maker (DM), and report generator (RG). IP handles inputs from the parametric model, material properties, and weather data. OAF perform actual evaluations. DM evaluates processed inputs, and based on this information, select different OAs from the options to perform the analysis. Finally, the RG collects results from the analysis functions and plots them for feedback.

5.2.1. Input processor (IP)

Inputs come from different sources such as dimensions from parametric models, weather data, or material properties. In addition, they are defined at different times along the design process. While parameters about massing studies of the building are shared in early design stages, parameters about material properties are defined in later stages. The task of the input processing functions is to properly evaluate and classify the incoming parameters to deliver them to the analysis functions. The input processing function must read different formats from weather data in spreadsheets and parameters from text files to inputs provided by users. This function must also check if some parameters have or do not have information to assign default values or declare them null if necessary.

5.2.1. Optional analysis functions (OAF)

OAFs embed the calculation methods. Functions for design day, solar positioning, shaded windows area, albedo factor, and viewing factor must feed functions for sky and ground temperature, and diffuse, direct, and reflected solar radiation to determine the heat transmitted through walls and windows, the total solar radiation, and the internal gains that together determine the energy consumed to maintain thermal comfort. The OAFs can be implemented in two different ways: parallel functions with alternative methods to calculate the same value and one general function containing these multiple methods. In the first alternative, the DM selects what function must be used; in the second alternative, internal conditionals switch between different methods. In addition, OAFs should be as independent as possible, avoiding unnecessary cross linkage among modules to guarantee the required flexibility to re-arrange them along incremental steps.

5.2.1. Decision-maker (DM)

The DM controls the overall workflow. It reads pre-processed parameters from the inputs processing function and, based on such information, defines what function, or set of them, form the OAF is invoked to perform the analysis. The selected set of OAF will form a temporary tree depending on the required analysis and the desired degree of precision of the analysis.

5.2.1. Report generator (RG)

RG collects the outcomes and plot reports about current value of key parameters, the degree of accuracy of the overall analysis, and the amount of energy required for thermal comfort. Key parameters values can be represented through charts, but the graphic interface is a discussion for further steps. Although precision should increase while users add more resolution in terms of material specification, the uncertainty regarding material property inputs and the whole set of calculation methods involved in the analysis can affect the reliability of the results. The RG must take uncertainty into account for predictions, a coefficient (de Wit and Augenbroe, 2002) derived from inputs and calculation method. This coefficient is important since design decisions imply risk under uncertainty levels.

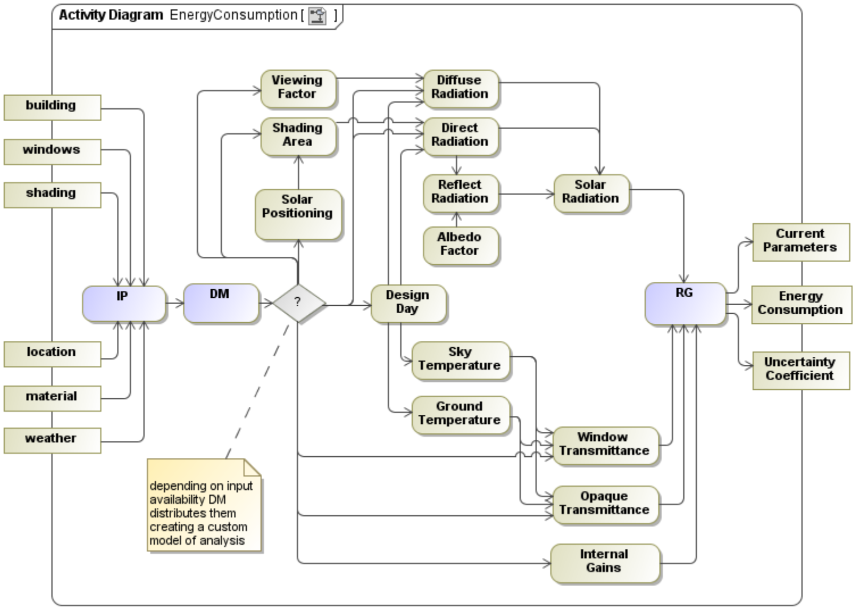

Figure 2. Activity diagram of the workflow of the proposed model

6. Discussion

The proposed model for incremental precision along sequential design stages provides support for decision-making throughout the design evolution. The technique in which the parametric and analysis models share parameters can provide feedback in real-time after every modification while the design increases the resolution of the material specifications. The model switches from the black-box to the glass-box perspective in terms of procedure, making the analysis process more transparent. The approach of this model reveals and clarifies the resolution of the analysis and the related levels of precision and uncertainty, but it demands that users monitor the analysis and the outcomes. Although this model adds transparency to the analysis, the model by itself does not achieve fully reliable results since the input and the calculation methods may introduce uncertainty and variations that affect the outcome. Discrepancies between different energy analysis handbooks show that no absolute methods exist, and users must have enough background regarding physical phenomena to make choices among alternative calculation methods. The consistency between degree of precision of the analysis and the importance of a design decision must be balanced in every design stage since different decisions demand different information to be made. While some decisions in preliminary stages only require reference values, others in late stages could require more elaborated evaluations. Thus, the adequate degree of precision of the analysis is determined by the kind of decisions made by users. The precision of the predictions depends on one's level of knowledge, the extent of the investment in developing the model, and the degree of influence of the prediction on design decisions. Finally, knowledge from other domains can be translated to the analysis model by applying the same modular approach to extend the boundaries of the design environment.

Acknowledgements

I want to thank to Paola Sanguinetti, Maher El-khaldi, and Matthew Erwing from the Georgia Institute of Technology for their contribution to develop the energy consumption calculator during the Building Envelope class, Fall 2009, taught by Professor Fried Augenbroe.

References

Becker, R.: 2008, Fundamentals of performance-based building design, *Building Simulation Journal*, **1**(4), 356–371.

Bliss, R. W.: 1961, Atmospheric radiation near the surface of the ground: A summary for engineers. *Solar Energy,* **5**(3), 103–120.

Eastman, C.: 1999, Information exchange architectures, *in* Eastman, C. (ed), *Building product models,* CRC Press, Boca Raton, Florida, 321–349.

de Wit, S. and Augenbroe, G.: 2002, Analysis of uncertainty in building design evaluations and its implications. *Energy and Buildings,* **34**(9): 951–958.

Hagentoft, C.E.: 2001, *Introduction to building physics*, Studentlitteratur AB Press, Lund, Sweden.

Idso, S. and Jackson, R.: 1968, Significance of fluctuations in sky radiant emittance for infrared thermometry, *Agronomy Journal*, **60**(4), 388–392.

Kalay, Y.: 1999, Performance-based design, *Automation in Construction*, 8, 395–409.

Lee, G., Sacks, R., and Eastman, C.: 2005, Specifying parametric building object behavior (BOB) for building information modelling system, *Automation in construction*. **15**(6), 758–776.

Paredis C.: 2007, Slides from ME6105 Modeling and Simulation in Design, *Schools of Mechanical Engineering, Georgia Institute of Technology.* Atlanta, Georgia.

Parnas, D.L.: 1972, On the criteria to be used in decomposing systems into modules, *Communications of the ACM*, 15, 12.

Parnas D.L.: 1972, Some conclusion from an experiment in software engineering techniques, *AFIPS Joint Computer Conferences Proceedings*. Anaheim, California.

Sanguinetti, P., Bernal, M., El-Khaldi, M. and Erwing, M.: 2010, Real-time design feedback: Coupling Performance-Knowledge with Design, *Proceedings of Symposium on Simulation for Architecture and Urban Design,* Orlando, Florida, 23–30.

Shaefer, D.: 2010, Slides from ME6102 Open engineering systems, *Schools of Mechanical Engineering, Georgia Institute of Technology,* Atlanta, Georgia

Shea, K., Aish, R. and Gourtovaia, M.: 2003, Towards Integrated Performance-Based Generative Design Tools, *Proceedings of the 21th Conference in Educationand Reaseach in Computer Aided Design in Europe, eCAADe 2003*, Graz, ,Austria, 553–560

Szokolay S.V.: 2008, *Introduction to architectural science*, Architectural Press, Oxford, UK